

Tru64 UNIX

Network Administration: Connections

Part Number: AA-RH9CC-TE

June 2001

Product Version: Tru64 UNIX Version 5.1A or higher

This manual is intended for experienced system or network administrators. It describes the tasks for configuring your system to operate in a network, for configuring the network services, and for day-to-day management of the network, network interfaces, and network services. This manual also includes information for solving problems that might arise while using the network and network services.

© 2001 Compaq Computer Corporation

COMPAQ, the Compaq logo, Compaq Insight Manager, DECnet, TruCluster, and VAX Registered in U.S. Patent and Trademark Office. OpenVMS and Tru64 are trademarks of Compaq Information Technologies Group, L.P.

Microsoft, Windows, and Windows NT are trademarks of Microsoft Corporation. Motif, OSF/1, UNIX, and X/Open are trademarks of The Open Group. All other product names mentioned herein may be trademarks of their respective companies.

Confidential computer software. Valid license from Compaq required for possession, use, or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Compaq shall not be liable for technical or editorial errors or omissions contained herein. The information in this document is provided "as is" without warranty of any kind and is subject to change without notice. The warranties for Compaq products are set forth in the express limited warranty statements accompanying such products. Nothing herein should be construed as constituting an additional warranty.

Contents

About This Manual

1 Overview to Network Administration

1.1	Introduction to <i>Network Administration: Connections</i>	1-1
1.2	Administrative Methods	1-2
1.2.1	SysMan Menu	1-2
1.2.1.1	Quick Setup	1-4
1.2.1.2	Network Setup Wizard	1-5
1.2.1.3	Command-Line Integration	1-6
1.2.2	Compaq Insight Manager	1-7
1.2.3	Other Interfaces	1-8
1.2.4	Manually Editing Configuration Files	1-9
1.2.5	Installation and Configuration Cloning	1-9

2 Basic Network Connections

2.1	Network Environment	2-1
2.1.1	Network Interface	2-2
2.1.1.1	Multiple Interfaces in a Subnet	2-3
2.1.1.2	NetRAIN	2-4
2.1.1.3	Link Aggregation	2-6
2.1.2	Routing	2-7
2.2	Preparing for the Configuration	2-8
2.2.1	Information for Interfaces and Daemons	2-8
2.2.1.1	All Network Interfaces	2-9
2.2.1.2	Token Ring Interface	2-12
2.2.1.3	NetRAIN Interface	2-12
2.2.1.4	LAG Interface	2-12
2.2.1.5	rwhod Daemon	2-12
2.2.1.6	routed Daemon	2-13
2.2.1.7	Gateways File	2-14
2.2.1.8	gated Daemon	2-14
2.2.1.9	IP Router	2-15
2.2.2	Information for Network Files	2-16
2.2.2.1	Static Routes File (/etc/routes)	2-16
2.2.2.2	Hosts File (/etc/hosts)	2-17

2.2.2.3	Hosts Equivalencies File (/etc/hosts.equiv)	2-18
2.2.2.4	Networks File (/etc/networks)	2-18
2.3	Configuring the Network Components	2-19
2.3.1	Configuring Network Interfaces	2-19
2.3.2	Configuring the rwhod Daemon	2-21
2.3.3	Configuring the routed Daemon	2-22
2.3.4	Configuring the gated Daemon	2-23
2.3.5	Configuring the System as an IP Router	2-24
2.3.6	Configuring the Static Routes File	2-25
2.3.7	Configuring the hosts File	2-26
2.3.8	Configuring the hosts.equiv File	2-26
2.3.9	Configuring the networks File	2-27
2.3.10	Configuring IP Aliases	2-27
2.4	Managing Multiple Network Interfaces	2-28
2.4.1	Configuring NetRAIN	2-28
2.4.2	Monitoring NetRAIN Activity	2-32
2.4.3	Configuring a Link Aggregation Group	2-33
2.5	Enabling Access Filtering on an Interface	2-35
2.6	Displaying and Modifying the FDDI Parameters	2-35
2.7	Managing Token Ring Source Routing	2-37
2.8	Displaying and Modifying the Token Ring IP MTU Size	2-39
2.9	Managing Network Quality of Service	2-40
2.9.1	Managing the Traffic Control Subsystem	2-41
2.9.2	Managing RSVP	2-41
2.9.2.1	Starting and Stopping rsvpd	2-41
2.9.2.2	Adding and Deleting Network Interfaces	2-42
2.9.2.3	Displaying RSVP Session Information	2-42

3 Internet Protocol Version 6

3.1	Introduction to IPv6	3-1
3.2	Terms	3-2
3.3	IPv6 Addressing	3-2
3.3.1	Address Text Representation	3-2
3.3.2	Types of Addresses	3-3
3.3.2.1	Unicast Address	3-4
3.3.2.2	Multicast Address	3-7
3.3.3	Address Prefixes	3-9
3.3.4	Address Autoconfiguration	3-9
3.3.5	Address Resolution	3-10
3.3.6	Address Assignment	3-10
3.3.6.1	Aggregatable Global Unicast Address Format	3-11

3.3.6.2	Aggregatable Testing Address Format	3-12
3.4	IPv6 Environment	3-13
3.5	Planning IPv6	3-17
3.5.1	Verifying IPv6 Support in the Kernel	3-17
3.5.2	Preparing for the Configuration	3-18
3.5.2.1	DNS/BIND	3-21
3.5.2.2	Configured Tunnel	3-21
3.5.2.3	Router	3-22
3.5.2.4	Manual Routes	3-22
3.5.3	Configuring Systems in Sample IPv6 Configurations	3-23
3.5.3.1	Simple Host-to-Host Configuration	3-23
3.5.3.2	Host-to-Host with Router Configuration	3-23
3.5.3.3	IPv6 Network-to-IPv6 Network with Router Configuration	3-24
3.5.3.4	Multiple IPv6 Networks and Multiple Routers Configuration	3-25
3.5.3.5	Host-to-Host over Tunnel Configuration	3-26
3.5.3.6	Host-to-Router over Tunnel Configuration	3-26
3.5.3.7	IPv6 Network-to-IPv6 Network over Tunnel Configuration	3-28
3.6	Configuring IPv6 on Your System	3-29
3.6.1	Configuring an IPv6 Host	3-30
3.6.2	Configuring an IPv6 Router	3-31
3.7	Postconfiguration Tasks	3-33
3.7.1	Connecting to the 6bone Network	3-34
3.7.2	Initializing a New Interface for IPv6	3-35
3.7.2.1	Setting the IPv6 Interface Identifier	3-35
3.7.3	Removing IPv6 from an Interface	3-35
3.7.4	Creating a Configured Tunnel	3-36
3.7.5	Adding an Address to an Interface	3-36
3.7.6	Deleting an Address from an Interface	3-37
3.7.7	Adding or Deleting a Default Router	3-37
3.7.8	Manually Adding a Route for an On-Link Prefix	3-37
3.7.9	Configuring Routing Support in the Kernel	3-38
3.7.10	Editing the Run-Time Configuration File	3-38
3.7.11	Editing the Router Configuration File	3-40
3.7.12	Tuning the Kernel Subsystems	3-41
3.8	IPv6 Daemon Log Files	3-41

4 Asynchronous Transfer Mode

4.1	ATM Environment	4-1
-----	-----------------------	-----

4.1.1	Classical IP Environment	4-2
4.1.2	LAN Emulation Environment	4-3
4.1.3	IP Switching	4-4
4.2	Planning ATM	4-6
4.2.1	Verifying That the ATM Subsets Are Installed	4-6
4.2.2	Configuring ATM into the Kernel	4-6
4.2.3	Preparing for the Configuration	4-7
4.2.3.1	Adapter Information	4-7
4.2.3.2	Classical IP Information	4-9
4.2.3.3	LAN Emulation Information	4-11
4.2.3.4	IP Switching Information	4-13
4.3	Configuring ATM	4-15
4.3.1	Configuring an ATM Adapter	4-16
4.3.2	Configuring Classical IP	4-17
4.3.2.1	Creating PVC Mappings on Your ATM Switch	4-17
4.3.2.2	Adding Servers to the atmhosts File	4-17
4.3.2.3	Adding Hosts to the hosts Database	4-18
4.3.2.4	Running the ATM Configuration Application	4-19
4.3.2.5	Configuring the Classical IP Logical Interface	4-20
4.3.2.6	Adding Static Routes (SVC only)	4-20
4.3.2.7	Verifying the PVC Configuration (PVCs only)	4-20
4.3.3	Configuring LAN Emulation	4-20
4.3.3.1	Adding Servers to the atmhosts File	4-21
4.3.3.2	Adding Hosts to the hosts Database	4-21
4.3.3.3	Running the ATM Configuration Application	4-21
4.3.3.4	Configuring the LAN Emulation Logical Interfaces ...	4-22
4.3.4	Configuring IP Switching	4-22
4.3.4.1	Adding IP Addresses to the hosts File	4-23
4.3.4.2	Running the ATM Configuration Application	4-23
4.3.4.3	Configuring the IP Switching Logical Interfaces	4-25
4.3.4.4	Adding Routes	4-25
4.4	Managing the ATM Environment	4-25
4.4.1	ATM Networking and Displaying Information About ATM Networks	4-26
4.4.2	Signaling Module	4-26
4.4.3	Classical IP Environment	4-26
4.4.4	LAN Emulation Environment	4-27
4.4.4.1	Managing LAN Emulation Clients	4-27
4.4.4.2	Displaying the LE-ARP Table	4-27
4.4.5	IP Switching	4-27
4.4.6	ATM Subsystem Messages	4-28

5 Dynamic Host Configuration Protocol

5.1	DHCP Environment	5-2
5.1.1	DHCP Parameter Assignment	5-2
5.1.2	DHCP and Security	5-3
5.2	Planning DHCP	5-4
5.2.1	Verifying Installation of the DHCP Software	5-4
5.2.2	Preparing for the Configuration	5-4
5.2.2.1	Server/Security Parameters	5-4
5.2.2.2	Information for Basic DHCP Parameters	5-8
5.3	Configuring a DHCP Server	5-12
5.3.1	Configuring Server/Security Parameters	5-13
5.3.2	Configuring IP Ranges	5-13
5.3.3	Configuring Host Name Lists	5-14
5.3.4	Configuring a Subnetwork	5-14
5.3.5	Configuring a DHCP Client Node	5-15
5.3.6	Setting Group Parameters	5-16
5.3.7	Starting the DHCP Server (joined)	5-17
5.4	Managing DHCP	5-17
5.4.1	Starting the DHCP Client	5-18
5.4.2	Monitoring DHCP Client Configuration	5-18
5.4.3	Mapping Client IP Addresses Permanently	5-19
5.4.4	Restricting Access to the DHCP Server	5-19
5.4.5	Configuring a BOOTP Client	5-20
5.4.6	Disabling DHCP Address Assignment	5-20

6 Point-to-Point Connections

6.1	Serial Line Internet Protocol (SLIP)	6-1
6.1.1	SLIP Environment	6-1
6.1.2	Planning SLIP	6-2
6.1.2.1	Verifying the Hardware	6-3
6.1.2.2	Preparing for the Configuration	6-4
6.1.3	Configuring SLIP	6-7
6.1.3.1	Configuring a Dial-In System	6-8
6.1.3.2	Configuring a Dial-Out System	6-9
6.1.4	Terminating a SLIP Dial-Out Connection	6-10
6.2	Point-to-Point Protocol (PPP)	6-11
6.2.1	PPP Environment	6-11
6.2.1.1	Chat Scripts	6-13
6.2.1.2	PPP Options	6-14
6.2.1.3	Authentication	6-15

6.2.2	Planning PPP	6-16
6.2.2.1	Verifying the Hardware	6-16
6.2.2.2	Verifying PPP Support in the Kernel	6-16
6.2.2.3	Preparing for Configuration	6-17
6.2.3	Configuring a Dial-Out System with PPP	6-19
6.2.3.1	Setting Up Initial Communications for a Dial-Out System	6-20
6.2.3.2	Creating Options Files for a Dial-Out System	6-21
6.2.3.3	Setting Up Authentication for a Dial-Out System	6-22
6.2.3.3.1	Creating Entries in the PAP Secrets File	6-22
6.2.3.3.2	Creating Entries in the CHAP Secrets File	6-23
6.2.3.4	Setting Up Message Logging	6-23
6.2.3.5	Initiating and Monitoring a PPP Connection	6-24
6.2.3.6	Connecting to a Microsoft NT Remote Access Server ..	6-25
6.2.3.6.1	Configuring an NT RAS Server	6-25
6.2.3.6.2	Solving Microsoft CHAP Authentication Problems	6-26
6.2.4	Configuring a Dial-In System with PPP	6-26
6.2.4.1	Setting Up Initial Communications for a Dial-In System	6-27
6.2.4.2	Creating Options Files for a Dial-In System	6-27
6.2.5	Terminating PPP Connections	6-28
6.3	Guidelines for Using Modems	6-29
6.3.1	Using the Correct Modem Cables	6-29
6.3.2	Configuring a System for Dial-In Access	6-29
6.3.2.1	Setting Up a Modem for Dial-In Access	6-31
6.3.3	Configuring Your System for Dial-Out Access	6-32
6.3.3.1	Creating Entries in the /etc/remote File	6-33

7 Local Area Transport Connections

7.1	LAT Environment	7-1
7.1.1	Types of LAT Connections	7-2
7.1.2	Controlling Access in a LAT Network	7-3
7.1.3	Specifying Passwords for Remote Services	7-4
7.1.4	Load Balancing	7-4
7.2	Planning LAT	7-4
7.2.1	Verifying That the LAT Subset Is Installed	7-4
7.2.2	Verifying DLB Support in the Kernel	7-4
7.2.3	Preparing for the Configuration	7-5
7.3	Configuring LAT	7-6
7.3.1	Configuring LAT with latsetup	7-7
7.3.2	Starting and Stopping LAT	7-7

7.3.3	Creating a LAT Startup File	7-8
7.3.4	Customizing the inittab File	7-9
7.3.5	Running LAT Over Specific Network Adapters	7-10
7.4	Configuring LAT Connections	7-10
7.4.1	Setting Up Printers	7-10
7.4.1.1	Setting Up the Printer on a Terminal Server	7-11
7.4.1.2	Testing the Port Configuration	7-12
7.4.1.3	Setting Up a Service Node for the Printer	7-12
7.4.1.4	Setting Up the Print Spooler on the Service Node	7-12
7.4.1.5	Testing the Printer	7-13
7.4.2	Setting Up Host-Initiated Connections	7-13
7.4.2.1	Setting Up the System for Host-Initiated Connections	7-13
7.4.2.2	Program Interface	7-14
7.4.3	Setting Up Outgoing Connections	7-15
7.4.3.1	Setting Up the System for Outgoing Connections	7-15
7.4.3.2	Program Interface	7-15
7.4.4	Setting Up the LAT/Telnet Gateway	7-16
7.4.5	Creating Dedicated or Optional Services	7-17
7.4.6	Providing a Dedicated tty Device on a Terminal	7-18
7.4.6.1	Setting Up a Dedicated tty Device	7-18
7.4.6.2	Removing a Dedicated tty Device	7-19

8 Solving Network and Network Services Problems

8.1	Using the Diagnostic Map	8-1
8.2	Getting Started	8-2
8.3	Solving IPv4 Network Problems	8-5
8.4	Solving IPv6 Network Problems	8-9
8.4.1	Solving IPv6 Host Problems	8-10
8.4.2	Solving IPv6 Router Problems	8-15
8.5	Solving ATM Problems	8-21
8.5.1	Solving CLIP Problems	8-22
8.5.2	Solving LANE Problems	8-24
8.5.3	Solving IP Switching Problems	8-26
8.6	Solving DHCP Problems	8-29
8.7	Solving SLIP Problems	8-32
8.8	Solving PPP Problems	8-34
8.9	Solving LAT Problems	8-36

9 Using the Problem Solving Tools

9.1	Detecting Network Interface Failures	9-1
-----	--	-----

9.1.1	Configuring and Deconfiguring NIFF	9-2
9.1.2	Viewing NIFF Events	9-3
9.2	Testing Access to Internet Network Hosts	9-4
9.3	Displaying Network Statistics	9-6
9.4	Displaying and Modifying the Internet (IPv4) to MAC Address Translation Tables	9-7
9.5	Displaying a Datagram's Route to a Network Host	9-8
9.6	Displaying Headers of Packets on the Network	9-10
9.7	Viewing the Error Log File	9-11
9.8	Viewing the syslogd Daemon Message Files	9-11

10 Reporting Network Problems

10.1	Gathering General Information	10-1
10.2	Gathering Hardware Architecture Information	10-2
10.3	Gathering Software Architecture Information	10-2

A Monitoring the Network Interfaces

A.1	Monitoring the Ethernet Interface	A-1
A.2	Monitoring the FDDI Interface	A-4
A.2.1	FDDI Counters	A-6
A.2.2	FDDI Status	A-9
A.2.3	FDDI Characteristics	A-17
A.3	Monitoring the Token Ring Interface	A-19
A.3.1	Token Ring Counters	A-20
A.3.2	Token Ring and Host Information	A-23

Index

Examples

2-1	Creating One NetRAIN Set	2-31
2-2	Creating Two NetRAIN Sets	2-31
2-3	Sample Link Aggregation Statements	2-34
3-1	Sample IPv6 Host Configuration Variables	3-40
3-2	Sample IPv6 Router Configuration Variables	3-40
3-3	Sample ip6rtrd.conf File	3-41
7-1	Sample /etc/latstartup.conf File	7-9

Figures

1-1	SysMan Menu	1-3
-----	-------------------	-----

1-2	Quick Setup	1-4
1-3	Network Setup Wizard	1-6
1-4	Compaq Management Agents	1-8
2-1	Sample Single Interface Configuration	2-2
2-2	Sample Multiple Interfaces in a Subnet	2-3
2-3	Sample NetRAIN Configuration	2-5
2-4	Sample Link Aggregation Configuration	2-7
2-5	Interface and Daemon Worksheet	2-9
2-6	Network Files Worksheet	2-16
3-1	Creating an Interface ID from a MAC Address	3-5
3-2	Simple Host-to-Host Configuration	3-14
3-3	Host-to-Host with Router Configuration	3-14
3-4	IPv6 Network-to-IPv6 Network with Router Configuration ...	3-15
3-5	Multiple IPv6 Networks and Multiple Routers Configuration .	3-15
3-6	Host-to-Host over Tunnel Configuration	3-16
3-7	Host-to-Router over Tunnel Configuration	3-16
3-8	IPv6 Network-to-IPv6 Network over Tunnel Configuration ...	3-17
3-9	IPv6 Configuration Worksheet	3-19
4-1	Classical IP over an ATM Network	4-3
4-2	Emulated LAN over an ATM Network	4-4
4-3	IP Switching over an ATM Network	4-5
4-4	ATM Setup Worksheet	4-8
4-5	ATM Classical IP Worksheet	4-10
4-6	ATM LAN Emulation Worksheet	4-12
4-7	ATM IP Switching Worksheet	4-14
5-1	DHCP Configuration (acme-net)	5-2
5-2	DHCP Server/Security Parameters Worksheet	5-5
5-3	Basic DHCP Parameters Worksheet	5-9
6-1	Sample Simple SLIP Configuration	6-2
6-2	SLIP Configuration with Gateway System	6-2
6-3	SLIP Setup Worksheet	6-4
6-4	Simple PPP Configurations	6-12
6-5	Network PPP Configuration	6-13
6-6	PPP Setup Worksheet	6-17
7-1	Sample LAT Network Configuration	7-2
7-2	LAT Setup Worksheet	7-5

Tables

2-1	Comparison of Multiple Interface Configurations	2-2
2-2	Options to the fddi_config Command	2-35
2-3	Options to the srconfig Command	2-37

3-1	Well-Known Multicast Addresses	3-9
3-2	IPv6 Address Types and Prefixes	3-9
4-1	ATM Kernel Options	4-7
6-1	Types of Null Modem Cable	6-3
6-2	Mandatory startslip Subcommands	6-5
6-3	Optional startslip Subcommands	6-6
6-4	slhosts File Options	6-7
6-5	Modem Commands for Dial-Out Access	6-9
6-6	slhosts File Options	6-19
6-7	Types of Modem Cable	6-29
6-8	Modem Commands for Dial-In Access	6-31
7-1	LAT Parameters	7-8
8-1	Problem Solving Starting Points	8-3
9-1	Options to the ping Command	9-4
9-2	Options to the netstat Command	9-6
9-3	Options to the traceroute Command	9-8

About This Manual

This manual describes how to configure and manage network interfaces and network transports, and solve network problems that might arise on systems running the Tru64™ UNIX operating system software.

This manual assumes that the operating system software and the appropriate networking subsets are installed.

Audience

This manual is intended for system and network administrators responsible for configuring and managing network services. Administrators are expected to have knowledge of operating system concepts, commands, and configuration. It is also helpful to have knowledge of Transmission Control Protocol/Internet Protocol (TCP/IP) networking concepts and network configuration; this manual is not a TCP/IP networking tutorial.

New and Changed Features

The *Network Administration* manual has been reorganized into two volumes: *Network Administration: Connections* and *Network Administration: Services*.

This volume, *Network Administration: Connections*, contains all of the information regarding network connections and transports that previously existed in the *Network Administration* manual, plus the following new and revised sections:

- A new section on how to configure link aggregation
- A new section on how to use the Network Interface Failure Finder (NIFF) to monitor network interface connectivity

Organization

The *Network Administration: Connections* manual is divided into several chapters, each of which contains information about configuring a different connection or transport. The manual also includes appendixes that contain supplemental information.

The following list describes the content in more detail:

<i>Chapter 1</i>	Describes network administration and lists the components that this manual covers
<i>Chapter 2</i>	Describes the tasks to administer the basic network connections on Internet Protocol Version 4 (IPv4) networks
<i>Chapter 3</i>	Describes the tasks to administer Internet Protocol Version 6 (IPv6) networks
<i>Chapter 4</i>	Describes the tasks to administer an Asynchronous Transfer Mode (ATM) network adapter
<i>Chapter 5</i>	Describes the tasks to administer the Dynamic Host Configuration Protocol (DHCP)
<i>Chapter 6</i>	Describes the tasks to administer point-to-point connections
<i>Chapter 7</i>	Describes the tasks to administer Local Area Transport (LAT)
<i>Chapter 8</i>	Describes how to diagnose network problems
<i>Chapter 9</i>	Describes the various diagnostic tools available to help solve problems
<i>Chapter 10</i>	Describes how to report problems to Compaq and the information you need to provide
<i>Appendix A</i>	Describes how to monitor the Ethernet, Fiber Distributed Data Interface (FDDI), and token ring network interfaces by using the <code>netstat</code> command

Related Documents

For more information about Tru64 UNIX networking and communications, see the following books:

- *Network Administration: Services*
Provides information about the network services that run over the connections and transports covered in this manual. Explains how to configure and manage the following services and applications:
 - Domain Name System (DNS)
 - Network Information Service (NIS)
 - Network File System (NFS)
 - UNIX-to-UNIX Copy Program (UUCP)
 - Network Time Protocol (NTP)
 - Mail system, including sendmail, the Post Office Protocol (POP) and the Internet Message Access Protocol (IMAP)
 - Simple Network Management Protocol (SNMP)
- *Command and Shell User's Guide*

Introduces users to the basic uses of commands and shells in the operating system.

- *JOIN Server Administrator's Guide* by Join Systems, Inc.

Provides more detailed information about implementing the Dynamic Host Configuration Protocol in your network. This manual can be accessed by opening the following file with a web browser:

</usr/doc/join/TOC.html>

- Request for Comments (RFC)

Many sections of this manual refer to RFCs (for example, RFC 1577) for more information about certain networking topics. These documents publicize Internet Standards, new research concepts, and status memos about the Internet. You can access the full range of RFC documents and more information about the Internet Engineering Task Force (IETF) at the following URL:

<http://www.ietf.org>

- Best Practices

Compaq Tru64 UNIX Best Practices describe some networking concepts and tasks, as well as other topics. You can find these documents on the Tru64 UNIX Publications Home Page at the following URL:

<http://www.tru64unix.compaq.com/docs/>

Icons on Tru64 UNIX Printed Manuals

The printed version of the Tru64 UNIX documentation uses letter icons on the spines of the manuals to help specific audiences quickly find the manuals that meet their needs. (You can order the printed documentation from Compaq.) The following list describes this convention:

- G Manuals for general users
- S Manuals for system and network administrators
- P Manuals for programmers
- R Manuals for reference page users

Some manuals in the documentation help meet the needs of several audiences. For example, the information in some system manuals is also used by programmers. Keep this in mind when searching for information on specific topics.

The *Documentation Overview* provides information on all of the manuals in the Tru64 UNIX documentation set.

Reader's Comments

Compaq welcomes any comments and suggestions you have on this and other Tru64 UNIX manuals.

You can send your comments in the following ways:

- Fax: 603-884-0120 Attn: UBPG Publications, ZKO3-3/Y32
- Internet electronic mail: `readers_comment@zk3.dec.com`

A Reader's Comment form is located on your system in the following location:

```
/usr/doc/readers_comment.txt
```

Please include the following information along with your comments:

- The full title of the manual and the order number. (The order number appears on the title page of printed and PDF versions of a manual.)
- The section numbers and page numbers of the information on which you are commenting.
- The version of Tru64 UNIX that you are using.
- If known, the type of processor that is running the Tru64 UNIX software.

The Tru64 UNIX Publications group cannot respond to system problems or technical support inquiries. Please address technical questions to your local system vendor or to the appropriate Compaq technical support office. Information provided with the software media explains how to send problem reports to Compaq.

Conventions

This document uses the following typographic conventions:

%

\$

A percent sign represents the C shell system prompt. A dollar sign represents the system prompt for the Bourne, Korn, and POSIX shells.

#

A number sign represents the superuser prompt.

% **cat**

Boldface type in interactive examples indicates typed user input.

file

Italic (slanted) type indicates variable values, placeholders, and function argument names.

[|]

{ | }

In syntax definitions, brackets indicate items that are optional and braces indicate items that are required. Vertical bars separating items inside brackets or braces indicate that you choose one item from among those listed.

...

In syntax definitions, a horizontal ellipsis indicates that the preceding item can be repeated one or more times.

cat(1)

A cross-reference to a reference page includes the appropriate section number in parentheses. For example, `cat(1)` indicates that you can find information on the `cat` command in Section 1 of the reference pages.

Return

In an example, a key name enclosed in a box indicates that you press that key.

Ctrl/*x*

This symbol indicates that you hold down the first named key while pressing the key or mouse button that follows the slash. In examples, this key combination is enclosed in a box (for example, Ctrl/C).

Overview to Network Administration

Network administration comprises those tasks that deal with setting up and configuring network interfaces, software, and daemons, and those tasks that deal with the day-to-day management of those interfaces, software, and daemons, including solving problems that might arise.

This chapter describes:

- How to use this manual in the day-to-day management of your network (Section 1.1)
- Several utilities and methods you can use to administer network components (Section 1.2)

1.1 Introduction to *Network Administration: Connections*

This manual describes the administration of the following:

- Basic network connections, including Ethernet, Token Ring, and Fiber Distributed Data Interface (FDDI) interfaces, automatic network adapter failover (NetRAIN), and network daemons (Chapter 2)
- Internet Protocol Version 6 (IPv6) (Chapter 3)
- Asynchronous Transfer Mode (ATM) (Chapter 4)
- Dynamic Host Configuration Protocol (DHCP) (Chapter 5)
- Point-to-point connections, including Serial Line Internet Protocol (SLIP) and Point-to-Point Protocol (PPP) (Chapter 6)
- Local Area Transport (LAT) (Chapter 7)

Information regarding network services and applications is maintained in a separate volume, *Network Administration: Services*.

Day-to-day management varies with each network connection, as each one provides different capabilities. Typically, management involves making small changes and adjustments, such as adding a new host to the `/etc/hosts` database, configuring a new LAT device, or obtaining status information. Chapters 2–7 of this manual describe specific tasks, presenting the generic steps required to perform the tasks followed by examples and additional information.

In addition to the day-to-day management of the network connections and transports, this manual contains information to help you solve problems that might occur. Problem solving is handled differently from administration because it is not something that you have to do every day.

Unlike the administration chapters, problem-solving chapters are structured according to specific problems. Within each problem section are the steps to resolve the problem.

The key to successful problem solving is in isolating the source of the problem. Frequently, complex networks and interactions between network services make this difficult to do. If you encounter a problem, whether by error message or event (for example, slow response), do the following:

1. Check your system, its network interface, and connections to the network.
2. Check the network and your system's ability to reach a remote system.

Most problems can be solved after you perform these two steps. If not, go to the appropriate problem-solving section and follow the steps.

1.2 Administrative Methods

The following sections provide a brief overview of the methods for administering networking components in the operating system. As explained in Section 1.2.4, it is best to not to edit configuration files manually for network configuration tasks. Instead, it is highly recommended that you use the SysMan Menu utility whenever possible.

1.2.1 SysMan Menu

The SysMan Menu utility enables you to administer your system locally via a graphical user interface or command-line interface, or even remotely via the World Wide Web. It provides a single, hierarchical menu interface that allows you to quickly find and invoke suitlets (integrated utilities) to perform the most common management tasks.

In this manual, wherever the SysMan Menu utility is mentioned in relation to configuration tasks, it is presumed that you know how to invoke it. To invoke the SysMan Menu utility from CDE, do the following:

1. Select the Application Manager icon on the CDE front panel.
2. Select the System_Admin application group icon.
3. Select the SysMan Menu. The SysMan Menu is displayed and lists various system management tasks.

If you are not using CDE, you can invoke the SysMan Menu in one of the following ways:

```
# /usr/sbin/sysman
```

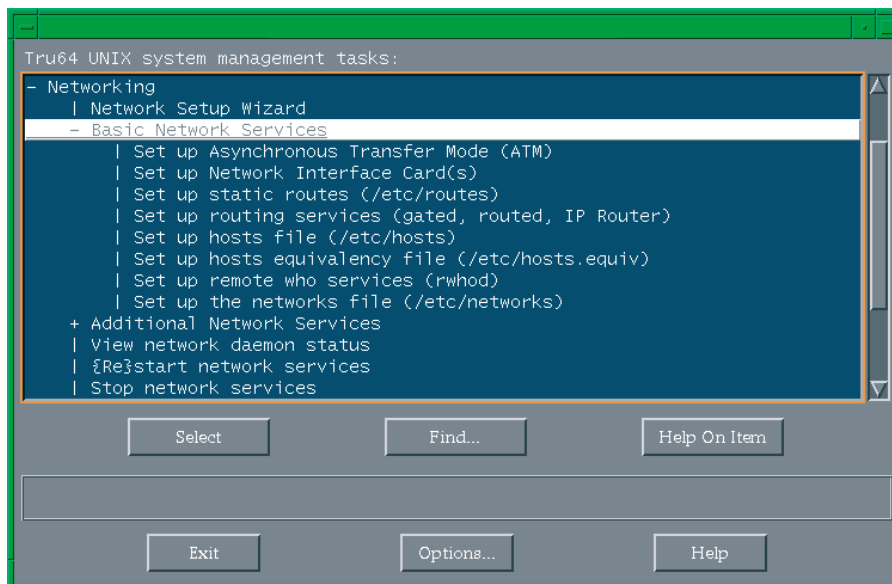
From a character-cell terminal or terminal window, for curses mode, enter:

```
# sysman -ui cui
```

After you invoke the SysMan Menu, double-click on menu items to select them. Or, on a system without graphics capabilities, use the arrow keys and the Enter key to select items. Many menu items will expand to offer more choices. Navigate the menu until you find the desired suitlet.

In Figure 1–1, the user selects the Basic Network Services menu item, which expands to reveal the suitlets for configuring network adapters and other basic networking components.

Figure 1–1: SysMan Menu



To exit the SysMan Menu, select Exit. On a system without graphics capabilities, use the Tab key to move the cursor to Exit, then press the Enter key.

For more information about the SysMan Menu, see *System Administration*, `sysman(8)`, and the online help.

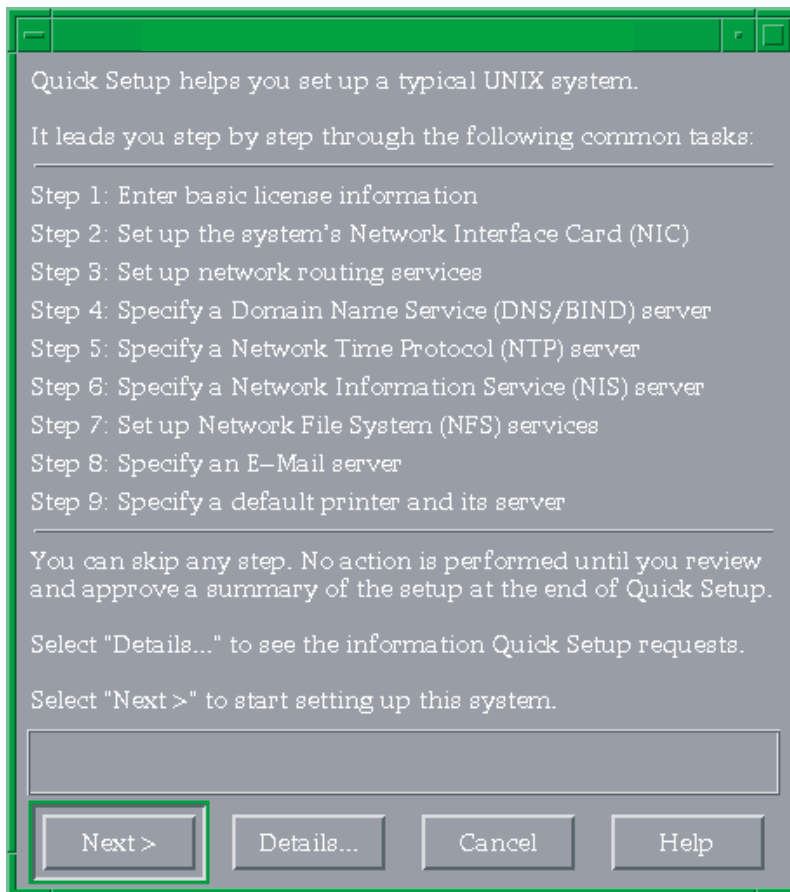
1.2.1.1 Quick Setup

The SysMan Menu includes a Quick Setup utility that you can use to configure basic components and services on a client system. The Quick Setup utility starts automatically when the system boots following a full installation of the operating system. However, to use the utility at any time, invoke the SysMan Menu and select General Tasks→Quick Setup, or enter the following command on a command line:

```
# /usr/bin/sysman quicksetup
```

The Quick Setup utility, as shown in Figure 1–2, is displayed.

Figure 1–2: Quick Setup



The utility leads you through the displayed configuration steps, many of which prepare your system for operation on a network. Enter the information for each step of the process and select Next to display the subsequent step. You can move back and forth through the steps if you

have missed something. No information is saved until you confirm the configuration by selecting Finish in the last step.

If necessary, you can configure additional components or modify your configuration after you use the utility. For more information about the Quick Setup utility, see the online help.

1.2.1.2 Network Setup Wizard

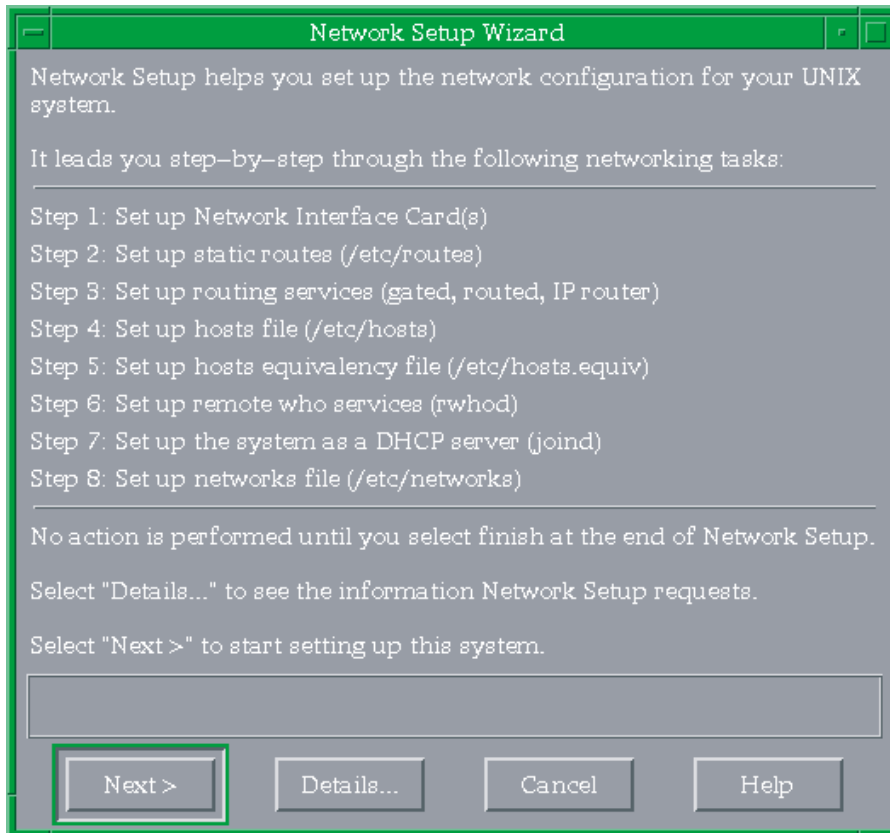
The SysMan Menu also includes a Network Setup Wizard utility that you can use to configure network components on your system. As discussed in Section 2.3, you can invoke the configuration suitlets through the SysMan Menu to configure basic network services on an individual basis, or you can use the Network Setup Wizard, which leads you step-by-step through the setup process for all of the basic network services.

To use the Network Setup Wizard, invoke the SysMan Menu and select Networking→Network Setup Wizard, or enter the following command on a command line:

```
# /usr/bin/sysman net_wizard
```

The Network Setup Wizard utility, as shown in Figure 1–3, is displayed.

Figure 1–3: Network Setup Wizard



The utility leads you through the displayed configuration steps. Enter the information for each step of the process and select Next to display the subsequent step. You can move back and forth through the steps if you have missed something. No information is saved until you confirm the configuration by selecting Finish in the last step.

If necessary, you can configure additional components or modify your configuration after you use the utility. For more information about the Network Setup Wizard utility, see the online help.

1.2.1.3 Command-Line Integration

The SysMan Menu allows you to access and manipulate many configuration options directly from the command line. This feature is particularly useful for administrators who want to create site-specific shell scripts to perform configuration tasks.

To use the command-line interface, invoke the `sysman -cli` command. For the command's arguments, specify the component and group on which you want to operate and the action you want to perform.

For example, suppose you want to list all of the entries in the `/etc/hosts` file. You would enter the following command:

```
# sysman -cli -list val -comp networkedSystems \  
-group hostMappings
```

You could also add a host to the file by entering this command:

```
# sysman -cli -add row -comp networkedSystems \  
-group hostMappings -data "{queen} \  
{DNS server} {18.240.32.40} {queen.abc.xyz.com}"
```

You can even change an existing value in the file, like an IP address, as follows:

```
# sysman -cli -set val -comp networkedSystems \  
-group hostMappings -attr networkAddress="18.240.32.45" \  
-key1 queen.abc.xyz.com -key2 18.240.32.40
```

For more information about this command line interface for the SysMan Menu, see *System Administration* and `sysman_cli(8)`.

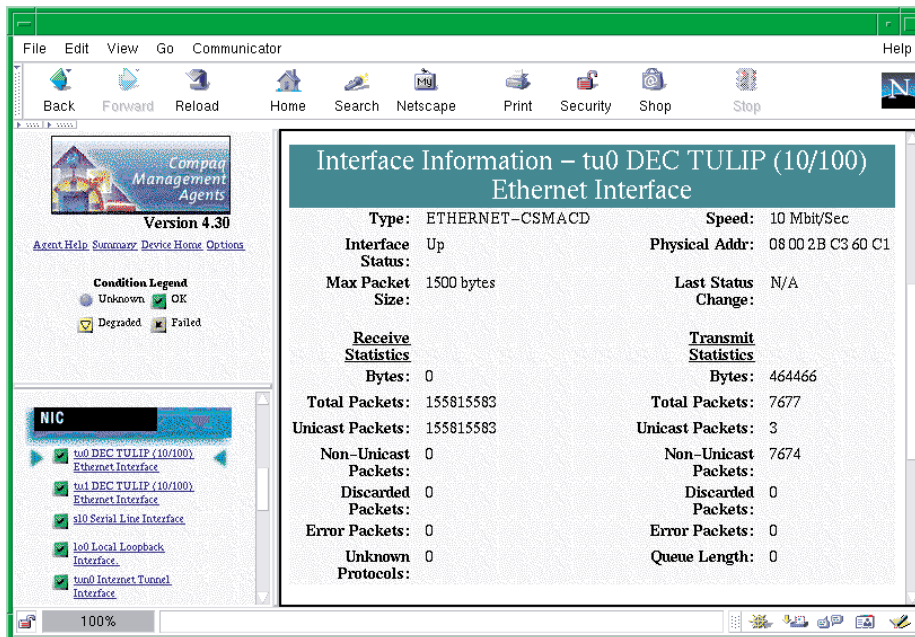
1.2.2 Compaq Insight Manager

Compaq Insight Manager is a Web-based system management utility. It consists of two different components: the Management Agents, which run on many different operating systems (including Tru64 UNIX), and the Management Console, which runs exclusively on Microsoft Windows NT.

By enabling the Compaq Management Agents on your Tru64 UNIX systems, you can provide a conduit for communication between these systems and the World Wide Web. Once enabled, this conduit allows you to access information about the configuration of your systems and their peripherals from a Web browser on any system. In some Java-enabled Web browsers, you can also invoke the SysMan Menu through this interface to manage these systems.

Figure 1-4 shows an example of using the Management Agents to obtain statistics for an Ethernet network adapter.

Figure 1–4: Compaq Management Agents



Using the Compaq Insight Manager XE Management Console, you can view and manage your systems as well as many standalone devices (such as printers, routers, and more) on your network. The Management Console is especially useful for managing heterogeneous environments, as it can communicate with the Management Agents for all of the supported operating systems and environments.

For more information about Compaq Insight Manager, see `insight_manager(5)` and *System Administration*.

1.2.3 Other Interfaces

The operating system includes alternative system administration applications, some that require graphics capabilities and others that allow you to configure your system from the command line. This manual mentions these optional utilities, when available, in relation to specific configuration tasks.

See Chapter 2 of *System Administration* for a comprehensive list of the utilities that are available. See the reference pages and online help for more information about each utility.

1.2.4 Manually Editing Configuration Files

Some sections of this manual describe the system files that are updated or modified when you perform an administrative task. Experienced UNIX administrators might prefer to administer their systems by manually editing these files, as opposed to invoking the documented utility; however, it is strongly recommended that you use the appropriate utilities to update the system files so that the structure of these files is preserved.

Important considerations are:

- Context-Dependent Symbolic Links (CDSLs)
Many system files now exist as special symbolic links (CDSLs) created to facilitate TruCluster Server clusters. The links are transparent to most users, but if the links are broken, the system cannot join a cluster in the future without recreating them. This manual mentions a few of the CDSLs, especially when you must create them manually. See the `hier(5)` reference page for a complete list of the CDSLs in the file system. See *System Administration* for more information.
- Binary databases, configuration definitions
Many system components write data to both text and binary files, and their administrative utilities often recreate the binary files. Other system information is often preserved so that when you update your system, it can be recovered and reused, saving you time and effort.
- Latent support for clusters
Individual systems are capable of joining TruCluster Server clusters, and many system files have been modified to provide latent support for clusters. For example, the `rc.config` file now has two related files, `rc.config.common` and `rc.config.site`, which can store run-time configuration variables. Altering these files with the `rcmgr` utility ensures the integrity and consistency of these files.
- Update installation
During the update installation process, changed information is merged into existing system files. The `.new.*` and `.proto.*` files might be important in this process. Refer to the *Installation Guide* for more information.

In many cases, the SysMan Menu utility is the best alternative to manually editing system files, thus it is the utility that is most frequently covered in this manual.

1.2.5 Installation and Configuration Cloning

The operating system includes two features, Installation Cloning and Configuration Cloning, that allow you to minimize the amount of manual

intervention that is necessary to install and configure systems. These features are particularly useful if you need to set up many identical systems in the same way, because they allow you to capture the configuration of a working system in configuration description files (CDFs) and use those files to install and configure subsequent systems.

See *Installation Guide — Advanced Topics* for more information.

2

Basic Network Connections

This chapter describes:

- The basic Tru64 UNIX network environment (Section 2.1)
- How to prepare for your network configuration (Section 2.2)
- How to configure the network components (Section 2.3)
- How to manage multiple network interfaces (Section 2.4)
- How to enable access filtering on a network interface (Section 2.5)
- How to display and modify FDDI parameters (Section 2.6)
- How to manage Token Ring source routing (Section 2.7)
- How to display and modify the Token Ring IP MTUsize (Section 2.8)
- How to manage network Quality of Service (QoS) (Section 2.9)

Note

This chapter discusses the configuration of network interfaces in an Internet Protocol Version 4 (IPv4) environment. All references to the Internet Protocol (IP) and the Transmission Control Protocol/Internet Protocol (TCP/IP) are IPv4-specific. For information about configuring IPv6 in a network environment, see Chapter 3.

For information about ATM and point-to-point connections, see Chapter 4 and Chapter 6, respectively.

For troubleshooting information, see Section 8.3.

2.1 Network Environment

When you install a system in a network, you need to know how to configure your network interface card (NIC) and how to route messages from your system to other systems. This section helps answer both of these questions.

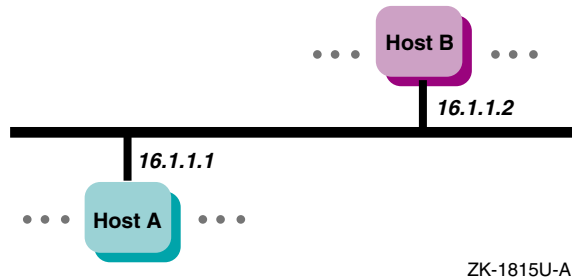
2.1.1 Network Interface

Your system is connected to the network through a NIC (which is also called a network interface or network adapter). End systems or hosts can have the following interface options:

- Single interface in a subnet
- Multiple interfaces in a subnet
- Multiple interfaces with automatic failover (NetRAIN)
- Multiple aggregated interfaces (link aggregation)

Routers typically have multiple interfaces, with each connected to a different subnet. Figure 2–1 shows a network with two hosts, Host A and Host B, each with a single network interface in a subnet.

Figure 2–1: Sample Single Interface Configuration



If you need one of the multiple interface options, Table 2–1 summarizes the characteristics of each multiple interface option to help you choose the option that is right for you.

Table 2–1: Comparison of Multiple Interface Configurations

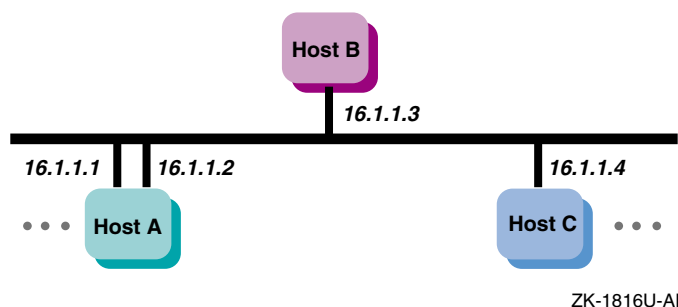
Configuration	Characteristics
Multiple interfaces in a subnet	Higher throughput, load sharing across interfaces based on connections (outbound traffic only)
NetRAIN	Reliability and availability
Link aggregation or trunking	Higher throughput, load sharing across interfaces (inbound and outbound traffic), and availability

The following sections describe each option in more detail.

2.1.1.1 Multiple Interfaces in a Subnet

You can configure multiple active network interfaces in one system, even if they operate on the same subnetwork. For example, you can configure a `tu0` interface at 16.1.1.1 and a `tu1` interface at 16.1.1.2, both with the same netmask, as shown for Host A in Figure 2–2.

Figure 2–2: Sample Multiple Interfaces in a Subnet



When you establish a connection, the kernel routes the connection through the interface that has the fewest number of connections. This connection-balancing effect can lead to greater throughput than on a system with just one network adapter per subnetwork.

This feature differs from NetRAIN because it does not give you increased reliability or failover; it simply gives a system multiple paths to access the network.

Network administrators might choose to configure a system with multiple interfaces in the same subnetwork for various reasons. For example:

- The current environment has only a single subnet, but additional bandwidth is needed to certain systems.
- The site cannot upgrade its network infrastructure to newer, faster technologies, such as Gigabit Ethernet, which would improve network throughput.
- The source of a bottleneck is a particular system's network connection, but the switch to which it is connected is underutilized and has additional ports and bandwidth available. Another connection to this system would reduce resource contention.
- There are no additional IP subnetworks assigned or available for configuration, and the host requires more bandwidth to access the current subnetwork than one network interface card allows.

For the system to function properly when configured in this manner, it must meet all of these conditions:

- It must be part of one of the following physical network layouts:
 - Switched Ethernet (10/100/Gigabit)
 - Switched Fiber Distributed Data Interface (FDDI)
 - ATM Classical IP (CLIP)
 - ATM LAN Emulation (LANE)
 - Point-to-Point (PPP)
- It must not be running a routing daemon (either `gated` or `routed`).
- It must have access to all remote systems through each interface that is configured in the same subnet. For example, you must be able to successfully issue a `ping` command to the same remote system when each network interface is configured by itself. This implies that all interfaces in the system are connected to the same physical network switch.

This feature might affect the operation of network software or commands that rely on the network interface staying constant for the life of a connection. For example:

- Multicast transmission might not work properly.
- Utilities such as `traceroute` might give inconsistent output, since the interface used might change from packet to packet.

No special settings are required to use this feature. Configure the network interfaces as directed in Section 2.3.1 and assign the interfaces IP addresses in the same subnet.

By default, configuring an interface adds interface route into the routing table. If you want to add routes using the `route` command or the `/etc/routes` file, see `route(8)` for details on adding routes on multiple interfaces. For example, you might want to add a default route on multiple interfaces. See `netstat(1)` for information on how to view the kernel routing table.

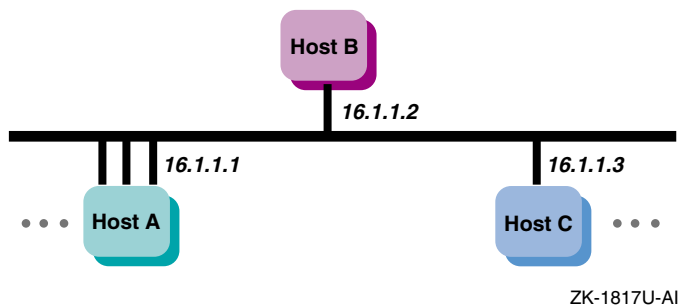
2.1.1.2 NetRAIN

The Redundant Array of Independent Network Adaptors (NetRAIN) interface provides a mechanism to protect against certain kinds of network connectivity failures.

NetRAIN integrates multiple network interfaces on the same local area network (LAN) segment into a single virtual interface called a NetRAIN set. One network interface in the set is always active while the others remain idle. If the active interface fails, one of the idle set members comes online with the same IP address within an adjustable failover time period.

Figure 2–3 shows Host A with three interfaces that are part of a NetRAIN set. The NetRAIN virtual interface is assigned the address 16.1.1.1.

Figure 2–3: Sample NetRAIN Configuration



See Section 2.4.1 for information on configuring NetRAIN.

NetRAIN monitors the status of its network interfaces with the Network Interface Failure Finder (NIFF), a tool used to detect and report possible network failures. This tool can be used independently of NetRAIN. For more information about NIFF, see `niff(7)`.

NetRAIN and MAC Address Licensing Schemes

Licensing schemes that use a network adapter's Media Access Control (MAC) address to uniquely identify a machine can be affected by how NetRAIN changes the MAC address.

All network drivers support the `SIOCRPHYSADDR` ioctl that fetches MAC addresses from the interface. This ioctl returns two addresses in an array:

- Default hardware address
The permanent address that is taken from the small PROM that each LAN adapter contains
- Current physical address
The address that the network responds to on the wire

Licensing schemes based on MAC addresses must use the default hardware address returned by the `SIOCRPHYSADDR` ioctl; do not use the current physical address because NetRAIN modifies this address for its own use. See the reference page for your network adapter (for example `ln(7)` and `tu(7)`) for a sample program that uses the `SIOCRPHYSADDR` ioctl. For more information about ioctls, see `ioctl(2)`.

2.1.1.3 Link Aggregation

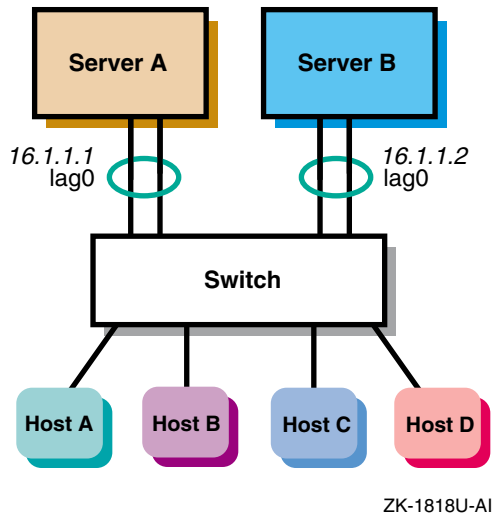
Link aggregation, or trunking, enables administrators to combine one or more physical Ethernet NICs and create a single logical link. (Upper-layer software sees this link aggregation group as a single logical interface.) The single logical link can carry traffic at higher data rates than a single interface because the traffic is distributed across all of the physical ports that make up the link aggregation group.

Using link aggregation provides the following capabilities:

- Increased network bandwidth — The increase is incremental based on the number and type of ports, or NICs, added to the link aggregation group.
- Fault tolerance — If a port in a link aggregation group fails, the software detects the failure and reroutes traffic to the other available ports. This capability is available for DEGPA (alt) and DE60x (ee) devices only.
- Load sharing — A link aggregation group performs load sharing of both inbound and outbound traffic. When transmitting packets, the system uses a load distribution algorithm to determine on which attached port to transmit the packets. The following load distribution algorithm is supported:
 - For IP packets, the port is selected based on a hash of the destination IP address. For non-IP packets, the port is selected based on a hash of the destination MAC address. All traffic addressed to a specific destination system uses the same port in the link aggregation group. This ensures that the packets arrive in order.

You can use a link aggregation group virtual interface for the following point-to-point connections: server-to-server and server-to-switch. Figure 2–4 shows Server A and Server B, each with two interfaces in a link aggregation group, attached to a switch. A single IP address is assigned to each link aggregation virtual interface.

Figure 2–4: Sample Link Aggregation Configuration



See Section 2.4.3 for information on configuring link aggregation.

2.1.2 Routing

All systems (hosts and routers) connected to a network must be configured to support network routing in order to communicate with other systems on other networks. A route is the path a packet takes through a network from one system to another. As such it enables you to communicate with other systems on other networks. Routes are stored on each system in the routing tables or routing database. Each route entry consists of the following:

- A destination address (either a network or a host)
- The address of the next hop from your system to the destination
- The address of your system on the network if the route is through an interface
- A network interface (for example, tu0 and fta0)
- Metrics (for example, hop count and MTU)

When you configure your system you automatically get a route for your loopback interface (100). In addition, you get a route for each interface that you configure by using the SysMan Configure Interfaces application. If you want additional routes, you can do one of the following:

- Create routes manually based on your map of the network. These routes are called static routes. Any time there is physical change in the network, you might have to modify the routing tables on each system. This depends on whether nodes are changing addresses or subnets.

- Run either the `gated` or `routed` daemon to have routes dynamically created, maintained, and updated. These are called dynamic routes. Any time there is physical change in the network, these daemons receive messages from other nodes or routers to modify the routing table entries automatically.

In addition to either of the previous choices, additional routes might be added to your routing tables based on Internet Control Message Protocol (ICMP) redirect messages. These are messages from routers to hosts that tell the host to forward traffic to another router on the local network. Section 2.2 presents the routing choices and information to help you make the correct choice.

2.2 Preparing for the Configuration

You configure the network components by using the Network Configuration application. The following sections contain worksheets that you can use to record the information required to configure the network components.

2.2.1 Information for Interfaces and Daemons

Figure 2–5 shows the Interface and Daemon Worksheet. The following sections explain the information you need to record on this worksheet. If you are viewing this manual online, you can use the print feature to print a copy of the worksheet.

Figure 2–5: Interface and Daemon Worksheet

Interface and Daemon Worksheet	
All Network Interfaces	
Adapter name:	_____
Host name:	_____
Internet address source:	DHCP server <input type="checkbox"/> User supplied <input type="checkbox"/>
Internet address:	_____
Network mask:	_____
Token Ring interface	
Adapter speed:	_____
NetRAIN interface	
Set members:	_____
Link Aggregation interface	
Ports:	_____
rwhod Daemon	
rwhod:	Yes <input type="checkbox"/> No <input type="checkbox"/>
Flags:	Broadcast only <input type="checkbox"/> Listen only <input type="checkbox"/> Both <input type="checkbox"/>
routed Daemon	
routed:	Yes <input type="checkbox"/> No <input type="checkbox"/>
Flags:	<input type="checkbox"/> Run routed on gateway host <input type="checkbox"/> Write all packets to standard output <input type="checkbox"/> Log additional information
RIP data:	Supply <input type="checkbox"/> Run quietly <input type="checkbox"/>
Gateways File	
Destination type:	Net <input type="checkbox"/> Host <input type="checkbox"/>
Destination:	_____
Gateway:	_____
Hop count:	_____
Route type:	External <input type="checkbox"/> Passive <input type="checkbox"/> Active <input type="checkbox"/>
gated Daemon	
gated:	Yes <input type="checkbox"/> No <input type="checkbox"/>
Configuration file:	_____
IP Router	
IP router:	Yes <input type="checkbox"/> No <input type="checkbox"/>

2.2.1.1 All Network Interfaces

Adapter name

The device names of the network interfaces. The following table contains a list of selected network interfaces that the operating system supports:

Interface	Device Name
Ethernet	ee le ln tu xna
Fiber Distributed Data Interface (FDDI)	faa fta fza
Gigabit Ethernet	alt
Token Ring	tra

Note that if you configuring a NetRAIN interface, as described in Section 2.4.1, the adapter name is the virtual device name of your NetRAIN set (`nr`). If you are configuring a link aggregation group, as described in Section 2.4.3, the adapter name is the virtual device name of your group (`lag`).

Host name

The fully qualified host name assigned to your system. A fully qualified host name contains the host name and the domain name, with host name and each level of the domain name separated by a period (.). Ask the network administrator for a unique host name.

Internet address source

The source of your system's network address for Ethernet, FDDI, and NetRAIN interfaces only. If your network uses a Dynamic Host Configuration Protocol (DHCP) server to assign IP addresses to systems at boot time, check the DHCP server box. If you plan to assign an IP address and network mask as part of system configuration, check the User supplied box.

Internet address

The IP address of your system. If you are going to supply your own IP address, write it in this space. If you will be using DHCP to assign IP addresses on a temporary basis, leave this space blank.

If you do not have a designated IP address for your network, you need to obtain one from one of the following services. Then, after you receive your network's address, assign a unique IP address and host name to each system on your network.

To obtain an Internet address for your network, contact:

American Registry for Internet Numbers
4506 Daly Drive, Suite 200
Chantilly, VA 20151

Voice: (703) 227-0660
FAX: (703) 227-0676
E-mail: reg-services@arin.net (for general information)
hostmaster@arin.net (for IP address registrations)
WWW: <http://www.arin.net>

In Europe, you can contact:

RIPE Network Coordination Center
Singel 258
1016 AB Amsterdam
The Netherlands

Voice: +31 20 535 4444
FAX: +31 20 535 4445

E-mail: ncc@ripe.net (for general information)
hostmaster@ripe.net (for IP address registrations)
WWW: <http://www.ripe.net>

In Asia and the Pacific region, you can contact:

Asia Pacific Network Information Center
Level 1, 33 Park Road
P.O. Box 2131
Milton, QLD 4064
Australia

Voice: +61 7 3367 0490
FAX: +61 7 3367 0482

E-mail: info@apnic.net (for general information)
hostmaster@apnic.net (for IP address registrations)
WWW: <http://www.apnic.net>

Note

It is a good idea to register your network even if you do not intend to connect to the Internet network. Then, if you decide to connect to the Internet network later, you will not have to change all the host addresses on your network.

Network mask

Your network's subnet mask. Subnetworks allow the systems on a LAN be known by one address to the Internet network, while being known locally by a set of addresses. Subnetworks can represent logical groupings of hosts, or different physical networks. If your network uses subnetwork routing, each system on the network must have the same subnet mask defined. Use the following table to help identify

your subnet mask. If you are not using subnetworks, the *n* is zero (0); otherwise, the *n* is greater than zero and less than or equal to 255.

Class	IP Address Range	Subnet Mask
A	0.0.0.0 to 127.0.0.0	255. <i>n.n.n</i>
B	128.0.0.0 to 191.0.0.0	255.255. <i>n.n</i>
C	192.0.0.0 to 223.0.0.0	255.255.255. <i>n</i>

If you are connecting your system to an existing network that is using subnetwork routing, ask the network administrator for the correct subnet mask.

2.2.1.2 Token Ring Interface

Adapter speed

If your system supports token ring, the speed of your system's token ring adapter. Two speeds are supported: 4Mb/s and 16Mb/s. The default speed is 16Mb/s.

2.2.1.3 NetRAIN Interface

NetRAIN interfaces provide higher availability on systems that contain multiple network adapters. See Section 2.1.1.2 for more information.

Set members

The device names of the network interfaces that are part of the NetRAIN set. When one interface in the set ceases to function, NetRAIN will fail over to another interface on this list.

2.2.1.4 LAG Interface

Link aggregation interfaces provide higher availability, fault tolerance, and load sharing on systems that contain multiple network adapters. See Section 2.1.1.3 for more information.

Ports

The device names of the network interfaces that are ports in a link aggregation group. When one interface in the group ceases to function, traffic is rerouted to the other available port or ports.

2.2.1.5 rwhod Daemon

The `rwhod` daemon maintains the database that is used by the `rwho` and `ruptime` programs. These programs provide basic information about the system and its current users to users on remote systems.

rwhod

If you want to run the `rwhod` daemon, check Yes; otherwise, check No.

Running the `rwhod` daemon allows you to use the `rwho` and `ruptime` commands.

Flags

If the `rwhod` daemon is to send `rwho` packets and ignore incoming packets, check Broadcast Only. If the daemon is to collect incoming packets, but not broadcast `rwho` packets, check Listen Only. If the daemon is to do both, check Both.

See `rwhod(8)` for additional information.

2.2.1.6 routed Daemon

The `routed` daemon allows your system's internal routing tables for the Routing Information Protocol (RIP) to be updated automatically.

routed

If you want to run the `routed` daemon, check Yes; otherwise, check No. Use the `routed` daemon to manage your routes dynamically only if your network and system requirements match the criteria in the following table:

Criterion	Type or Value
Size of network	Medium to large LAN or WAN, with multiple subnets
Network Topology	Variable
Number of routes required	Loopback, network interface route, and many others
Routers advertising routes	Yes
Configuration complexity	Low
System overhead	Low

You can choose to run the `routed` daemon or `gated` daemon, but not both. For more information about these daemons and static routing, see the *Best Practice for Network Routing* on the Tru64 UNIX Publications Home Page at the following URL:

<http://www.tru64unix.compaq.com/docs/>

Flags

Specifies how you want the `routed` daemon to run. You can run the `routed` daemon on a gateway host, write all packets to standard output, or log debugging information. Check the options you want. See `routed(8)` for more information.

RIP data

If the `routed` daemon is to supply RIP information, check `Supply`; otherwise, check `Run Quietly`.

2.2.1.7 Gateways File

The `gateways` file contains Internet routing information for the `routed` daemon. Specify the following parameters for the file:

Destination Type

If the route is to a network, check `Net`. If the route is to a specific host, check `Host`.

Destination

The destination name or IP address (in dotted-decimal format).

Gateway

The name or IP address of the gateway host to which messages will be forwarded.

Hop count

The hop count, or number of gateways, from the local network to the destination network.

Route type

If the gateway is expected to exchange RIP routing information, check `Active`. If the gateway is not expected to exchange routing information, check `Passive`. If the gateway is to notify `routed` that another routing process will install the route (it is not advertised through RIP), check `External`.

See `gateways(4)` for additional information.

2.2.1.8 gated Daemon

The `gated` daemon allows your system's internal routing tables for various routing protocols to be updated automatically.

gated

If you want to run the `gated` daemon, check Yes; otherwise, check No. Use the `gated` daemon to manage your routes dynamically only if your network and system requirements match the criteria in the following table:

Criterion	Type or Value
Size of network	Medium to large, with multiple subnets
Network Topology	Variable
Number of routes required	Loopback, network interface route, and many others
Routers advertising routes	Yes
Configuration complexity	Moderate to high
System overhead	Low
System role	Host, router, or cluster member

You can choose to run the `gated` daemon or `routed` daemon, but not both. For more information about these daemons and static routing, see the *Best Practice for Network Routing* on the Tru64 UNIX Publications Home Page at the following URL:

<http://www.tru64unix.compaq.com/docs/>

Configuration file

The name of an alternate configuration file. By default, the `gated` daemon uses the `/etc/gated.conf` file.

2.2.1.9 IP Router

An IP router is a gateway host connected to more than one TCP/IP network that receives and forwards packets between the networks.

You can configure your system as an IP router if you have more than one network interface installed and configured. In addition, you must have configured either the `routed` or the `gated` daemon.

IP router

If you want the system to run as an IP router, check Yes; otherwise, check No.

2.2.2 Information for Network Files

Figure 2–6 shows the Network Files Worksheet. The following sections explain the information you need to record on this worksheet. If you are viewing this manual online, you can use the print feature to print a copy of the worksheet.

Figure 2–6: Network Files Worksheet

Network Files Worksheet		
Static Routes File (/etc/routes)		
Destination type:	Default gateway <input type="checkbox"/>	Host <input type="checkbox"/> Network <input type="checkbox"/>
Destination:	_____	
Route via:	Gateway <input type="checkbox"/>	Interface <input type="checkbox"/>
Gateway:	_____	
Hosts File (/etc/hosts)		
Host name:	_____	_____
Internet address:	_____	_____
Alias:	_____	_____
Hosts Equivalencies File (/etc/hosts.equiv)		
Host name:	_____	_____
User name:	_____	_____
Networks File (/etc/networks)		
Network name:	_____	_____
Network address:	_____	_____
Alias:	_____	_____

2.2.2.1 Static Routes File (/etc/routes)

The `routes` file specifies static routes that will be added to your system's internal routing tables when the system boots.

Use static routes only if your network and system requirements match the criteria in the following table:

Criterion	Type or Value
Size of network	Small LAN (hosts and one gateway/router)
Network Topology	Stable
Number of routes required	Loopback, network interface route, and a few others
Routers advertising routes	No
Configuration complexity	Low
System overhead	None

For more information about static routing, as well as the `gated` and `routed` daemons, see the *Best Practice for Network Routing* on the Tru64 UNIX Publications Home Page at the following URL:

<http://www.tru64unix.compaq.com/docs/>

If you choose to use static routes, specify the following parameters for the `routes` file:

Destination type

The specific path, as stored in the `/etc/routes` file, from your system to another host or network. A static route is not updated by network software. If you want to route to a default gateway, check `Default Gateway`; to a host, check `Host`; or to a network, check `Network`.

Destination

The name or IP address of the route destination. For default gateway, the default destination is `default`.

Route via

If you are routing through a gateway, check `Gateway`. If you are routing through an interface, check `Interface`.

Gateway

The name or IP address of the gateway or interface.

See `routes(4)` for additional information.

2.2.2.2 Hosts File (/etc/hosts)

The `hosts` file contains critical address information for the known hosts on the network. Specify the following parameters for the file:

Host name

The names of other hosts on the network to be added to the `/etc/hosts` file.

If your network is running a distributed database lookup service (DNS/BIND or NIS), you do not need to list each host on your network in your `/etc/hosts` file. However, it is a good idea to list four or five systems on the network designated as DNS/BIND or NIS servers in your `/etc/hosts` file.

Internet address

The IP addresses of other hosts on the network to be added to the `/etc/hosts` file.

Alias

The aliases, if any, of other hosts on the network to be added to the `/etc/hosts` file.

See `hosts(4)` for additional information.

2.2.2.3 Hosts Equivalencies File (`/etc/hosts.equiv`)

The `hosts.equiv` file contains the names of remote systems and users that can execute commands on the local system. Specify the following parameters for the file:

Host name

The name of the trusted hosts to be put in the `/etc/hosts.equiv` file. Systems listed in the `/etc/hosts.equiv` file are logically equivalent to, and therefore treated exactly the same as, the local system.

Setting up an `/etc/hosts.equiv` file is optional but, if you choose to have one on your system, you need to create it and add the names of any trusted hosts.

User name

The name of a user on a trusted host.

See `hosts.equiv(4)` for additional information.

2.2.2.4 Networks File (`/etc/networks`)

The `networks` file contains information about the known networks that your system needs to access. Specify the following parameters for the file:

Network name

The official Internet name of the network.

Network address

The IP address of the network.

Alias

The unofficial names used for the network to be added to the `/etc/networks` file.

See `networks(4)` for additional information.

2.3 Configuring the Network Components

Use the SysMan Menu application of the Common Desktop Environment (CDE) Application Manager to configure the following network components on your system:

- Network interfaces (Ethernet, FDDI, and Token Ring)
- Remote who service (`rwhod` daemon)
- Routing services (`routed` daemon, `gated` daemon, IP router)
- Static routes file (`/etc/routes`)
- Hosts file (`/etc/hosts`)
- Host equivalent file (`/etc/hosts.equiv`)
- Networks file (`/etc/networks`)

To invoke the SysMan Menu application, follow the instructions in Section 1.2.1. See the same section for information about time-saving alternatives for configuration tasks.

2.3.1 Configuring Network Interfaces

Use the following procedure to configure Ethernet, FDDI, or Token Ring network interfaces. For information about how to configure NetRAIN, see Section 2.4.1. For information about how to configure a link aggregation group, see Section 2.4.3.

Note

If you are configuring a system that is new to this environment, verify that the network adapter mode is set correctly at the console level before continuing. For example, if you have a 10base2 Ethernet network and your system is configured to use 10baseT Ethernet, your system fails to see the network until you set the appropriate console variable. See the prerequisite tasks for a full installation in the *Installation Guide* for more information.

1. From the SysMan Menu, select Networking→Basic Network Services→Set up Network Interface Card(s) to display the Network Interface Card (NIC) dialog box.

Alternatively, enter the following command on a command line:

```
# /usr/bin/sysman interface
```

All network adapters that are installed on the system are listed in the dialog box.

2. Select the network adapter that you want to configure. The dialog box for the selected interface is displayed.
3. Enter the name for the interface in the Host Name field.
4. To configure an Ethernet interface, do the following:
 - a. To obtain the IP address data from the DHCP server, select the Use DHCP radio button. Otherwise, select the User Supplied Value radio button and enter the IP address and network mask data in the appropriate fields.
 - b. Select the Additional Flags button to display the Additional Flags dialog box, which shows advanced configuration parameters for the selected interface.
 - c. Select the check boxes and radio buttons for the other interface options that you want to enable and enter values where necessary for optional `ifconfig` arguments.
 - d. Go to step 7.
5. To configure an FDDI interface, do the following:
 - a. If you are to obtain the IP address data from the DHCP server, select the Use DHCP radio button. Otherwise, select the User Supplied Value radio button and enter the IP address and network mask data in the appropriate fields.
 - b. Select the Additional Flags button to display the Additional Flags dialog box, which shows advanced configuration parameters for the selected interface.
 - c. Select the check boxes and radio buttons for the interface options that you want to enable and enter values where necessary for optional `ifconfig` arguments.
 - d. Go to step 7.
6. To configure a Token Ring interface, do the following:
 - a. Enter the IP address for the host device in the IP Address field.
 - b. Enter the mask variable for the interface in the Network Mask field.
 - c. Select the Additional Flags button to display the Additional Flags dialog box, which shows advanced configuration parameters for the selected interface.
 - d. Select the check boxes and radio buttons for the interface options that you want to enable and enter values where necessary for

optional `ifconfig` arguments. Select the appropriate adapter speed: 4 or 16.

- e. Go to step 7.
7. Select OK to validate the parameters you entered and to close the Additional Flags dialog box. The dialog box for the adapter you are configuring is displayed.
8. Select OK to validate the configuration for network interface and close the dialog box for the adapter. The NIC dialog box is displayed.
9. Repeat steps 2 through 8, if necessary, to configure additional adapters; otherwise, select OK start network services and apply your changes now. The system applies the changes and closes the NIC dialog box.

You can also use the NIC dialog box to modify and deconfigure network interfaces. See the online help for more information.

Note

After you have configured a system to use the network for the first time, CDE becomes network-dependent, and it might function inconsistently if network services become unavailable. Therefore, if you modify or deconfigure the network interface on a system with only one interface, your system might be left in an unpredictable state. For this reason, it is best to reboot immediately after modifying the network interface to prevent problems. Furthermore, if you deconfigure the network interface, you must configure a new network interface to replace it before rebooting.

For information about monitoring and testing the connectivity of the network interfaces that you have configured, see Chapter 9.

2.3.2 Configuring the `rwhod` Daemon

To configure the `rwhod` daemon, do the following:

1. From the SysMan Menu, select Networking→Basic Network Services→Set up remote who services (`rwhod`) to display the Remote Who dialog box.

Alternatively, enter the following command on a command line:

```
# /usr/bin/sysman rwhod
```

The utility asks if you want to run the remote who service on your system.

2. Select the Yes radio button to enable the remote who service.
3. Select the appropriate rwhod flag radio button.
4. Select OK to save the changes. The utility notifies you that the changes are saved and asks if you want to apply the changes now.
5. Select Yes to apply your changes now, or select No to close the Routing Services dialog box and apply the changes the next time you reboot your system.
6. Select OK to dismiss the informational message and to close the Remote Who dialog box.

You can also use the Remote Who dialog box to disable the rwhod daemon. See the online help for more information.

2.3.3 Configuring the routed Daemon

To configure the routed daemon, do the following:

1. From the SysMan Menu, select Networking→Basic Network Services→Set up routing services (gated, routed, IP Router) to display the Routing Services dialog box.

Alternatively, enter the following command on a command line:

```
# /usr/bin/sysman routing
```

The utility displays a list of options you can use to configure the gated and routed daemons and to set up your system as an IP router.

2. Select Yes (use routed) radio button to enable the routed daemon.
3. Select the appropriate checkbox if you want to run your system as an IP router.
4. Select the appropriate check box if you want to run the routed daemon on a gateway.
5. Select the Supply RIP Data radio button if you want the routed daemon to run on a gateway host and supply Routing Information Protocol (RIP) data. Select the Run Quietly radio button if you do not want the routed daemon to supply RIP information.
6. Select the Configure Gateways button to display the Gateways dialog box. Do the following:
 - a. Select Add to add a new gateway. The Add/Modify dialog box is displayed.
 - b. In the Destination Type field, select the Network radio button if the destination is a network. Select the Specific Host radio button if the destination is a host.

- c. Enter the destination name, IP address, or “default” in the Destination field.
 - d. Enter the name or IP address of the gateway host in the Gateways field.
 - e. Enter the hop count in the Hop Count field.
 - f. Select one of the Gateway Type radio buttons.
 - g. Select OK to validate the information you entered and close the Add/Modify dialog box. Repeat steps a through g for additional gateways.
 - h. Select OK to save the changes and close the Gateways dialog box.
7. Select OK in the Routing Services dialog box to save the changes. The utility displays a dialog box to confirm the changes and to ask if you want to start the daemon now.
 8. Select Yes to start the daemon and apply your changes now, or select No to close the Routing Services dialog box and apply the changes the next time you reboot your system.

If you choose Yes, you are informed that the daemon is running. Select OK to dismiss the message and to close the Routing Services dialog box.

You can also use the Routing Services dialog box to disable the routed daemon. See the online help for more information.

See `routed(8)` and `gateways(4)` for more information about the routed daemon and the `gateways` file.

2.3.4 Configuring the gated Daemon

To configure the `gated` daemon, do the following:

1. From the SysMan Menu, select Networking→Basic Network Services→Set up routing services (`gated`, `routed`, IP Router) to display the Routing Services dialog box.

Alternatively, enter the following command on a command line:

```
# /usr/bin/sysman routing
```

The utility displays a list of options you can use to configure the `gated` and `routed` daemons and to set up your system as an IP router.

2. Select the Yes (use `gated`) radio button to enable the `gated` daemon.
3. Select the appropriate check box if you want to run your system as an IP router.
4. Enter the file name of the `gated` configuration file in the Configuration File field.

Note

To configure the `gated` daemon, you must set up the `/etc/gated.conf` file in the format specified in `gated.conf(4)`. A default `/etc/gated.conf` file is provided when you install the software.

5. Select OK in the Routing Services dialog box to save the changes. A dialog box is displayed to confirm the changes and to ask if you want to start the daemon now.
6. Select Yes to start the daemon and apply your changes now, or select No to close the Routing Services dialog box and apply the changes the next time you reboot your system.

If you choose Yes, you are informed that the daemon is running. Select OK to dismiss the message and to close the Routing Services dialog box.

You can also use the Routing Services dialog box to disable the `gated` daemon. See the online help for more information.

See `gated(8)` and `gated.conf(4)` for more information about the `gated` daemon and the `gated.conf` file.

2.3.5 Configuring the System as an IP Router

In order to function as an IP router, your system must have two network interfaces installed and configured and must have the `routed` or `gated` daemon configured. To configure the system as an IP router, do the following:

1. From the SysMan Menu, select Networking→Basic Network Services→Set up routing services (`gated`, `routed`, IP Router) to display the Routing Services dialog box.

Alternatively, enter the following command on a command line:

```
# /usr/bin/sysman routing
```

The utility displays a list of options you can use to configure the `gated` and `routed` daemons and to set up your system as an IP router.

2. Select the appropriate check box to run your system as an IP router.
3. Select OK to save the changes. A dialog box is displayed to confirm the changes and to ask if you want to start or restart the `routed` or `gated` daemon.
4. Select Yes to start the daemon and apply your changes now, or select No to close the Routing Services dialog box and apply the changes the next time you reboot your system.

If you choose Yes, you are informed that the daemon is running. Select OK to dismiss the message and to close the Routing Services dialog box.

You can also use the Routing Services dialog box to deconfigure the system as an IP router. See the online help for more information.

2.3.6 Configuring the Static Routes File

To configure the `routes` file, you add entries (static routes) to the `routes` file. Do the following:

1. From the SysMan Menu, select Networking→Basic Network Services→Set up static routes (`/etc/routes`) to display the Static Routes dialog box.

Alternatively, enter the following command on a command line:

```
# /usr/bin/sysman route
```

2. Select Add to add a static route. The Add/Modify dialog box is displayed.
3. Select one of the Destination Type radio buttons.
4. For host and net destinations:
 - a. Enter the full name or IP address of the destination network or host in the Destination field.
 - b. Select one of the Route Via radio buttons. Select the Gateway button if the route is through a gateway. Select the Interface button and skip to step 6 if the route is through an interface.
5. For a gateway, enter the full name or IP address of the gateway host to which messages will be forwarded in the Gateway field.
6. Select OK to validate the entry and add it to the list. Repeat steps 2 through 6 for additional static routes.
7. Select OK to save the current changes. A dialog box is displayed to confirm the changes and to ask if you want to start the static routes service.
8. Select Yes to start the service and apply your changes now. Or, select No to close the Static Routes dialog box and apply the changes the next time you reboot your system.

If you choose Yes, select OK to close the Static Routes dialog box.

You can also use the Static Routes dialog box to modify and delete entries in the `routes` file. See the online help for more information.

See `routes(4)` for more information about the `routes` file.

2.3.7 Configuring the hosts File

To configure the `hosts` file, do the following:

1. From the SysMan Menu, select Networking→Basic Network Services→Set up hosts file (`/etc/hosts`) to display the Hosts dialog box.

Alternatively, enter the following command on a command line:

```
# /usr/bin/sysman host
```

2. Select Add to add a host. The Add/Modify dialog box is displayed.
3. Enter an official host name in the Host Name field.
4. Enter the IP address of the new host in the Host Address field.
5. Optionally, enter any unofficial name or names for this host in the Aliases field. Also, provide pertinent information, for example, the location of the host, in the Comment field.
6. Select OK to validate the entry and add it to the list. Repeat steps 2 through 6 for additional hosts.
7. Select OK to update the `/etc/hosts` file and to close the Hosts dialog box.

You can also use the Hosts dialog box to modify and delete entries in the `hosts` file. See the online help for more information.

See `hosts(4)` for more information about the `hosts` file.

2.3.8 Configuring the hosts.equiv File

To configure the `hosts.equiv` file, do the following:

1. From the SysMan Menu, select Networking→Basic Network Services→Set up host equivalency file (`/etc/hosts.equiv`) to display the Hosts Equivalency dialog box.

Alternatively, enter the following command on a command line:

```
# /usr/bin/sysman hosteq
```

2. Select Add to add a host. The Add/Modify dialog box is displayed.
3. Enter the remote host name in the Host field.

Note

If the host is not on the network, you cannot add the host.

4. Enter the name of a user on the remote host in the User field.

5. Select OK to validate the entry and add it to the list. Repeat steps 2 through 5 for additional remote hosts.
6. Select OK to update the `/etc/hosts.equiv` file and to close the Hosts Equivalency dialog box.

The Hosts Equivalency dialog box also enables you to modify and delete entries in the `hosts.equiv` file. See the online help for additional information.

See `hosts.equiv(4)` for more information about the `hosts.equiv` file.

2.3.9 Configuring the networks File

To configure the `networks` file, do the following:

1. From the SysMan Menu, select Networking→Basic Network Services→Set up the networks file (`/etc/networks`) to display the Networks dialog box.

Alternatively, enter the following command on a command line:

```
# /usr/bin/sysman networks
```

2. Select Add to add a network. The Add/Modify dialog box is displayed.
3. Enter the official network name in the Network Name field.
4. Enter the IP address of the network in the Network Address field.
5. If an unofficial name (alias) is assigned to the new network, enter the aliases in the Aliases field.
6. Select OK to validate the entry and add it to the list. Repeat steps 2 through 6 for additional networks.
7. Select OK to update the `/etc/networks` file and to close the Networks dialog box.

You can also use the Networks dialog box to modify and delete entries in the `networks` file. See the online help for more information.

See `networks(4)` for more information about the `networks` file.

2.3.10 Configuring IP Aliases

An IP alias is an additional network address for an interface. The alias is usually an address in the same subnet as the primary IP address on the interface.

To configure an IP alias, you need the following information:

- IP alias address

- Netmask value associated with the IP alias address
- Host name associated with the IP alias address

To configure an IP alias, do the following:

1. Add the IP address and host name to the `/etc/hosts` file (see Section 2.3.7).
2. Edit the `/etc/inet.local` file and add the command to configure the alias. Use the following syntax:

```
ifconfig interface alias IP_alias_address netmask IP_alias_netmask
```

For example:

```
ifconfig tu0 alias 18.54.76.129 netmask 255.255.255.0
```

See `ifconfig(8)` for more information on `ifconfig` parameters.

3. Restart network services by entering the following command:

```
# rcinet restart
```

2.4 Managing Multiple Network Interfaces

This section describes how to perform the following tasks on systems that contain multiple network interfaces:

- Configure NetRAIN
- Monitor NetRAIN
- Configure a link aggregation group

2.4.1 Configuring NetRAIN

Before you set up the NetRAIN virtual interface, note the following hardware restrictions and configuration tips:

- You must construct a NetRAIN set out of interfaces that are currently idle. This means the interfaces cannot be marked as "up" in the Set up Network Interface Card(s) dialog box of the SysMan Menu and they cannot have IP addresses assigned to them.
- You must use two or more of the same type of network interface (FDDI, ATM LAN Emulation, or Ethernet) dedicated to a single LAN segment. If you use Ethernet adaptors, they must all be of the same speed.
- You cannot run LAT over a NetRAIN virtual interface (`nr`) or any of the interfaces that compose a NetRAIN set.
- Run separate cables from each network interface to the appropriate hub or concentrator to provide physically redundant paths back to the

network. This reduces the chance of network failure due to cables being accidentally unplugged.

- If necessary, you can adjust the timeout values to ensure that NetRAIN will successfully detect and respond to network failure. You can tune these parameters with the `sysconfig` command, `ifconfig` command, and the `ioctl` system call. See `nr(7)`, `ifconfig(8)`, `sysconfig(8)`, `dxkerneltuner(8)`, and `sys_attrs_netrain(5)` for more information.

By default, these parameters are tuned for operation over Ethernet, but it is possible that the default values and other suggested timeout values will not work in your environment. For example, if you are connected to a switch, failover time will depend on the switch and its configuration.

- You must use UNI Version 3.1 when running NetRAIN over LANE to obtain acceptable failover times with some ATM switches, including the Gigaswitch. If you use UNI Version 3.0, the failover time might be long because the T309 timer is set to 90 seconds by default on some switches. If the T309 timer is adjustable on your switch, you can set the T309 timer to 10 seconds as in UNI Version 3.1 to try to achieve acceptable failover times.

NetRAIN configuration parameters are stored in the `/etc/rc.config` file along with the parameters for other network interfaces. Use the `rcmgr` utility to change the values of the variables. For more information about the `rcmgr` utility, see `rcmgr(8)`.

Note

The NetRAIN parameters in the following steps are case sensitive and must be typed in uppercase as shown.

To configure NetRAIN, do the following:

1. Log in as root.
2. Construct the NetRAIN set or sets, as follows:
 - a. Set the NetRAIN interface name or names:

```
# rcmgr set NRDEV_n netrain-interface-id
```

The `netrain-interface-id` must have the form `nrn`.

Specify the same integer `n` for the `NRDEV_n` variable and the `nrn` interface. For example, if no NetRAIN interfaces are configured on your system, you can specify `NRDEV_0` and `nr0`, respectively.

- b. Indicate which network interfaces will be part of the NetRAIN set or sets and, if necessary, provide failover timeout values:

```
# rcmgr set NRCONFIG_n interface-id,interface-id [nrtimers integer,integer]
```

Note

When specifying the interfaces, do not leave any spaces between the *interface-id* parameters and the commas. For example, for two Ethernet interfaces, you can specify `tu0,tu1` but not `tu0, tu1`.

The `nrtimers` values dictate how long the system is to wait before switching between interfaces. For more information about `nrtimers` values, see `ifconfig(8)`.

- c. Indicate to the system that you have configured a NetRAIN set:

```
# rcmgr set NR_DEVICES integer
```

Increment *integer* by the number of NetRAIN sets you have created. For example, if you create one NetRAIN set, *integer* is 1.

3. Configure the network parameters for the NetRAIN set or sets that you created, as follows:

- a. Set the interface name:

```
# rcmgr set NETDEV_n netrain-interface-id
```

For *netrain-interface-id*, use the same `nrn` ID you specified in step 2a.

If you configured other network interfaces in the `rc.config` file, you need to find and use the next available `NETDEV_n` variable. For example, if you used `NETDEV_0` to configure an Ethernet card that is not part of the NetRAIN set, the next available variable is `NETDEV_1`.

- b. Set the `ifconfig` parameters that will be used to initialize the NetRAIN interface:

```
# rcmgr set IFCONFIG_n IP-address netmask network-mask
```

As in step 3a, if you configured other network interfaces in the `rc.config` file, you need to use the next available `IFCONFIG_n` variable.

- c. Indicate to the system that you have configured an additional network interface:

```
# rcmgr set NUM_NETCONFIG integer
```

Increment *integer* by the number of NetRAIN interfaces you have created. If you configured other network interfaces in the `rc.config` file, you need to add the number of NetRAIN interfaces to the current `NUM_NETCONFIG` value from that file.

4. Restart network services to apply the changes.

After you configure a NetRAIN set, the NetRAIN interface is available each time you restart your system.

Optionally, you can configure NetRAIN interfaces from the command line by using the `ifconfig` command, but the changes are not preserved when you reboot. For more information, see `ifconfig(8)`.

Example 2–1 and Example 2–2 show the commands you would enter to establish two different NetRAIN configurations.

To create one NetRAIN set with two Ethernet interfaces, `tu0` and `tu1`, on a system where no other network interfaces have been configured, you would enter the commands in Example 2–1.

Example 2–1: Creating One NetRAIN Set

```
# rcmgr set NRDEV_0 nr0 [1]
# rcmgr set NRCONFIG_0 tu0,tu1 [2]
# rcmgr set NR_DEVICES 1 [3]
# rcmgr set NETDEV_0 nr0 [4]
# rcmgr set IFCONFIG_0 18.240.32.40 netmask 255.255.255.0 [5]
# rcmgr set NUM_NETCONFIG 1 [6]
```

- [1] Creates a NetRAIN set called `nr0`.
- [2] Indicates that the `nr0` set consists of the `tu0` and `tu1` interfaces. Both interfaces must be marked “down” prior to this command.
- [3] Indicates to the system that there is one NetRAIN set.
- [4] Creates a network interface called `nr0` for the NetRAIN virtual interface.
- [5] Defines the IP address and network mask for the NetRAIN virtual interface.
- [6] Indicates to the system that there is one network interface.

To create two NetRAIN sets, one with two FDDI interfaces called `fta0` and `fta1` and the other with two ATM LANE interfaces called `elan0` and `elan1`, on a system where one other network interface has been configured (suppose `NETDEV_0` is `tu0`), you would enter the commands in Example 2–2.

Example 2–2: Creating Two NetRAIN Sets

```
# rcmgr set NRDEV_0 nr0 [1]
# rcmgr set NRDEV_1 nr1
# rcmgr set NRCONFIG_0 fta0,fta1 [2]
# rcmgr set NRCONFIG_1 elan0,elan1 nrtimers 4,16 [3]
# rcmgr set NR_DEVICES 2 [4]
# rcmgr set NETDEV_1 nr1 [5]
# rcmgr set NETDEV_2 nr2
# rcmgr set IFCONFIG_1 18.240.31.40 netmask 255.255.255.0 [6]
# rcmgr set IFCONFIG_2 18.240.31.42 netmask 255.255.255.0
```

Example 2–2: Creating Two NetRAIN Sets (cont.)

```
# rcmgr set NUM_NETCONFIG 3 7
```

- 1 Creates two NetRAIN sets called `nr0` and `nr1`.
- 2 Indicates that the `nr0` set consists of the `tu0` and `tu1` interfaces. Both interfaces must be marked “down” prior to issuing this command.
- 3 Indicates that the `nr1` set consists of the `elan0` and `elan1` interfaces. Both interfaces are currently idle. Also provides `nrtimers` failover values for the set. The values in this example are suggested starting values for ATM LANE. They might not work for your configuration, as described at the beginning of this section. For more information about `nrtimers` values, see `ifconfig(8)`.
- 4 Indicates to the system that there are two NetRAIN sets.
- 5 Creates network interfaces called `nr0` and `nr1` for the two NetRAIN virtual interfaces.
- 6 Defines the IP address and network mask for each NetRAIN virtual interface.
- 7 Indicates to the system that there are three network interfaces, the two NetRAIN virtual interfaces and the preexisting Ethernet interface.

2.4.2 Monitoring NetRAIN Activity

To check which member of a NetRAIN set is the active interface, use the `ifconfig` command. For example:

```
# ifconfig nr0
nr0: flags=8c63      NetRAIN Attached Interfaces: ( fta0 fta1 ) Active Interface:
( fta0 )inet 18.240.32.40 netmask ffffffff broadcast 18.240.32.255 ipmtu 4352
```

This example shows that:

- The virtual interface `nr0` is running; its IP address is 18.240.32.40.
- The NetRAIN set consists of two physical interfaces, `fta0` and `fta1`.
- NetRAIN is using `fta0` for communication. If NetRAIN determines that `fta0` is not active, it switches to the next interface in the set, `fta1`.

To see the status of all set members while the NetRAIN interface is running, use the `niffconfig` command. For example:

```
# niffconfig -u
Interface:  tu1, state: DEAD, t1: 4, dt: 2, t2: 10, time to dead: 0,
current_interval: 2, next time: 2
Interface:  nr0, state: GREEN, t1: 4, dt: 2, t2: 10, time to dead: 0,
current_interval: 4, next time: 4
Interface:  tu0, state: GREEN, t1: 4, dt: 2, t2: 10, time to dead: 0,
```

```
current_interval: 4, next time: 4
```

In this example, you can see that the virtual interface `nr0` is running and NetRAIN is using `tu0` for communication. This example also shows the `nrtimers` values for each member of the set. See `ifconfig(8)` for more information on these values.

For more information about monitoring the connectivity of network interfaces, see Section 9.1.

2.4.3 Configuring a Link Aggregation Group

Before configuring a link aggregation group, verify that the link aggregation kernel subsystem (`lag.mod`) is configured in the kernel, by issuing the following command:

```
# sysconfig -q lag
```

If the `lag:` subsystem attributes are not displayed, do the following:

1. Edit the system configuration file and add the following entry to it:

```
options LAG
```

The default configuration file is `/sys/conf/SYSTEM_NAME`, where `SYSTEM_NAME` is the name of your host processor, in uppercase letters.

2. Build a new kernel by issuing the `doconfig -c` command. If you are unfamiliar with rebuilding the kernel, see *System Administration*.
3. Reboot the system. Make sure that there are no other users on the system. Use a command similar to the following:

```
# shutdown -r +5 "Adding Link Aggregation software option ..."
```

You are now ready to configure a link aggregation group. Before you set up the link aggregation virtual interface, note the following hardware restrictions and configuration tips:

- You must construct a link aggregation group out of interfaces that are currently idle. This means the interfaces cannot be marked as "up" in the Set up Network Interface Card(s) dialog box of the SysMan Menu and they cannot have IP addresses assigned to them.
- You must use two or more of the same type of network interface (Ethernet) dedicated to a single server or switch. The interfaces must all be of the same speed and operate in full duplex mode.
- The server or switch to which you are connected must also be configured for link aggregation.
- You cannot run LAT over a link aggregation virtual interface (`lag`) or any of the interfaces that compose a link aggregation group.

- Failover is supported on DEGPA (alt) and DE60x (ee) devices only. In addition, you cannot modify the failover time.

To configure a link aggregation group, do the following:

1. Log in as root.
2. Edit the `/etc/inet.local` file.
3. Enter a `lagconfig -c` statement to create a link aggregation group.
4. Enter a `lagconfig -p` statement to enable one port (physical interface) for link aggregation. To enable additional ports, enter additional `lagconfig -p` statements.
5. Enter an `ifconfig` statement to assign an IP address to the link aggregation group virtual interface and enable it.
6. Save the changes and close the file.
7. Restart network services by entering the following command:

```
# rcinet restart
```

After you configure a link aggregation group, it is available each time you restart your system.

Optionally, you can configure a link aggregation group from the command line by using the `lagconfig` and `ifconfig` commands. However, the changes do not persist across system reboots. For more information, see `lagconfig(8)` and `ifconfig(8)`.

Example 2–3 shows the statements you would add to the `/etc/inet.local` file to create a link aggregation group made up of three ports or interfaces.

Example 2–3: Sample Link Aggregation Statements

```
# lagconfig -c 1
# lagconfig -p tu0 key=1 2
# lagconfig -p tu1 key=1 3
# lagconfig -p tu2 key=1 4
# ifconfig lag0 16.1.2.3 netmask 255.255.255.0 up 5
```

- 1 Creates a link aggregation group with a default key value and the next available interface number. Since no link aggregation group is configured on the system, this creates a group with a key value of 1 and an interface number of 0 (`lag0`).
- 2 Enables `tu0` for link aggregation. The interface must be marked “down” prior to issuing this command.

- ❸ Enables `tu1` for link aggregation. The interface must be marked “down” prior to issuing this command.
- ❹ Enables `tu2` for link aggregation. The interface must be marked “down” prior to issuing this command.
- ❺ Sets the IP address of the link aggregation virtual interface to 16.1.2.3. The enabled ports then attach to the link aggregation group that has the same key assigned to it, and are available to carry traffic.

2.5 Enabling Access Filtering on an Interface

Interface access filtering helps you detect and prevent IP spoofing attacks. To enable interface access filtering on an interface, do the following:

1. Create an `/etc/ifaccess.conf` file and add entries against which the source address of input packets are checked.
2. Use the `ifconfig` command with the `+filter` parameter to enable access filtering on the network interface.

See `ifaccess.conf(4)` and `ifconfig(8)` for more information.

2.6 Displaying and Modifying the FDDI Parameters

You use the `fddi_config` command to display and modify the FDDI adapter parameters.

To display the FDDI adapter parameters, use the `fddi_config` command with the following syntax:

```
fddi_config -i interface_name -d
```

To modify the FDDI adapter parameters, log in as root and use the `fddi_config` command with one or more of the options in Table 2–2.

Table 2–2: Options to the `fddi_config` Command

Option	Function
<code>-i interface_name</code>	Changes or displays the FDDI characteristics for <i>interface_name</i> . You must provide the interface name.
<code>-c counter_update_interval</code>	Determines how often the driver counters are updated by the DEFTA adapter. The default is 1 second. Setting the interval time to zero (0) disables counter updates. (For the DEFTA (fta) FDDI interface only.)
<code>-d</code>	Displays the FDDI interface parameters you can set.

Table 2–2: Options to the `fddi_config` Command (cont.)

Option	Function
<code>-l lem_threshold</code>	Sets the error rate threshold of Link Error Monitor (LEM). The LEM error rate threshold is 1×10^{-n} , where n ranges from 5 to 8, inclusively. The default LEM threshold is 1×10^{-8} .
<code>-p [1 0]</code>	Sets the ring purger state for the specified FDDI interface. A value of 1 enables the ring purger ability; a value of 0 disables it.
<code>-r restricted_token_timeout</code>	Sets the Restricted Token Timeout parameter, defining how long a single restricted mode dialog can last before being terminated. The range for this parameter is from 0 to 10000 milliseconds. The default value is 1000 milliseconds.
<code>-t token_request_time</code>	Sets the Request Token Rotation Time (T_{req}) for <i>interface_name</i> . T_{req} is used during the ring initialization process to negotiate a Target Token Rotation Time (TTRT) for the ring. The range for this parameter is from 4.0 milliseconds to 167.77208 milliseconds. The default value is 8.0 milliseconds.
<code>-v valid_transmit_time</code>	Sets the Valid Transmission Time (TVX) timer for a specific FDDI interface. The range for the TVX timer is from 2.35 milliseconds to 5.2224 milliseconds. The default is 2.6214 milliseconds.
<code>-x [1 0]</code>	Enables (1) or disables (0) full-duplex operation for the interface. If the full-duplex operation is enabled, the interface is in one of the following states: Idle, Request, Confirm, or Operational. (For the DEFTA (fta) FDDI interface only.)

See `fddi_config(8)` for more information on this command and its options.

The following example shows how to display the FDDI interface parameters you can set:

```
% /usr/sbin/fddi_config -i fza0 -d
fza0 ANSI FDDI settable parameters

Token Request Time:          0.0000 ms
Valid Transmission Time:     0.0000 ms
LEM Threshold:               0
Restricted Token Timeout:    15.8314 ms
Ring Purger State:          (null)
```



```
fza0 Full Duplex Mode: Disabled
```

```
fza0 Counter Update Interval: 10 sec
```

The following example shows how to change the Token Request Time (TRT) value for the fza0 interface to 10.2:

```
# fddi_config -t10.2 -i fza0
```

The following example shows how to turn the ring purger off:

```
# fddi_config -p 0 -i mfa0
```

2.7 Managing Token Ring Source Routing

Source routing is a bridging mechanism that systems on a token ring LAN use to send messages to a system on another interconnected token ring LAN. Under this mechanism, the system that is the source of a message uses a route discovery process to determine the optimum route over token ring LANs and bridges to a destination system. The source system stores the optimum routes in its source routing table.

When the system is booted with the DETRA adapter installed and configured, token ring source routing is initialized by default. To manage token ring source routing, use the `srconfig` command.

Table 2-3 shows the `srconfig` command options. All `srconfig` command options are case insensitive; type them in uppercase, lowercase, or mixed case. The short form for each flag is indicated by uppercase letters.

Table 2-3: Options to the `srconfig` Command

Option	Function
-DElentry <i>mac_address</i> ^a	Deletes a source routing table entry.
-DISEntry <i>mac_address</i> ^a	Disables a source routing table entry. This marks the entry as Stale.
-RAttr	Displays the source routing attributes.
-RCounter	Displays the source routing counters.
-REntry <i>mac_address</i>	Displays a specific source routing table entry.
-RTable	Displays the source routing table.
-SETAgetimer <i>timer</i> ^a	Sets the value of the Source Routing Aging Timer, specifying the length of time a source routing table entry remains valid until being marked as invalid or Stale. If not set, the system default is 120 seconds.

Table 2–3: Options to the srconfig Command (cont.)

Option	Function
<code>-SETDsctimer timer^a</code>	Sets the Source Routing Discovery Timer, specifying the amount of time a route discovery process can take before it terminates. If not set, the system default is 5 seconds.
<code>-SETMaxentry value^a</code>	Sets the maximum number of entries allowed in the source routing table. The range for this entry is a multiple of 256 from 1024 to 2048. This parameter can be increased, but not decreased. If not set, the system default is 1024.
<code>-u</code>	Specifies that the MAC addresses are in uncanonical form. This option can be used with the <code>-DElEntry mac_address</code> , <code>-DIEntry mac_address</code> , and <code>-RTable</code> options only.
<code>-Zcounter</code>	Sets the source routing counters to zero.

^a Requires superuser privileges.

See `srconfig(8)` for more information on this command and its options.

The following example increases the number of routing table entries from 1024 to 1280 by using the shortened form of the `-SetMaxEntry` option:

```
# srconfig -setm 1280
Current SR Table size is : 1024
New SR Table size is : 1280
```

The following example displays the source routing attributes by using the shortened form of the `-RAttr` option:

```
# srconfig -ra
Source Routing is enabled
Current SR Aging Timer      : 120
Current SR Discovery Timer  : 10
Current SR Table size is   : 1024
```

The following example displays the source routing counters by using the shortened form of the `-RCounter` option:

```
# srconfig -rc
ARE Frames Sent           : 00000001
ARE Frames received      : 00000000
Route Discovery Failures  : 00000001
```

The following example displays all entries, with MAC addresses in canonical form, in the source routing table, by using the shortened form of the `-RTable` option. The backslash (`\`) character indicates line continuation and does not appear in the actual output.

```
# srconfig -rt
```

```

Target Node MAC Address 00-00-0C-01-08-E9 (ip = 130.180.4.3) \
Have Route 1
Routing Information: SRF, length 8, direction 0,largest frame \
4472 octets 2
Route Descriptors: 021C 7FFC 0220 0000 0000 0000 0000 0000 3
Target Node MAC Address 00-00-C9-10-1B-F5 On Ring 4
Target Node MAC Address 08-00-2B-2C-F1-F9 (ip = 130.180.4.2) \
Stale (Have Route) 5
Routing Information: SRF, length 8, direction 0,largest frame 4472 octets
Route Descriptors: 021C 7FFC 0220 0000 0000 0000 0000 0000
Target Node MAC Address 00-00-C9-0B-33-80 Stale (On Ring)

```

- 1 Have Route indicates the source system has a valid path to the destination system.
- 2 Information returned by the destination system in response to the route discovery process.
- 3 The LAN segments and bridges that constitute the path to the destination system.
- 4 On Ring indicates the destination system is on the same ring as the source system and does not need source routing.
- 5 Stale indicates the entry is invalid and needs to be updated by the route discovery process.

The following example shows all entries, with MAC addresses in noncanonical form, in the source routing table by using the shortened form of the `-rTable` option. The backslash (`\`) character indicates line continuation and does not appear in the actual output.

```

# srconfig -rt -u
Target Node MAC Address 00:00:30:80:10:97 (ip = 130.180.4.3) Have Route
Routing Information: SRF, length 8, direction 0,largest frame 4472 octets
Route Descriptors: 021C 7FFC 0220 0000 0000 0000 0000 0000

Target Node MAC Address 00:00:93:08:D8:AF On Ring

Target Node MAC Address 10:00:D4:34:8F:9F (ip = 130.180.4.2) Stale \
(Have Route)
Routing Information: SRF, length 8, direction 0,largest frame 4472 octets
Route Descriptors: 021C 7FFC 0220 0000 0000 0000 0000 0000

Target Node MAC Address 00:00:93:D0:CC:01 Stale (On Ring)

```

2.8 Displaying and Modifying the Token Ring IP MTU Size

By default, the DETRA adapter uses an IP maximum transfer unit (MTU) size of 4092 bytes. In a multivendor environment with different adapters using different IP MTU sizes, the bridges connecting different networks can be set up to forward smaller packet sizes. As a result, bridges might drop packets or remote hosts might reject packets. If either occurs on your

network, reduce the IP MTU size for all hosts on the network and ensure that all hosts use the same size.

The following command displays the DETRA interface IP MTU size as 4092 bytes:

```
% ifconfig tra0
tra0: flags=9863<UP,BROADCAST,NOTRAILERS,RUNNING>
    inet 16.141.208.3 netmask ffffffff broadcast 16.141.208.255 ipmtu 4092
```

The following example sets the IP MTU size of DETRA interface to 2044 bytes:

```
% ifconfig tra0 ipmtu 2044
```

2.9 Managing Network Quality of Service

As applications place increasing demands for bandwidth on the Internet network, increasing the network bandwidth is only a temporary solution. Newer real-time applications demand both increased bandwidth and low latency. Clearly, the importance of bandwidth management is increasing.

An IP network with its Best Effort delivery service performs a form of passive bandwidth management. If an outgoing queue is full, indicating high network traffic and congestion, the packets are quietly dropped. Some upper-level protocols can detect data loss, others cannot.

Quality of service (QoS) is the phrase commonly associated with the concept of actively managing network bandwidth. In this scenario, all network elements (for example, hosts, applications, and routers) and all network protocol layers cooperate to ensure consistent traffic and service end-to-end in a network. Network bandwidth for real-time applications is reserved, while sufficient bandwidth remains for best-effort traffic.

The major network QoS components in this operating system are as follows:

- Traffic Control subsystem — Provides an application data flow with a QoS that approximates Best Effort delivery through unloaded network interfaces.

Traffic control is supported on the Ethernet and FDDI interfaces.

- Resource ReSerVation Protocol (RSVP) — Provides a mechanism to reserve bandwidth on the local system and through the network. On this operating system, RSVP is implemented in the form of the `rsvpd` daemon. The `rsvpd` daemon uses the Traffic Control subsystem to install and modify flows and filters for a specific network interface.
- RSVP Application Programming Interface (RAPI) — Enables a local application that requires enhanced QoS to communicate with the `rsvpd` daemon. Using the RAPI routines, an application can make resource (bandwidth) reservations on the local system or advertise services to

other nodes in the network, or both. See the *Network Programmer's Guide* for a description of the RAPI routines.

2.9.1 Managing the Traffic Control Subsystem

The Traffic Control subsystem performs the following tasks:

- Implements an admission control mechanism that maintains interface parameters, such as the device's peak output rate, the percentage of bandwidth that can be reserved, and the maximum number of concurrent flows.
- Ensures that applications do not pace data at a rate faster than allowed.
- Interfaces with the `rsvpd` daemon and the `iftcntl` command to install and remove flows and filters.
- Matches all outgoing packet headers with any existing filter specifications to determine on which output queue to place the packets.

See `iftcntl(8)` for more information.

The `rsvpd` daemon requires that traffic control be enabled on the local system in order to install and modify flows and filters for a specific network interface. To enable traffic control on your local system, check that the `ether_cl_scheduler` system attribute is enabled (set to 1). If it is not enabled, enable it by using the `sysconfig` command or `dxkerneltuner`. Then, reboot the system.

2.9.2 Managing RSVP

RSVP assigns QoS to specific IP data flows or sessions, which can be either multipoint-to-multipoint or point-to-point. In order to receive data packets for a particular multicast session, a host must have joined the corresponding IP multicast group. A given session may have multiple senders and if the destination is a multicast address, multiple receivers.

The `rsvpd` daemon performs the following functions:

- Listens for incoming RSVP messages
- Communicates with RSVP-enabled applications on the local host through RAPI
- Interfaces with the operating system's Traffic Control subsystem

See `rsvpd(8)` for more information.

2.9.2.1 Starting and Stopping `rsvpd`

To start the `rsvpd` daemon, enter the following command:

```
# /usr/sbin/rsvpd
```

If you want to start the daemon automatically at system boot time, include the command in the `/etc/inet.local` file. See `rsvpd(8)` for more information on the daemon and its options.

To stop the `rsvpd` daemon, enter the following command:

```
# kill -9 `cat /var/run/rsvdpd.pid`
```

The `rsvdpd` daemon does not start or stop any applications during its startup or shutdown procedures. It also does not maintain any on-disk configuration information about applications. Whenever the `rsvdpd` daemon starts, it has no information about previous reservations.

Typically all daemons on the operating system are started or stopped together, as the system changes run levels. But applications must correctly handle situations where they start before the `rsvdpd` daemon, or are running while the `rsvdpd` daemon is restarted. In these situations, local applications need to reinitiate communications with the `rsvdpd` daemon.

2.9.2.2 Adding and Deleting Network Interfaces

When you add or delete a network interface on your system, you must stop and restart the `rsvdpd` daemon in order for it to update its table of available interfaces. Enter the following commands:

```
# kill -9 `cat /var/run/rsvdpd.pid`  
# /usr/sbin/rsvpd
```

2.9.2.3 Displaying RSVP Session Information

You can display RSVP session information on routing systems or end systems to determine if RSVP is working correctly on your system. RSVP session information will show you if connections are being set up and if reservations are being honored.

To monitor active RSVP sessions on the local system, enter the following command:

```
# /usr/sbin/rsvpsstat
```

By default, the `rsvpsstat` command displays a list of all RSVP sessions, sender and receiver, active on this system. Information includes the session number, destination address, IP protocol, port number, and the number of PATH and RESV states for the session.

To display sender information, including the contents of the actual PATH message from the sender, enter the following command:

```
# /usr/sbin/rsvpsstat -Sv
```

To display receiver information, including the contents of the actual RESV message from the receiver, enter the following command:

```
# /usr/sbin/rsvpstat -Rv
```

See `rsvpstat(8)` for more information.

Internet Protocol Version 6

Internet Protocol Version 6 (IPv6) is both a completely new network layer protocol and a major revision of the Internet architecture. As such, it builds upon and incorporates experiences gained with IPv4. This chapter describes the following:

- The history and purpose of IPv6 (Section 3.1)
- Terms (Section 3.2)
- IPv6 addressing (Section 3.3)
- The IPv6 environment (Section 3.4)
- How to plan for your IPv6 configuration (Section 3.5)
- How to configure your system to support IPv6 addressing (Section 3.6)
- How to perform post-configuration tasks (Section 3.7)
- How to log IPv6 activity (Section 3.8)

For troubleshooting information, see Section 8.4.

3.1 Introduction to IPv6

In the early 1990s the members of the Internet community realized that the address space and certain aspects of the current TCP/IP architecture were not capable of sustaining the explosive growth of the Internet. The problems included the exhaustion of the Internet address space, the size of routing tables, and requirements for new technology features.

The Internet Engineering Task Force (IETF) made several efforts to study and improve the use of the 32-bit Internet Protocol (IPv4) addresses. They also tackled the longer-term goal of identifying and replacing protocols and services that would limit growth.

These efforts identified the 32-bit addressing architecture of IPv4 as the principal problem, in terms of router overhead and of network administration. In addition, IPv4 addresses were often unevenly allocated in blocks that were too large or too small, and therefore difficult to change within any existing network.

In July 1994, the Internet Protocol Next Generation (IPng) directorate announced the Internet Protocol Version 6 (IPv6) as the replacement network

layer protocol, and IETF working groups began to build specifications. See RFC 1752, “The Recommendation for the IP Next Generation Protocol,” for additional information on the IPv6 protocol selection process.

3.2 Terms

The following terms are used in this chapter:

node

Any system that uses the IPv6 protocol to communicate.

router

A node that forwards IPv6 packets addressed to other nodes. These systems typically have more than one network interface card (NIC) installed and configured.

host

Any node that is not a router.

link

A medium or facility over which nodes communicate with each other at the link layer. Examples include Ethernet, FDDI, PPP links, or internet layer tunnels.

interface

A node’s attachment to a link, which is usually assigned an IPv6 address or addresses. This can be a physical NIC (for example, `tu0` or `ee0`) or virtual network interface (for example, `ipt0`, described in Section 3.5.2.2).

3.3 IPv6 Addressing

This section is intended for administrators who need an introduction to IPv6 addressing. If you already know this information, skip to Section 3.4.

The most noticeable feature of IPv6 is the IPv6 address. The address size is increased from 32 bits to 128 bits. This section describes:

- Address text representation
- Address autoconfiguration
- Address resolution
- Address assignment

3.3.1 Address Text Representation

You can use the following syntax to represent IPv6 addresses as text strings:

`x::x::x::x::x::x`

The *x* is a hexadecimal value of a 16-bit piece of the address. For example, the following addresses are IPv6 addresses:

```
FEDC:BA98:7654:3210:FEDC:BA98:7654:3210
```

```
1070:0:0:0:0:800:200C:417B
```

IPv6 addresses can contain long strings of zero (0) bits. To make it easier to write these addresses, you can use two colon characters (:) one time in an address to represent 1 or more 16-bit groups of zeros. For example, you can compress the second IPv6 address example as follows:

```
1070::800:200C:417B
```

Alternatively, you can use the following syntax to represent IPv6 addresses in an environment of IPv4 and IPv6 nodes:

```
x::x::x::x::d.d.d.d
```

In this case, *x* is a hexadecimal value of a 16-bit piece of the address (six high-order pieces) and *d* is a decimal value of an 8-bit piece of address (four low-order pieces) in standard, dotted-quad IPv4 form. For example, the following are IPv6 addresses:

```
0:0:0:0:0:0:13.1.68.3
```

```
0:0:0:0:0:FFFF:129.144.52.38
```

When compressed, these addresses are as follows:

```
::13.1.68.3
```

```
::FFFF:129.144.52.38
```

Like IPv4 address prefixes, IPv6 address prefixes are represented using the Classless Inter-Domain Routing (CIDR) notation. This notation has the following format:

```
ipv6-address/prefix-length
```

For example, you can represent the 60-bit hexadecimal prefix 12AB00000000CD3 in any of the following ways:

```
12AB:0000:0000:CD30:0000:0000:0000:0000/60
```

```
12AB::CD30:0:0:0:0/60
```

```
12AB:0:0:CD30::/60
```

3.3.2 Types of Addresses

There are three types of IPv6 addresses:

- Unicast
- Anycast
- Multicast

Note

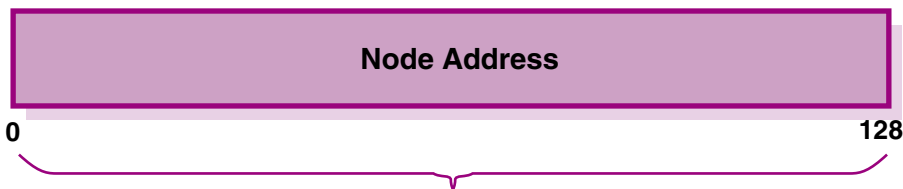
Unlike IPv4, IPv6 does not define a broadcast address. To get the function of a broadcast address, use a multicast address with link-local scope (see Section 3.3.2.2).

The following sections describe only the unicast and multicast address types and provide examples.

3.3.2.1 Unicast Address

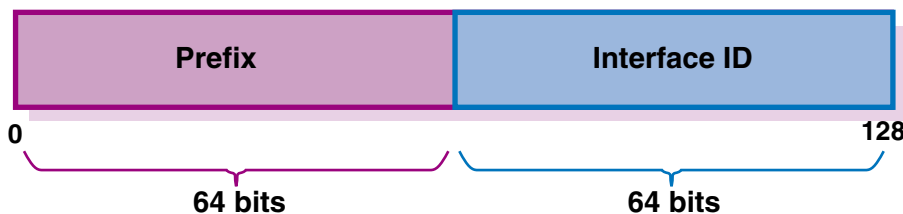
A unicast address is an identifier for a physical network interface. Packets sent to a unicast address are delivered to the node containing the interface identified by the address.

Unicast addresses typically have the following format:



ZK-1291U-AI

This address typically consists of a 64-bit prefix followed by a 64-bit interface ID as follows:



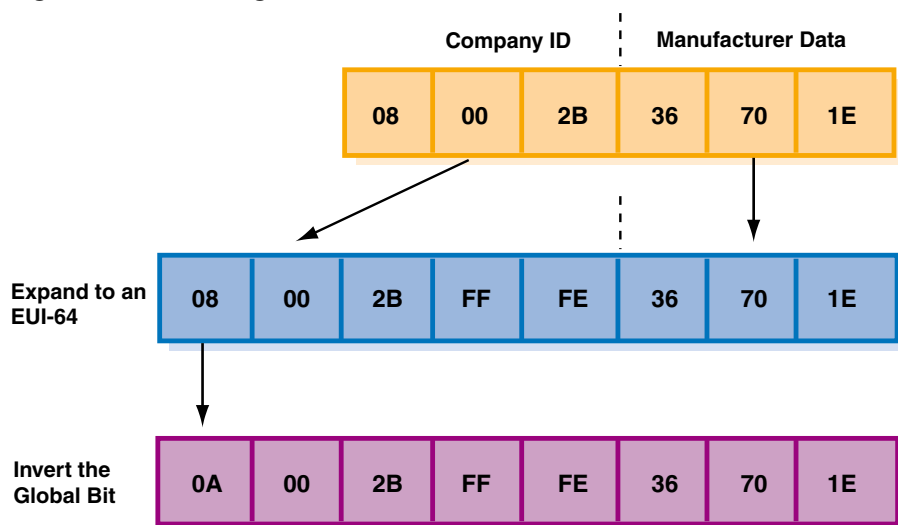
ZK-1292U-AI

An interface ID identifies an interface on a link. The interface ID must be unique on a link, but can also be unique over a broader scope. In many cases, an interface's ID is derived from its link-layer address. The same interface ID can be used on multiple interfaces on a single node.

According to RFC 2373, most prefixes must have 64-bit interface identifiers. For 48-bit MAC addresses, the interface identifier is created by inserting the hexadecimal values of 0xFF and 0xFE in the middle of the address

and inverting the universal/local bit (bit 7) in the resulting 64-bit address. Figure 3–1 shows how this process works.

Figure 3–1: Creating an Interface ID from a MAC Address



ZK-1849U-AI

The following list describes commonly used unicast addresses and their values:

Unspecified address

Indicates the absence of an address, and is never assigned to an interface. The unspecified address has the value 0:0:0:0:0:0:0:0 in the normal form or :: in the compressed form.

Loopback address

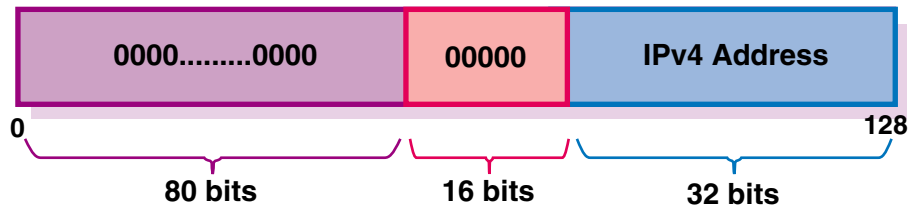
Used by a node to send IP datagrams to itself, and is typically assigned to the loopback interface. The IPv6 loopback address has the value 0:0:0:0:0:0:0:1 in the normal form or ::1 in the compressed form.

IPv6 addresses with embedded IPv4 addresses

Used in mixed IPv4 and IPv6 environments, and can be either of the following:

- IPv4-compatible IPv6 address

Used by IPv6 nodes to tunnel IPv6 packets across an IPv4 routing infrastructure. The IPv4 address is carried in the low-order 32-bits. The format of this address is as follows:

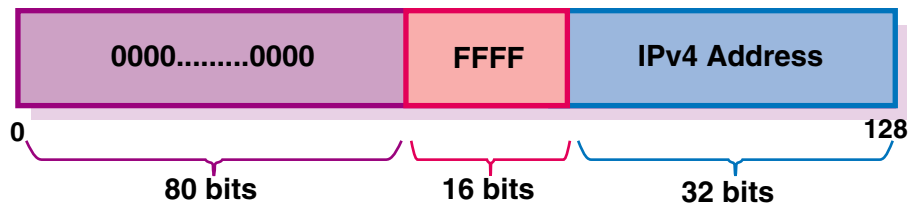


ZK-1293U-AI

Note

Do not use IPv4-compatible IPv6 addresses in the Domain Name System (DNS) or the local `/etc/ipnodes` file.

- IPv4-mapped IPv6 address
Used to represent an IPv4 address and to identify nodes that do not support IPv6 (IPv4-only nodes). It is not used in an IPv6 packet. The format of this address is as follows:

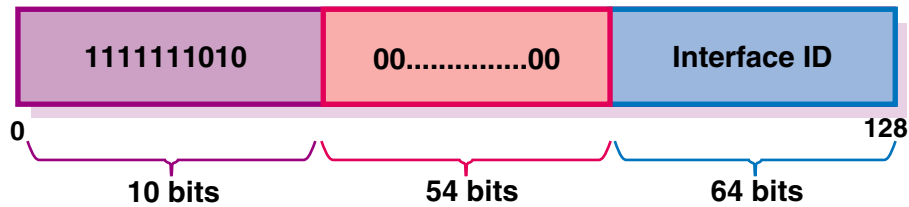


ZK-1294U-AI

Local-use IPv6 unicast addresses

Can be either of the following:

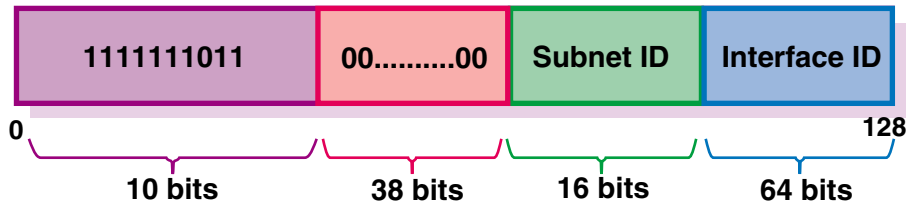
- Link-local
Used for addressing on a single link when performing address autoconfiguration, neighbor discovery, or when no routers are present. This address is assumed to be unique only on the link to which the interface is connected. The format of this address is as follows:



ZK-1295U-AI

- Site-local

Used for sites or organizations that are not connected to the global Internet. This address is assumed to be unique only in the site to which the interface is connected. The format of this address is as follows:



ZK-1296U-AI

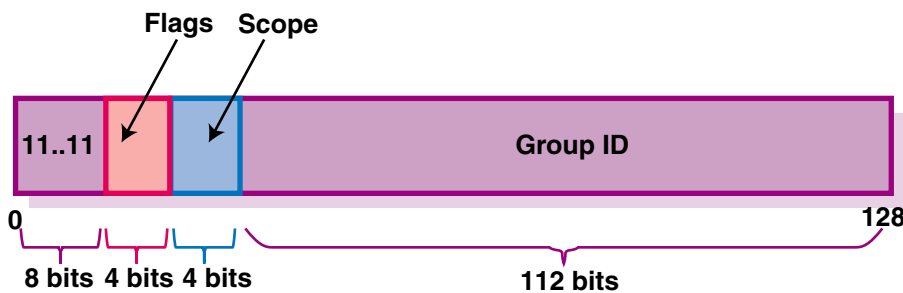
If you plan to use site-local addresses, be aware of the following guidelines:

- Do not connect a single node to multiple sites.
- Do not use site-local addresses in the global DNS (the addresses cannot be visible outside the site).
- Dynamic DNS updates for site-local addresses are not supported.
- Do not advertise or propagate routes containing site-local prefixes outside the site.

Interfaces typically have multiple IPv6 addresses. After IPv6 is configured and the system boots, the LAN, PPP, and configured tunnel interfaces are automatically assigned a link-local address. If a router is on the link, the system also autoconfigures a global unicast address on the interfaces.

3.3.2.2 Multicast Address

A multicast address is an identifier for a group of nodes, similar to an IPv4 multicast address. Multicast addresses have the following format:



ZK-1303U-AI

In the preceding address format, the fields have the following definition:

11111111 Identifies the address as multicast.

Flags Can be either 0000, which indicates a permanently-assigned (well-known) multicast address, or 0001, which indicates a temporary (transient) multicast address.

Scope Indicates the scope of the multicast group. The following table lists the scope values:

Value (Hex)	Scope
1	Node-local
2	Link-local
5	Site-local
8	Organization-local
E	Global

Group ID Identifies the multicast group within the specified scope.

Table 3–1 lists some well-known multicast addresses.

Table 3–1: Well-Known Multicast Addresses

Multicast Address	Meaning
FF02::1	All nodes (link-local)
FF02::2	All routers (link-local)
FF02::9	All RIPng routers (link-local)

3.3.3 Address Prefixes

Each IPv6 address has a unique pattern of leading bits that indicates its address type. These leading bits are named the format prefix (also referred to as a prefix). Table 3–2 lists some of the IPv6 address types and their prefixes.

Table 3–2: IPv6 Address Types and Prefixes

Address Type	Prefix
Aggregatable Global Unicast	2000::/3
Link-local	FE80::/10
Site-local	FEC0::/10
Multicast	FF00::/8

3.3.4 Address Autoconfiguration

The IPv6 address changes have led to the following definitions for configuring addresses:

- Stateless address autoconfiguration
- Dynamic Host Configuration Protocol Version 6 (DHCPv6), which is stateful address autoconfiguration

In the stateless model, nodes learn address prefixes by listening for Router Advertisement packets. Addresses are formed by combining the prefix with a datalink-specific interface identifier, which is typically derived from the datalink address of the interface. This model is favored by administrators who do not need tight control over address configuration. See RFC 2462 for more information.

In DHCPv6, hosts can request addresses, configuration information, and services from dedicated configuration servers. This model is favored by administrators who want to delegate addresses based on a client/server model. The DHCPv6 Internet Drafts are currently undergoing revision. See the Dynamic Host Configuration charter web page at <http://www.ietf.org/html.charter/dhc-charters.html> for more information.

Note

This version of Tru64 UNIX does not support DHCPv6.

In both cases, the resulting addresses have associated lifetimes, and systems must be able to acquire new addresses and release expired addresses. Combined with the ability to register updated address information with Domain Name System (DNS) servers, these mechanisms provide a path towards network renumbering and provide network administrators with control over the use of network addresses without manual intervention on each host on the network.

3.3.5 Address Resolution

The Domain Name System (DNS) provides support for mapping names to IP addresses and mapping IP addresses back to their corresponding names. Because of the increase in size of the IPv6 address, the DNS has the following new features:

- AAAA resource record type
This holds IPv6 addresses, encoded in network byte order. The version of BIND shipped with operating system supports AAAA records. (BIND is the implementation of DNS that ships with Tru64 UNIX.)
- AAAA query
A query for a specified domain name in the Internet class returns all associated AAAA resource records in the response.
- IP6.INT domain for looking up a name for a specified address (address-to-name mapping)

An IPv6 address is represented in reverse order as a sequence of 4-bit nibbles separated by dots with the suffix `.IP6.INT` appended. For example, the IPv6 address `4321:0:1:2:3:4:567:89ab` has the following inverse lookup domain name:

```
b.a.9.8.7.6.5.0.4.0.0.0.3.0.0.0.2.0.0.0.1.0.0.0.0.0.0.0.1.2.3.4.IP6.INT
```

See *Network Administration: Services* for guidelines on configuring BIND in an IPv6 environment.

3.3.6 Address Assignment

IPv6 addresses are now being deployed by the regional registries. If you connect your system to a network that already runs IPv6, your system will automatically configure the IPv6 addresses it needs.

If you are a network administrator, contact your Internet Service Provider (ISP) for an IPv6 address range for your site. See the IANA web page at

<http://www.iana.org/ipaddress/ip-addresses.htm> for more information about regional registries and address allocations.

Because of the need to test various implementations of the IPv6 RFCs, the IETF has defined a temporary IPv6 address allocation scheme. You can assign the addresses in this scheme to hosts and routers for testing IPv6 on the 6bone. See the 6bone home page at the following location for more information on 6bone address allocation and assignment:

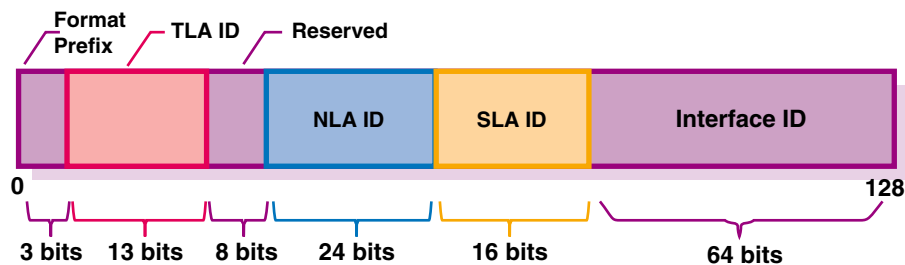
<http://www.6bone.net>

At the present time, the 6bone test addresses are aggregatable global unicast addresses. Contact your 6bone service provider (for example, `gw-6bone@pa.dec.com`) for a 6bone address delegation.

The following sections describe the aggregatable global unicast addresses and the aggregatable testing addresses.

3.3.6.1 Aggregatable Global Unicast Address Format

The aggregatable global unicast address format for IPv6 is designed to support current provider-based aggregation and new exchange-based aggregation. Whether a site connects to a provider or to an exchange, the address format enables efficient route aggregation for either type. Aggregatable global unicast addresses have the following form. See RFC 2374 for additional information.



ZK-1301U-AI

In the preceding address format, the fields have the following definition:

Format Prefix

The Format Prefix. For aggregatable global unicast addresses, the value for this field is 001.

TLA ID

The Top-Level Aggregation Identifier.

Reserved

Reserved for future use. At present, set to all zeros (0).

NLA ID

The Next-Level Aggregation Identifier. These are assigned by the TLA ID administrator to create an addressing hierarchy and to identify end user sites. Each organization assigned a TLA ID is also assigned 24-bits of NLA ID space whose layout and use is the responsibility of the organization.

SLA ID

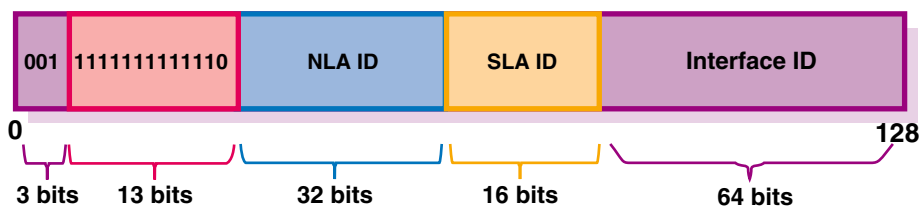
The Site-Level Aggregation Identifier. These are used by an end user site to create its own local addressing hierarchy and to identify subnets.

Interface ID

The 64-bit interface identifier of the interface that is connected to the link.

3.3.6.2 Aggregatable Testing Address Format

Aggregatable global unicast addresses for IPv6 testing have the following form. See RFC 2471 for more information on the proposed testing address allocation plan.



ZK-1341U-AI

In the preceding address format, the fields have the following definition:

001

The Format Prefix for aggregatable global unicast addresses.

11111111111110

The 6bone Top-Level Aggregation (TLA) Identifier, 0x1FFE, reserved by the Internet Assigned Numbers Naming Authority (IANA), and used temporarily for IPv6 testing.

Next-Level Aggregation (NLA) Identifier

The ID assigned by the TLA ID administrator to create an addressing hierarchy and to identify end user sites on the 6bone network.

Site-Level Aggregation (SLA) Identifier

The ID assigned by an end user site to create its own local addressing hierarchy and to identify subnets.

Interface ID

The 64-bit interface identifier of the interface that is connected to the link.

For the most recent information about pTLA and pNLA assignments, see the 6bone home page at the following location:

<http://www.6bone.net>

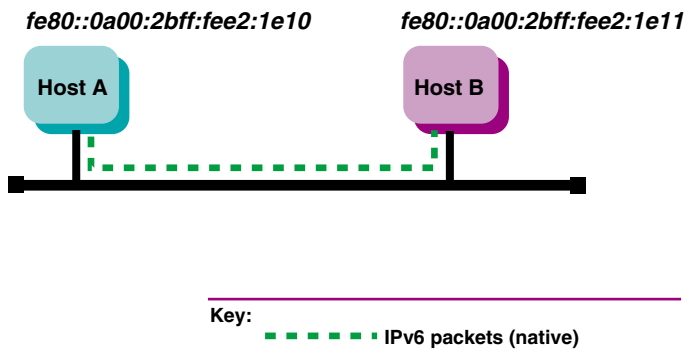
3.4 IPv6 Environment

This section shows some sample IPv6 configurations. Select a configuration that most closely matches the environment into which you want to configure IPv6 on your system. These configurations are used again in Section 3.5 to describe how to configure selected systems in each configuration. For those configurations that show an IPv6 global address or address prefix, the addresses use the format described in Section 3.3.6.2.

IPv6 is supported LAN and PPP network interfaces. See the *Technical Overview* for a list of commands and daemons that are supported in an IPv6 environment.

Figure 3–2 shows a simple LAN configuration in which Host A and Host B communicate using IPv6.

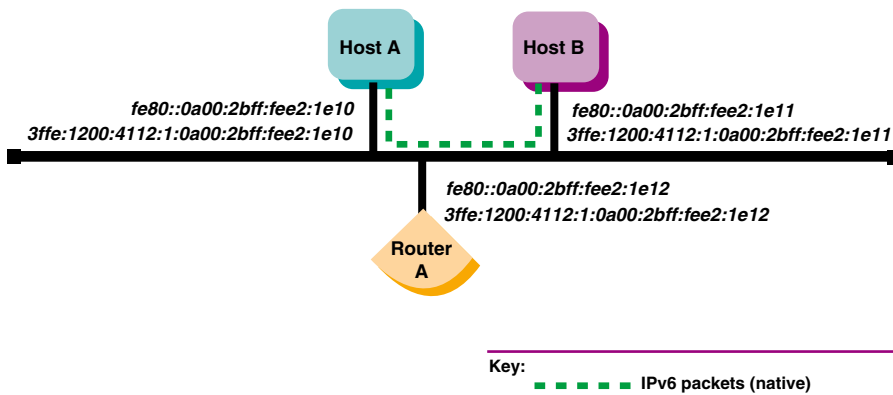
Figure 3–2: Simple Host-to-Host Configuration



ZK-1348U-AI

Figure 3–3 shows a simple LAN configuration in which Host A, Host B, and Router A communicate using IPv6 and in which Host A and Host B obtain global addresses from Router A.

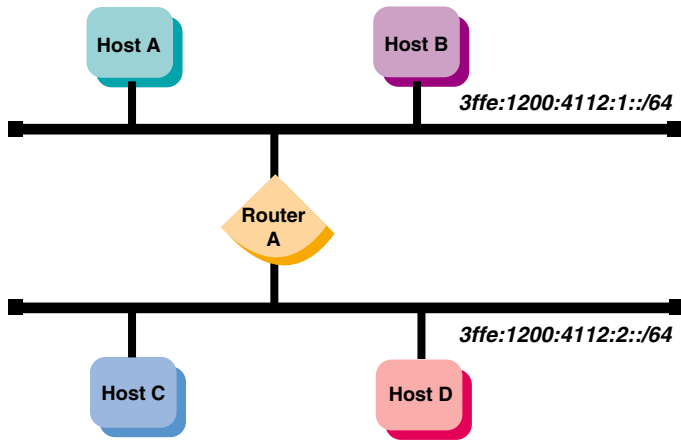
Figure 3–3: Host-to-Host with Router Configuration



ZK-1349U-AI

Figure 3–4 shows a configuration in which two IPv6 networks are connected through an IPv6 router, Router A.

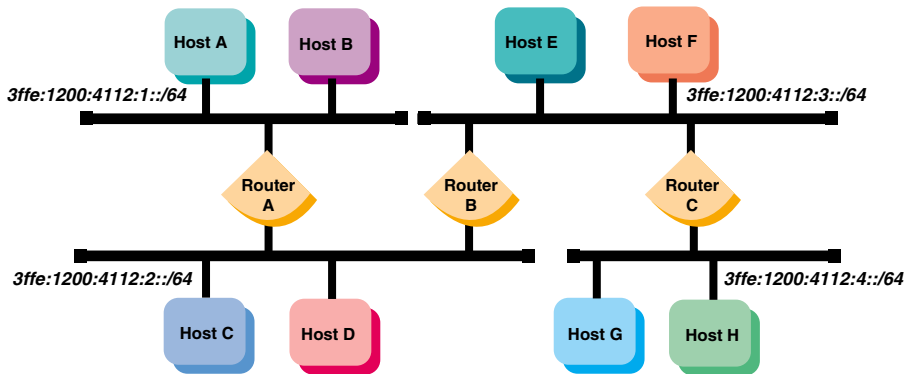
Figure 3–4: IPv6 Network-to-IPv6 Network with Router Configuration



ZK-1350U-AI

Figure 3–5 shows a configuration in which four IPv6 networks are connected using three routers. The three routers exchange routing information with each other using the RIPng protocol.

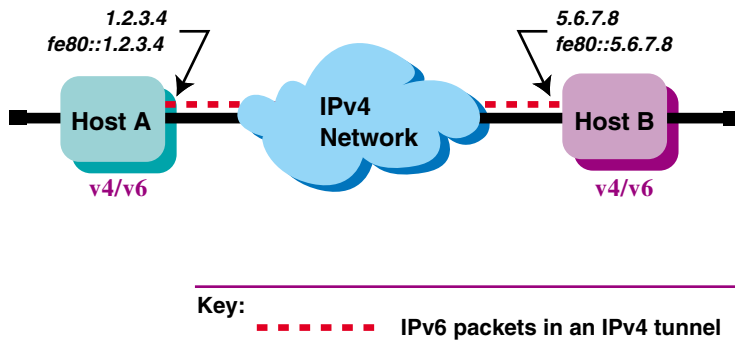
Figure 3–5: Multiple IPv6 Networks and Multiple Routers Configuration



ZK-1351U-AI

Figure 3–6 shows a configuration in which Host A and Host B, connected to an IPv4 network, communicate using IPv6 through an IPv4 tunnel.

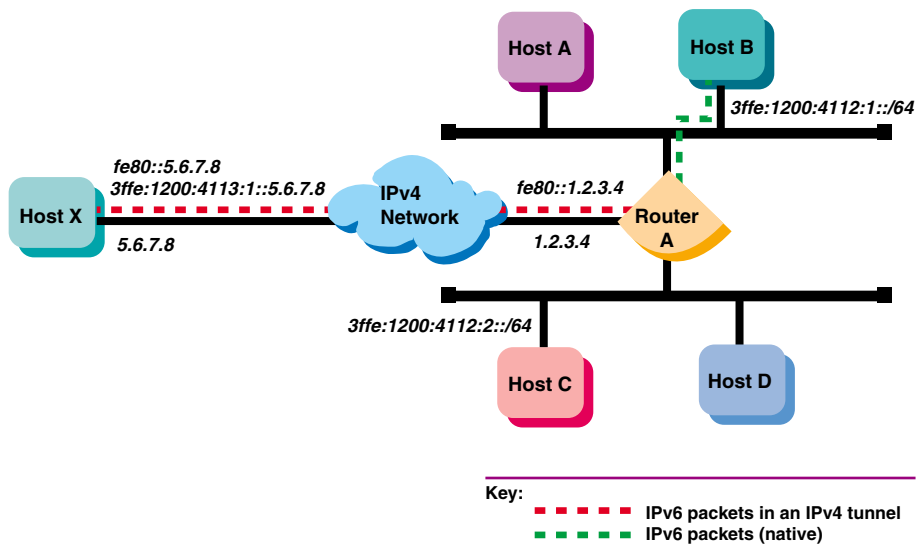
Figure 3–6: Host-to-Host over Tunnel Configuration



ZK-1298U-AI

Figure 3–7 shows a configuration in which Host X is connected to an IPv4 network and Router A, an IPv6 router, is connected to the same IPv4 network and also is connected to two IPv6 networks. Host X communicates with Host B using IPv6 through an IPv4 tunnel between Host X and Router A.

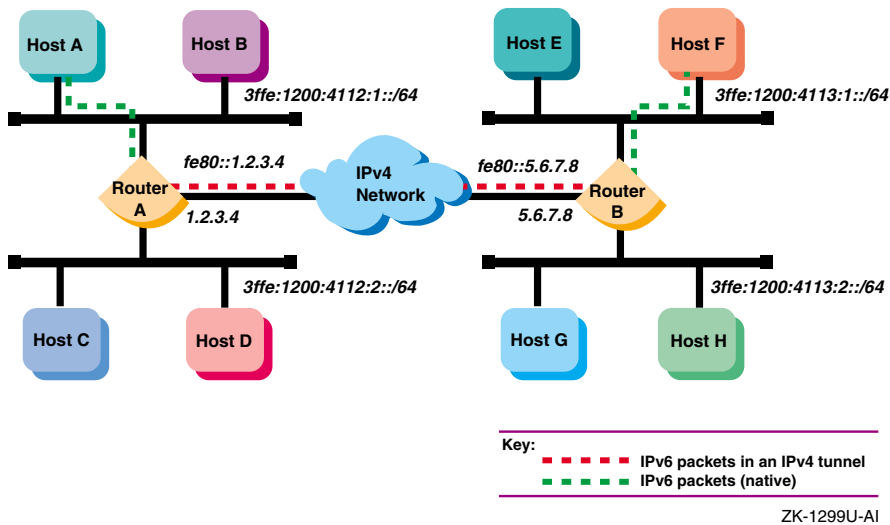
Figure 3–7: Host-to-Router over Tunnel Configuration



ZK-1347U-AI

Figure 3–8 shows a configuration in which four IPv6 networks are connected through two routers and an IPv4 network. Host A communicates with Host F through an IPv4 tunnel between router A and router B.

Figure 3–8: IPv6 Network-to-IPv6 Network over Tunnel Configuration



3.5 Planning IPv6

You can configure IPv6 on any node. For cluster members, you can configure IPv6 on each individual cluster member independently.

Note

IPv6 does not support cluster-wide communication. You cannot use an IPv6 address for the cluster alias. See the *Cluster Administration* manual for information on configuring a cluster.

This section describes those tasks that you need to do before configuring IPv6.

3.5.1 Verifying IPv6 Support in the Kernel

Verify that the IP Version 6 (IPV6) and IP-in-IP Tunneling (IPTUNNEL) support is in the kernel by issuing the following commands:

```
# sysconfig -q ipv6
# sysconfig -q iptunnel
```

If neither the `ipv6:` nor the `iptunnel:` subsystem attributes are displayed, do the following:

1. Build a new kernel by using the following command:

```
# doconfig -c SYSTEM_NAME
```

Choose the IPV6 and IPTUNNEL options in addition to any other options that you want.

2. Save the original kernel, then move the new kernel to the root directory.

```
# mv /vmunix /vmunix.save  
# mv /sys/SYSTEM_NAME/vmunix /vmunix
```

3. Reboot the system. Make sure there are no other users on the system. Use a command similar to the following:

```
# shutdown -r +5 "Adding IPv6 and IPTUNNEL kernel options ..."
```

You are now ready to configure your system to communicate in an IPv6 network environment.

3.5.2 Preparing for the Configuration

After you verify IPv6 support in the kernel, you configure your system to communicate in an IPv6 network environment by running the IPv6 configuration utility, `ip6_setup`. The `ip6_setup` utility enables you to configure the following:

- IPv6 host
- IPv6 router

When you run the `ip6_setup` configuration utility, it gathers information from the system and prompts you for additional configuration information.

Before you configure the IPv6 network software, you must gather information about your system and network environment. Figure 3–9 shows the IPv6 Configuration Worksheet. The following sections describe the information that you need to record on the worksheet.

If you are viewing this manual on line, you can use the print feature to print a copy of this worksheet.

Figure 3–9: IPv6 Configuration Worksheet

IPv6 Configuration Worksheet	
IPv6 router:	<input type="checkbox"/> Yes <input type="checkbox"/> No
DNS/BIND automatic updates (hosts only):	<input type="checkbox"/> Yes <input type="checkbox"/> No
IPv6 interfaces:	_____
IPv6 routing over PPP (routers only):	<input type="checkbox"/> Yes <input type="checkbox"/> No
Configured tunnel:	<input type="checkbox"/> Yes <input type="checkbox"/> No
Automatic tunnel:	<input type="checkbox"/> Yes <input type="checkbox"/> No
Manual routes:	<input type="checkbox"/> Yes <input type="checkbox"/> No
Start IPv6:	<input type="checkbox"/> Yes <input type="checkbox"/> No
DNS/BIND	
Domain name:	_____
Configured Tunnel	
Interface:	_____
Destination IPv4 address:	_____
Source IPv4 address:	_____
RIPng:	<input type="checkbox"/> Yes <input type="checkbox"/> No
Address prefix:	_____
Router	
Interface:	_____
RIPng:	<input type="checkbox"/> Yes <input type="checkbox"/> No
Address prefix:	_____
Interface:	_____
RIPng:	<input type="checkbox"/> Yes <input type="checkbox"/> No
Address prefix:	_____
Manual Routes	
Destination prefix:	_____
Interface:	_____
Next hop address:	_____
Destination prefix:	_____
Interface:	_____
Next hop address:	_____

IPv6 router

If you want this system to function as an IPv6 router, check Yes; otherwise, check No. If you check No, the system is configured as an IPv6 host.

An IPv6 router can advertise address prefixes to all hosts on connected links (for example, a LAN and a configured tunnel) and forward packets toward their destinations. Packets can be forwarded directly on the link or over IPv4 tunnels.

DNS/BIND automatic updates (hosts only)

If you want this system to record its addresses in the DNS/BIND database automatically, check Yes; otherwise, check No. If you check Yes, you must configure your system as a DNS/BIND client and your DNS/BIND server must support dynamic updates to the DNS database. See *Network Administration: Services* for information on configuring your DNS/BIND server.

IPv6 interfaces

Enter the device names of the network interface to the IPv6 network. For example, `le0` and `fta0`. If you are creating a configured tunnel only on your system, enter `none`.

IPv6 routing over PPP (routers only)

If you want IPv6 routing to run over a PPP interface, check Yes; otherwise, check No. See `ppp_manual_setup(7)` for information on configuring a PPP interface.

Configured tunnel

If you want IPv6 to run over a configured IPv4 tunnel, check Yes; otherwise, check No. A configured tunnel has one source and one destination in an IPv4 network. Use configured tunnels instead of automatic tunnels. You can define multiple configured tunnels.

Automatic tunnel

If you want to configure IPv6 to run over IPv4 automatic tunnels, check Yes; otherwise, check No.

Manual routes

If you want to configure routes to other systems manually, check Yes; otherwise, check No.

On a router, you might want to configure static routes if one of the following conditions is true:

- You want a configured tunnel and you are not advertising an address prefix on the tunnel link.
- You want a configured tunnel and the router at the other end of the tunnel is not running the RIPng protocol.
- Your system is not running the RIPng protocol.

On a host, you might want to configure static routes if you want a configured tunnel to a router and the router is not advertising itself as a default router on the tunnel link.

Start IPv6

If you want to start IPv6 directly from the configuration utility, `ip6_setup`, check Yes. If you want to start IPv6 during the next system boot, check No.

3.5.2.1 DNS/BIND

Domain name

The fully qualified domain name for your node. This consists of the host name and the DNS/BIND domain name (for example, `host1.subdomain.example`).

3.5.2.2 Configured Tunnel

Interface

The name of the configured tunnel interface (for example, `ipt0`, `ipt1`). The `ip6_setup` script supplies this value.

Destination IPv4 address

The remote node's IPv4 address (the remote end of the tunnel).

Source IPv4 address

Your node's IPv4 address (this end of the tunnel).

RIPng

If your system is a router and you want the router to run the RIPng protocol on the tunnel link to exchange IPv6 routing information with a router at the remote end of the tunnel, check Yes; otherwise, check No.

Address prefix

If your system is a router and you want to advertise address prefixes to the node at the remote end of the tunnel, enter a 64-bit prefix; otherwise, write Done.

If your system is an IPv6 host and the router at the remote end of the tunnel is not advertising an address prefix, enter a 64-bit prefix to be configured on the tunnel interface.

3.5.2.3 Router

Interface

The name of the interface (LAN, PPP, or configured tunnel) on which you want to run the RIPng protocol or advertise an address prefix.

RIPng

If you want the router to run the RIPng protocol on the specified interface and to exchange IPv6 routing information with other routers on the link (LAN, PPP, or configured tunnel), check Yes; otherwise, check No.

Address prefix

If you want to advertise address prefixes to all hosts on the link, enter a 64-bit prefix; otherwise, write Done.

If you write Done, the router will not advertise an address prefix. All hosts must obtain their prefix information from another source.

Prefixes in IPv6 define a subnet, and are typically configured on a router for a specific link by the network administrator. The router advertises this prefix to all nodes connected to that link, along with the length of the prefix, whether the prefix is on link (that is, a neighbor), whether the prefix can also be used for stateless address configuration, and the length of time the prefix is valid.

3.5.2.4 Manual Routes

Destination prefix

The address prefix of a remote IPv6 network. The address prefix contains a Classless Inter-Domain Routing (CIDR) style bit length, for example, 5F00::/8. If you want to use the default route, write Default.

Interface

The name of the interface through which you are sending traffic to the remote IPv6 network.

Next hop address

The IPv6 address of the first router in the path to the destination prefix. Write the link local address of the router. If the connection to the router is over an IPv4 tunnel, write the link local IPv6 address of the remote tunnel endpoint.

3.5.3 Configuring Systems in Sample IPv6 Configurations

This section describes each sample configuration presented in Section 3.4 and shows how selected systems are configured in each example. In some cases, this section presents additional options for you to consider in the configuration.

3.5.3.1 Simple Host-to-Host Configuration

In Figure 3–2, Host A and Host B use IPv6 link-local addresses. By default, the `ip6_setup` configuration utility automatically creates a link-local address for your system. The following is a sample completed worksheet for Host A:

IPv6 Configuration Worksheet		
IPv6 router:	<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
DNS/BIND automatic updates:	<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
IPv6 interfaces:	<u>tu0</u> _____	
Configured tunnel:	<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
Automatic tunnel:	<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
Manual routes:	<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
Start IPv6:	<input checked="" type="checkbox"/> Yes	<input type="checkbox"/> No

After configuring IPv6 on Host A, you edit the `/etc/ipnodes` file and insert the link-local address for Host B. The configuration process for Host B in this configuration is similar to Host A's.

With this configuration, no global address prefix is advertised on the LAN. If you want to advertise a global address prefix, you could either configure one of the nodes as a router by using the `ip6_setup` utility or add an IPv6 router to the LAN configuration. An IPv6 router advertises a global prefix on the link.

You can use the `netstat -in` command to view a local node's link-local and global addresses.

If you are on Host A and want to connect to Host B using the `telnet` command, the format of the command is as follows:

```
# telnet fe80::0a00:2bff:fee2:1e11
```

Instead of specifying the link-local address, place the address and the node name in the `/etc/ipnodes` file. Then, use the node name as the argument to the `telnet` command.

3.5.3.2 Host-to-Host with Router Configuration

In Figure 3–3, Host A and Host B are on a LAN with Router A. In this case, Router A advertises the global address prefix `3ffe:1200:4112:1::/64` on

the LAN. Host A and Host B use this address prefix to create global IPv6 addresses. See Section 3.3.6 for more information on obtaining experimental testing addresses. The following is a sample completed worksheet for Router A:

IPv6 Configuration Worksheet	
IPv6 router:	<input checked="" type="checkbox"/> Yes <input type="checkbox"/> No
DNS/BIND automatic updates:	<input type="checkbox"/> Yes <input checked="" type="checkbox"/> No
IPv6 interfaces:	<u>tu0</u>
Configured tunnel:	<input type="checkbox"/> Yes <input checked="" type="checkbox"/> No
Automatic tunnel:	<input type="checkbox"/> Yes <input checked="" type="checkbox"/> No
Manual routes:	<input type="checkbox"/> Yes <input checked="" type="checkbox"/> No
Start IPv6:	<input checked="" type="checkbox"/> Yes <input type="checkbox"/> No
Router	
Interface:	<u>tu0</u>
RIPng:	<input type="checkbox"/> Yes <input checked="" type="checkbox"/> No
Address prefix:	<u>3ffe:1200:4112:1::/64</u>
Interface:	_____
RIPng:	<input type="checkbox"/> Yes <input type="checkbox"/> No
Address prefix:	_____

After configuring IPv6 on Router A, you can edit the `/etc/ipnodes` file and add the global addresses for the other nodes. You would also do this on Host A and Host B. Alternatively, you could establish DNS/BIND in your network using the global addresses.

If you added a DNS/BIND server with dynamic updates enabled on the network, the worksheet for Host A would have the following information:

DNS/BIND automatic updates:	<input checked="" type="checkbox"/> yes <input type="checkbox"/> no
DNS/BIND	
Domain name:	<u>hosta.corp.example</u>

3.5.3.3 IPv6 Network-to-IPv6 Network with Router Configuration

In Figure 3–4, two IPv6 networks are connected to each other through Router A and its multiple interfaces. The following is a sample completed worksheet for Router A:

IPv6 Configuration Worksheet

IPv6 router: Yes No
DNS/BIND automatic updates: Yes No
IPv6 interfaces: tu0 tu1
Configured tunnel: Yes No
Automatic tunnel: Yes No
Manual routes: Yes No
Start IPv6: Yes No

Router

Interface: tu0
RIPng: Yes No
Address prefix: 3ffe:1200:4112:1::/64

Interface: tu1
RIPng: Yes No
Address prefix: 3ffe:1200:4112:2::/64

3.5.3.4 Multiple IPv6 Networks and Multiple Routers Configuration

In Figure 3–5, four IPv6 networks are interconnected to each other using the three routers. In this configuration, the routers must exchange routing information in order for the routers to learn the routes to other subnets in the network. To accomplish this, each router must run the RIPng protocol. The following is a sample completed worksheet for Router A:

IPv6 Configuration Worksheet

IPv6 router: Yes No
DNS/BIND automatic updates: Yes No
IPv6 interfaces: tu0 tu1
Configured tunnel: Yes No
Automatic tunnel: Yes No
Manual routes: Yes No
Start IPv6: Yes No

Router

Interface: tu0
RIPng: Yes No
Address prefix: 3ff3:1200:4112:1::/64

Interface: tu1
RIPng: Yes No
Address prefix: 3ff3:1200:4112:2::/64

The worksheets for the other routers are similar.

worksheet for Host X when Router A is advertising itself as the default router for the tunnel link and advertising a global address prefix on the tunnel link:

IPv6 Configuration Worksheet	
IPv6 router:	<input type="checkbox"/> Yes <input checked="" type="checkbox"/> No
DNS/BIND automatic updates:	<input type="checkbox"/> Yes <input checked="" type="checkbox"/> No
IPv6 interfaces:	<u>none</u>
Configured tunnel:	<input checked="" type="checkbox"/> Yes <input type="checkbox"/> No
Automatic tunnel:	<input type="checkbox"/> Yes <input checked="" type="checkbox"/> No
Manual routes:	<input type="checkbox"/> Yes <input checked="" type="checkbox"/> No
Start IPv6:	<input checked="" type="checkbox"/> Yes <input type="checkbox"/> No
Configured Tunnel	
Interface:	<u>ipt0</u>
Destination IPv4 address:	<u>5.6.7.8</u>
Source IPv4 address:	<u>1.2.3.4</u>
RIPng:	<input type="checkbox"/> Yes <input checked="" type="checkbox"/> No
Address prefix:	_____

If Router A is not advertising a global address prefix on the tunnel link, the value `3ffe:1200:4113:1::/64` would be in the Address prefix field in Configured Tunnel section of the Host X worksheet. If Router A is not advertising itself as the default router for the tunnel link, the following information would also be on the Host X worksheet:

Manual routes:	<input checked="" type="checkbox"/> Yes <input type="checkbox"/> No
Manual Routes	
Destination prefix:	<u>default</u>
Interface:	<u>ipt0</u>
Next hop address:	<u>fe80::1.2.3.4</u>

The following is a sample completed worksheet for Router A when Router A is advertising a global address prefix on the tunnel link:

IPv6 Configuration Worksheet

IPv6 router: Yes No
DNS/BIND automatic updates: Yes No
IPv6 interfaces: tu0 tu1 _____
Configured tunnel: Yes No
Automatic tunnel: Yes No
Manual routes: Yes No
Start IPv6: Yes No

Configured Tunnel

Interface: ipt0
Destination IPv4 address: 5.6.7.8
Source IPv4 address: 1.2.3.4
RIPng: Yes No
Address prefix: 3ffe:1200:4113:1::/64

If Router A is not advertising a global prefix on the tunnel link, the following information would be on the Router A worksheet. Note the manual route to Host X. Instead of specifying a destination network prefix, you specify the host route, `3ffe:1200:4113:1::5.6.7.8`, to Host X. The next hop is the link-local IPv6 address of Host X's tunnel interface, `fe80::5.6.7.8`.

Manual routes: Yes No

Manual Routes

Destination prefix: 3ffe:1200:4113:1::5.6.7.8
Interface: ipt0
Next hop address: fe80::5.6.7.8

3.5.3.7 IPv6 Network-to-IPv6 Network over Tunnel Configuration

In Figure 3–8, Host A communicates with Host F over a configured tunnel through an IPv4 network. The host configuration is similar to that of Host A in Section 3.5.3.1. All nodes automatically use their default router in order to communicate with nodes on other networks. The following is a sample completed worksheet for Router A:

IPv6 Configuration Worksheet

IPv6 router: Yes No
DNS/BIND automatic updates: Yes No
IPv6 interfaces: tu0 tu1
Configured tunnel: Yes No
Automatic tunnel: Yes No
Manual routes: Yes No
Start IPv6: Yes No

Configured Tunnel

Interface: ipt0
Destination IPv4 address: 5.6.7.8
Source IPv4 address: 1.2.3.4
RIPng: Yes No
Address prefix: _____

Router

Interface: tu0
RIPng: Yes No
Address prefix: 3ffe:1200:4112:1::/64

Interface: tu1
RIPng: Yes No
Address prefix: 3ffe:1200:4112:2::/64

You do not have to run RIPng on the `tu0` and `tu1` interfaces because there are no routers attached to the interfaces.

The configuration of Router B is similar, except that the source and destination addresses for the configured tunnel are `1.2.3.4` and `5.6.7.8`, respectively, and the address prefixes advertised on `tu0` and `tu1` are `3ffe:1200:4113:1::/64` and `3ffe:1200:4113:2::/64`, respectively.

Note

If the routers were not configured to use RIPng over the tunnel interface, each router would then need to specify a manual route to the other.

3.6 Configuring IPv6 on Your System

This section describes how to configure your system as either an IPv6 host or an IPv6 router.

3.6.1 Configuring an IPv6 Host

To configure an IPv6 host, do the following:

1. Log in as superuser.
2. Invoke the `ip6_setup` utility by entering the following command:

```
# /usr/sbin/ip6_setup
```

The utility displays a status message.
3. Press Enter to indicate that you want to configure the system as an IPv6 host and not as a router.
4. Indicate whether you want your DNS/BIND client to update IPv6 addresses in the DNS/BIND name database automatically.
If you want to update IPv6 addresses in the DNS/BIND name database automatically, enter `y`; if you do not, press Enter and go to step 6.
5. Enter the IPv6 fully qualified domain name.
6. Enter the names of the IPv6 LAN interfaces to use. Separate multiple names by a space character. If you want to use the default, press Enter. If you are configuring a configured tunnel only, enter `none`.
7. Indicate whether you want to create a configured tunnel or additional configured tunnels.
If you want to create a configured tunnel or additional configured tunnels, enter `y`; otherwise, press Enter and go to step 11.
8. Enter the tunnel's destination IPv4 address.
If you are finished creating configured tunnels, enter `Done`. Go to step 11.
9. Enter the tunnel's source IPv4 address. If you want to use the default, press Enter.
10. Indicate whether the host is to use an IPv6 address prefix on the tunnel interface.
If you want the host to use an IPv6 address prefix because a router is not advertising a global address prefix, enter the prefix. Enter as many prefixes as you want. When you are finished entering prefixes for the interface, enter `Done`. Go to step 8.
If you do not want the host to use an IPv6 address prefix on the tunnel interface, enter `Done`. Go to step 8.
11. Indicate whether you want to configure an automatic tunnel.
If you want to configure an automatic tunnel, enter `y`; otherwise, press Enter.

12. Indicate whether you want to define manual routes to an adjacent router or remote IPv6 network.
If you want to manually define routes, enter `y`.
If you do not want to manually define routes, enter `n`. Go to step 16.
13. Enter the IPv6 address prefix of the remote IPv6 network. When you are finished entering manual routes, enter `Done`; go to step 16.
14. Enter the name of the interface through which you will send traffic to the remote IPv6 network.
15. Enter the link-local IPv6 address of the first router in the path to the destination network. This address together with the IPv6 address prefix constitute the static routing table entry. Go to step 13.
16. The `ip6_setup` utility displays the configuration information and asks you to indicate whether you want to update the current startup procedures with the new configuration information.
If you are not satisfied with the configuration, enter `n`. The utility ends immediately without changing any of the current configuration files.
If you are satisfied with the configuration, enter `y`. The `ip6_setup` utility updates the `/etc/rc.config` and `/etc/routes` files with the IPv6 configuration information.
17. If IPv6 is not currently running on your system, indicate whether you want to start IPv6 now.
If you want to start IPv6 now, press `Enter`. The `ip6_setup` utility starts IPv6.
If you do not want to start IPv6 now, enter `n`. IPv6 will start during the next system boot.
If IPv6 is currently running, indicate whether you want to restart it now.

The `/etc/rc.config` file contains configuration information used by the system startup scripts to start IPv6.

3.6.2 Configuring an IPv6 Router

To configure an IPv6 router, do the following:

1. Log in as superuser.
2. Invoke the `ip6_setup` utility by entering the following command:

```
# /usr/sbin/ip6_setup
```

The utility displays a status message.
3. Enter `y` to configure the system as an IPv6 router.

4. Enter the names of the IPv6 LAN interfaces to use. Separate multiple names by a space character. If you want to use the default, press Enter. If you are configuring a configured tunnel only, enter none.

Note

Use the next two steps when configuring IPv6 LAN interfaces, IPv6 over PPP interfaces, and IPv6 over IPv4 configured tunnel interfaces. Follow the directions carefully.

5. Indicate if the router is to run the RIPng protocol on the designated interface.
If you want to run the RIPng protocol, press Enter; if you do not, enter n.
6. Indicate if the router is to advertise an IPv6 address prefix on the designated interface.
If you want the router to advertise an IPv6 address prefix, enter the prefix. Enter as many prefixes as you want. When you are finished entering prefixes for the interface, enter Done.
If you do not want the router to advertise an IPv6 address prefix on the designated interface, enter Done.
If you are configuring additional LAN interfaces, go to step 5.
If you are configuring PPP interfaces, go to step 8.
If you are configuring IPv6 over IPv4 configured tunnels, go to step 10.
7. Indicate if you want to use IPv6 routing over PPP links.
If you want to use IPv6 routing over PPP links, enter y; otherwise, press Enter and go to step 9.
8. Enter the name of the PPP interface over which to run IPv6 and press Enter. Go to step 5.
If you are finished entering routing information for PPP interfaces, enter Done.
9. Indicate if you want to create IPv6 over IPv4 configured tunnels.
If you want to create configured tunnels, enter y; otherwise, press Enter and go to step 12.
10. Enter the tunnel's destination IPv4 address.
If you are finished creating configured tunnels, enter Done. Go to step 12.
11. Enter the tunnel's source IPv4 address. If you want to use the default, press Enter. Go to step 5.
12. Indicate if you want to configure an automatic tunnel.

If you want to configure an automatic tunnel, enter `y`; otherwise, press `Enter`.

13. Indicate if you want to define manual routes to an adjacent router or remote IPv6 network.

If you want to manually define routes, enter `y`.

If you do not want to manually define routes, enter `n`. Go to step 17.

14. Enter the IPv6 address prefix of the remote IPv6 network. When you are finished entering manual routes, enter `Done`. Go to step 17.
15. Enter the name of the interface through which you will send traffic to the remote IPv6 network.
16. Enter the IPv6 address of the next node in the path to the destination network. This address together with the IPv6 address prefix constitute the static routing table entry. Go to step 14.
17. The `ip6_setup` utility displays the configuration information and asks you to indicate whether you want to update the current startup procedures with the new configuration information.
If you are not satisfied with the configuration, enter `n`. The utility ends immediately without changing any of the current configuration files.
If you are satisfied with the configuration, enter `y`. The `ip6_setup` utility updates the `/etc/rc.config`, `/etc/routes`, and `/etc/ip6rtrd.conf` files with the IPv6 configuration information. You can modify these values as necessary.

18. If IPv6 is not currently running on your system, indicate whether you want to start IPv6 now.

If you want to start IPv6 now, press `Enter`. The `ip6_setup` utility starts IPv6.

If you do not want to start IPv6 now, enter `n`. IPv6 will start during the next system boot.

If IPv6 is currently running, indicate whether you want to restart it now.

The `/etc/rc.config`, `/etc/routes`, and `/etc/ip6rtrd.conf` files contain configuration information used by the system startup procedures to start IPv6. You can edit them to change your configuration.

3.7 Postconfiguration Tasks

After using the `ip6_setup` utility to initially configure IPv6, you might want to do the following:

- Connect to the 6bone network
- Initialize a new interface for IPv6

- Remove IPv6 from an interface
- Create a configured tunnel
- Add addresses to or delete addresses from an interface
- Add or delete a default router
- Manually add a route for an on-link prefix
- Configure routing support in the kernel
- Edit the run-time configuration file (`/etc/rc.config`)
- Edit the router configuration file (`/etc/ip6rtrd.conf`)
- Tune the `ipv6` and `iptunnel` kernel subsystems

The following sections describe these tasks.

3.7.1 Connecting to the 6bone Network

To connect to the 6bone network, choose a 6bone point that appears to be reasonably adjacent to your normal IPv4 paths into the Internet. The 6bone Web site at <http://www.6bone.net> contains information on how to join the 6bone network and how to find an attachment point.

If you want to connect to the 6bone network through the Compaq Palo Alto, California site either before or after you configure IPv6 on your host or router, complete the following steps:

1. Register your IPv4 tunnel by sending the IPv4 address of your router to the following address:

```
gw-6bone@pa.dec.com
```

2. Wait for confirmation that support for your tunnel is configured at Compaq. Compaq will provide an IPv6 global address prefix for you to use at your site and the IPv4 address of the Compaq Palo Alto router.
3. Configure your tunnel by running the `ip6_setup` utility. See Section 3.6.1 for host configuration and Section 3.6.2 for router configuration. Alternatively, you could run the `iptunnel` command (see Section 3.7.4).
4. Verify that your tunnel is operational by issuing the `ping` command to one of the following Compaq IPv6 nodes:

```
altavista.ipv6.digital.com
ftp.ipv6.digital.com
www.ipv6.digital.com
```

3.7.2 Initializing a New Interface for IPv6

In some cases, you might want to add a new interface card to your system or change an interface card from one type to another. After the new card is installed, you must initialize it for IPv6 operation. To initialize an interface, use the `ifconfig` command with the following syntax:

```
ifconfig device ipv6 up
```

For LAN interfaces, the `ifconfig` command creates the link-local address (FE80::) and starts Duplicate Address Detection.

For example, to initialize Ethernet interface `ee0` for use with IPv6, enter the following command:

```
# ifconfig ee0 ipv6 up
```

To initialize the loopback interface for use with IPv6, enter the following command:

```
# ifconfig lo0 ipv6 up
```

To initialize the automatic tunnel interface, enter the following command:

```
# ifconfig tun0 ipv6 up
```

This chooses one of the system's IPv4 addresses for use as the tunnel endpoint.

If you are adding the interface card permanently, use the `ip6_setup` utility.

3.7.2.1 Setting the IPv6 Interface Identifier

You can set the IPv6 interface ID at the same time you initialize an interface by using the `ifconfig` command with the `ip6interfaceid` parameter. For example, to initialize Ethernet interface `ee0` for use with IPv6 and set its interface ID to the 64-bit value `0x0123456789abcdef`, enter the following command:

```
# ifconfig ee0 ip6interfaceid ::0123:4567:89ab:cdef ipv6 up
```

Although the interface ID is expressed in standard IPv6 address format, only the low order 64 bits are used.

3.7.3 Removing IPv6 from an Interface

Removing IPv6 from an interface removes the IPv6 configuration associated with the interface, including all IPv6 addresses and IPv6 routes through the interface. To remove IPv6 from an interface, use the `ifconfig` command with the following syntax:

```
ifconfig device -ipv6
```

For example, to remove IPv6 from Ethernet interface `ee0`, enter the following command:

```
# ifconfig ee0 -ipv6
```

3.7.4 Creating a Configured Tunnel

To create a configured (manual) tunnel, use the `/usr/sbin/iptunnel` command with the following syntax:

```
iptunnel create remote-tunnel-endpoint [local-tunnel-endpoint]
```

For example, to create a tunnel to the remote system `16.20.136.47`, enter the following command:

```
# iptunnel create 16.20.136.47
```

To initialize the tunnel for IPv6 operation, enter the following command:

```
# ifconfig ipt0 ipv6 up
```

If you want this change to be permanent, use the `ip6_setup` utility.

3.7.5 Adding an Address to an Interface

To add or assign an IPv6 prefix to an interface and to direct the kernel to automatically append the interface identifier, use the `ifconfig` command with the following syntax:

```
ifconfig interface-name inet6 ip6prefix prefix
```

The following command assigns the prefix `3ffe:1200:4112:2::/64` to interface `ln0` (the interface ID is `0a00:2bff:fe12:3456`).

As a result of this command, the address on the interface is `3ffe:1200:4112:2:0a00:2bff:fe12:3456`.

```
# ifconfig ln0 inet6 ip6prefix 3ffe:1200:4112:2::/64
```

The `ip6prefix` parameter directs the kernel to automatically append the interface identifier to the address prefix.

To add or assign a full IPv6 address to an interface manually, use the `ifconfig` command with the following syntax:

```
ifconfig interface-name inet6 address
```

The following command assigns the address `3ffe:1200:4112:2::1` to interface `ee0`:

```
# ifconfig ee0 inet6 3ffe:1200:4112:2::1
```

Note

For IPv6 hosts, the `nd6hostd` daemon configures interface prefixes automatically, depending on the contents of router advertisements.

For IPv6 routers, the `ip6rtrd` daemon configures interface prefixes automatically, depending on the contents of the `/etc/ip6rtrd.conf` file.

3.7.6 Deleting an Address from an Interface

To delete an IPv6 address from an interface manually, use the `ifconfig` command with the following syntax:

```
ifconfig interface-name inet6 delete address
```

For example:

```
# ifconfig ee0 inet6 delete 3ffe:1200:4112:2::1
```

3.7.7 Adding or Deleting a Default Router

To add a default router, use the `route` utility with the following syntax:

```
route add -inet6 default router-address -dev interface
```

For example:

```
# route add -inet6 default fe80::0a00:2bff:fe12:3456 -dev ee0
```

To delete a default router, use the `route` utility with the following syntax:

```
route delete -inet6 default router-address -dev interface
```

For example:

```
# route delete -inet6 default fe80::0a00:2bff:fe12:3456 -dev ee0
```

Note

For IPv6 hosts, the `nd6hostd` daemon performs the add and delete router operations automatically, depending on the contents of router advertisements.

3.7.8 Manually Adding a Route for an On-Link Prefix

After you manually add an address and prefix to an interface, you can also add a static route so that traffic to other nodes with the same prefix is sent directly to the destination rather than through a router. For example, if the

prefix `3ffe:1200:4112:5::/64` was added to an Ethernet interface, which was initialized with the link-local address `fe80::0a00:2bff:fe12:3456`, the following command adds a route to neighboring nodes with the same prefix:

```
# route add -inet6 3ffe:1200:4112:5::/64 fe80::0a00:2bff:fe12:3456 -interface
```

This command specifies that destinations with prefix `3ffe:1200:4112:5::/64` are reachable through the interface with address `fe80::0a00:2bff:fe12:3456`. In other words, `3ffe:1200:4112:5::/64` is an on-link prefix.

Note

For IPv6 hosts, the `nd6hostd` daemon automatically adds on-link prefixes, based on the contents of router advertisements.

3.7.9 Configuring Routing Support in the Kernel

Before configuring a router, you must enable forwarding by setting the `ipv6forwarding` and `ipv6router` attributes of the `ipv6` kernel subsystem to 1. You set these attributes by entering the following `sysconfig` commands:

```
# /sbin/sysconfig -r ipv6 ipv6forwarding=1
# /sbin/sysconfig -r ipv6 ipv6router=1
```

These commands are typically executed by the system startup scripts on nodes configured as IPv6 routers.

3.7.10 Editing the Run-Time Configuration File

After you configure the system, either as an IPv6 host or an IPv6 router, the `/etc/rc.config` file contains information used by the system startup procedures to start IPv6. You can modify this file as appropriate for your configuration by using the `rcmgr` command. The following variables are used by IPv6:

`IPV6="yes|no"`

If set to `yes`, starts IPv6 during system startup.

`IP6DEV_n="dev"`

Specifies an IPv6 device name. The device name must be in the `rc.config` file. The `n` value is an integer number that starts at 0 and increments sequentially for each device.

IP6IFCONFIG_*n*_*m*="*string*"

Specifies options and parameters to use on an `ifconfig` command line during system startup. The *n* value is an integer number that corresponds to the number in the `IP6DEV_n` variable. The *m* value is an integer that starts at 0 and increments sequentially for each `ifconfig` line needed for each device.

NUM_IP6CONFIG="*number*"

Specifies the number of IPv6 devices configured.

IP6ROUTER="yes|no"

If set to yes, configures the node as an IPv6 router. Otherwise, configures the node as a host.

IP6RTRD="yes|no"

If set to yes, starts the IPv6 router daemon, `ip6rtrd`, during IPv6 startup.

IP6RTRD_FLAGS="*string*"

Specifies a string of options and parameters to use in starting the `ip6rtrd` daemon.

ND6HOSTD="yes|no"

If set to yes, starts the IPv6 host daemon, `nd6hostd`, during IPv6 startup.

ND6HOSTD_FLAGS="*string*"

Specifies a string of options and parameters to use in starting the `nd6hostd` daemon.

IPTUNNEL_*n*="*string*"

Specifies a string of options and parameters to use to create a configured tunnel during system startup. This variable is used only when the device specified with the `IP6DEV_n` variable is a configured tunnel (for example, `ipt0`).

Example 3–1 shows sample variables for an IPv6 host in the `/etc/rc.config` file.

Example 3–1: Sample IPv6 Host Configuration Variables

```
IPV6="yes"
IP6DEV_0="tu0"
IP6IFCONFIG_0_0="ipv6 up"
IP6DEV_1="tun0"
IP6IFCONFIG_1_0="ipv6 up"
NUM_IP6CONFIG=2
IP6ROUTER="no"
IP6RTRD="no"
IP6RTRD_FLAGS=""
ND6HOSTD="yes"
ND6HOSTD_FLAGS=" -u -n host1.corp.com"
```

Example 3–2 shows sample variables for an IPv6 router in the `/etc/rc.config` file.

Example 3–2: Sample IPv6 Router Configuration Variables

```
IPV6="yes"
IP6DEV_0="tu0"
IP6IFCONFIG_0_0="ipv6 up"
IP6DEV_1="tu1"
IP6IFCONFIG_1_0="ipv6 up"
NUM_IP6CONFIG=2
IP6ROUTER="yes"
IP6RTRD="yes"
IP6RTRD_FLAGS="/etc/ip6rtrd.conf"
ND6HOSTD="no"
ND6HOSTD_FLAGS=""
```

3.7.11 Editing the Router Configuration File

After you configure the system as an IPv6 router, the `ip6rtrd` daemon sends out periodic router advertisements for the following reasons:

- To advertise itself as a potential default router for IPv6 traffic. The IPv6 nodes on the link receive these advertisements as part of their Neighbor Discovery processing.
- To advertise an IPv6 address prefix, in which case IPv6 nodes on the link perform address autoconfiguration.

The `/etc/ip6rtrd.conf` file contains the configuration data needed to send Router Advertisement messages. This file is created when `ip6_setup` is run, if the system is configured as a router. The link interface and advertised prefix are inserted, and other default values are used. You can

modify this file as appropriate for your network, for example, when using multiple prefix values. See `ip6rtrd.conf(4)` for more information.

Example 3–3 is a sample configuration file.

Example 3–3: Sample `ip6rtrd.conf` File

```
#
# Sample ip6rtrd configuration file
#
interface tu0 {
    MaxRtrAdvInterval 600
    MinRtrAdvInterval 200
    AdvManagedFlag 0
    AdvOtherConfigFlag 0
    AdvLinkMTU 1500
    AdvReachableTime 0
    AdvRetransTimer 0
    AdvCurHopLimit 64
    AdvDefaultLifetime 1800
    Prefix dec:1::/64 {
        AdvValidLifetime 1200
        AdvPreferredLifetime 600
        AdvOnLinkFlag 1
        AdvAutonomousFlag 1
    }
}
```

3.7.12 Tuning the Kernel Subsystems

You can use either the `sysconfig` utility or `dxkerneltuner` utility to tune the IPv6 subsystems. See `sys_attrs_ipv6(5)` and `sys_attrs_ip tunnel(5)` for information on tuning the IPv6 subsystem and IP tunnel subsystem, respectively.

3.8 IPv6 Daemon Log Files

The `nd6hostd` and `ip6rtrd` daemons log informational and severe events in the `/var/adm/syslog.dated/date/daemon.log` file. You can view the contents of this message file by using the Event Viewer that is part of the SysMan Menu utility. See Section 9.8 for more information about the Event Viewer.

By default, the daemons do not log debug information. To enable logging of debug information for the `nd6hostd` daemon, issue the following commands:

```
# rcmgr set ND6HOSTD_FLAGS "-d -l /usr/tmp/nd6hostd.log"
# /usr/sbin/rcinet restart inet6
```

To enable logging of debug information for the `ip6rtrd` daemon, issue the following commands:

```
# rcmgr set IP6RTRD_FLAGS "-d -l /usr/tmp/ip6rtrd.log"  
# /usr/sbin/rcinet restart inet6
```

Asynchronous Transfer Mode

Asynchronous Transfer Mode (ATM) networks provide the following capabilities:

- Speeds from 25 M/bps to 622 M/bps or greater through cell-switching.
- Multiple qualities of service.
- Connection-oriented interconnection with resource reservation for individual connections. These connections might be for conversations between two applications or for a connection over which many conversations between many applications and protocols are multiplexed.

ATM networks provide the high speed and the low latency (switched, full duplex network infrastructure) that applications, particularly those running on local area networks, require.

This chapter describes:

- The ATM network environment (Section 4.1)
- How to plan for your ATM configuration (Section 4.2)
- How to configure the ATM subsystem (Section 4.3)
- How to manage the ATM subsystem (Section 4.4)

See the *Asynchronous Transfer Mode* manual for information about writing device drivers and kernel modules for ATM. For troubleshooting information, see Section 8.5.

4.1 ATM Environment

An ATM network consists of the following:

- Switch

A specialized system that maintains a list of virtual channel identifiers (VCIs) and virtual path identifiers (VPIs), connects one end system to another, and forwards or switches ATM cells from one end system to another based on the VCI/VPI information contained in the cell.

- End system

A system physically connected to a switch that communicates with other end systems through the switch.

In the operating system's ATM environment, the following configurations are possible:

- Classical Internet Protocol (CLIP)
- Local Area Network (LAN) emulation
- IP switching

The following sections describe each of these configurations and the roles of systems in each.

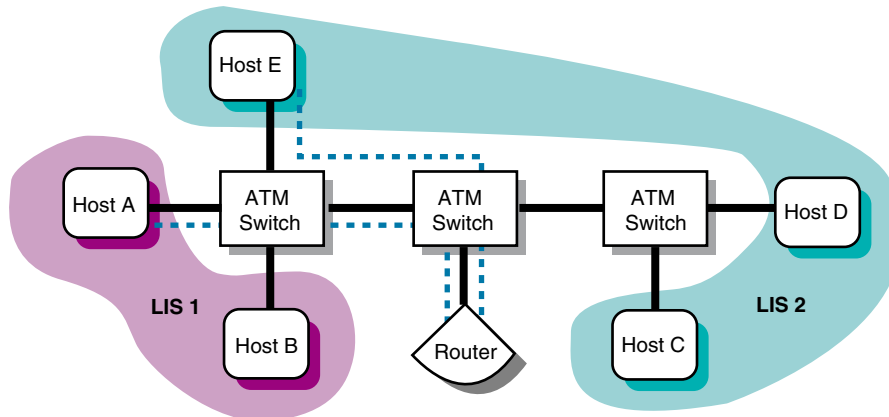
4.1.1 Classical IP Environment

The Classical IP environment, as described in RFC 1577, provides a basic means for carrying unicast IP traffic over ATM networks. In this environment, hosts that can communicate with each other are grouped into a Logical IP Subnetwork (LIS). An ATM network can contain multiple LISs. In a LIS, all hosts and routers have the following requirements:

- Have the same IP network or subnetwork number and mask.
- Are directly connected to the ATM network.
- Access members outside the LIS through a router.
- For switched virtual circuits (SVCs), use Address Resolution Protocol (ARP) to resolve IP protocol addresses to ATM hardware addresses. For SVCs and permanent virtual circuits (PVCs), use Inverse ARP to resolve ATM hardware addresses to IP protocol addresses.
- Can communicate with all other members in the same LIS (mesh topology).

Figure 4–1 shows an ATM network with two LISs. Host A and Host B are members of LIS 1; Host C, Host D, and Host E are members of LIS 2. The figure also shows a virtual circuit (VC) between Host A and the router and between Host E and the router. Although these hosts are connected to the same switch and might establish a VC for communications between one another, they cannot because all communications to a member of another LIS must go through a router.

Figure 4–1: Classical IP over an ATM Network



ZK-1307U-AI

4.1.2 LAN Emulation Environment

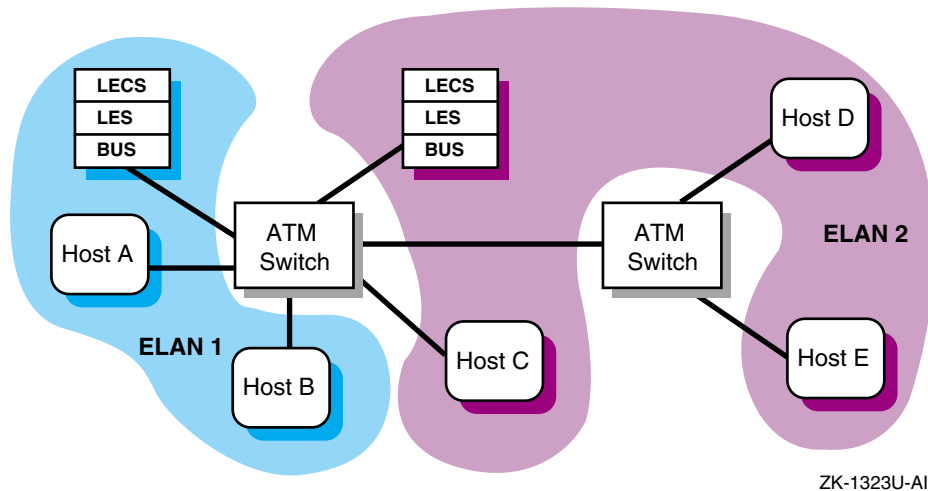
The LAN Emulation (LANE) environment, as defined by the ATM Forum, groups hosts into an entity called an emulated LAN (ELAN). A LANE environment has the following characteristics:

- Identifies hosts through their 48-bit media access control (MAC) addresses
- Supports multicast and broadcast services either through point-to-multipoint connections or through a multicast server, unlike the Classical IP environment
- Supports any protocol that uses an IEEE broadcast LAN

In addition, LANE interfaces (`elan`) are supported by NetRAIN. See `nr(7)` for more information.

Figure 4–2 shows an ATM network with two emulated LANs. Host A and Host B are LAN Emulation Clients (LECs) on ELAN 1. Host C, Host D, and Host E are LECs on ELAN 2. The LECS (LAN Emulation Configuration Server), the LES (LAN Emulation Server), and the BUS (Broadcast and Utility Server) are depicted as two separate systems, although these server functions are typically resident on an ATM switch.

Figure 4–2: Emulated LAN over an ATM Network



4.1.3 IP Switching

Note

IP switching support is provided for backward compatibility only; it will be retired in a future release. Do not use it to develop new applications.

The IP switching environment consists of one or more hosts connected to an IP switch. Each host is connected to the IP switch through a point-to-point physical connection, with each physical connection as a separate subnet. Communication between the host and the IP switch occurs over dynamically created PVCs.

The IP switch is a typical ATM switch with added IP controller software that performs IP routing and IP traffic classification functions. In this environment, a series of packets moving from one host to another with the same protocol type, type of service, and other characteristics indicated in the packet header is called a **flow**. When the IP controller identifies a flow that is of long duration, it instructs the ATM switch to make the appropriate hardware connections and to forward the ATM cells directly to the destination, bypassing the IP controller. This increases throughput at the switch and throughout the network.

The operating system's IP switching implementation is based on the Ipsilon Networks, Inc. reference model and has the following characteristics:

- Supports IP traffic only

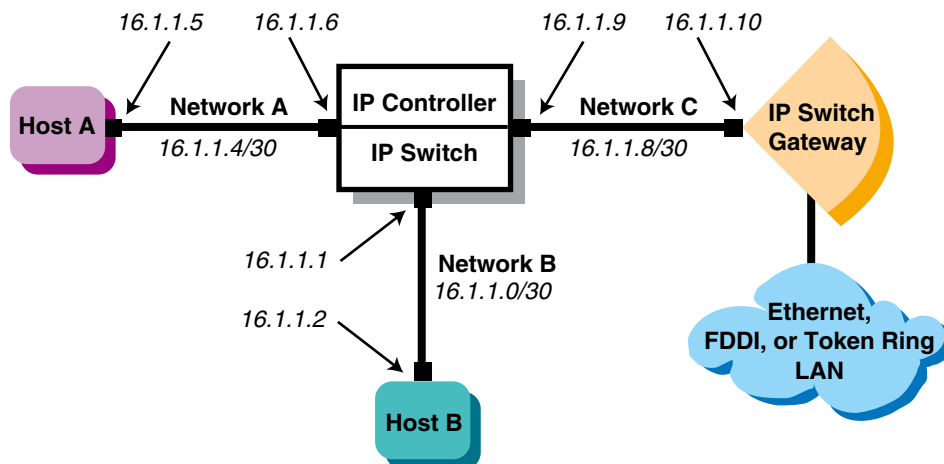
- Supports multicast and broadcast services
- Does not require systems to function as ARP servers or multicast servers
- Uses the Ipsilon Flow Management Protocol (IFMP) to exchange control information with the IP switch
- Does not require that ATM Forum signaling (options UNI3X) be configured on the system
- Requires fewer configuration steps than Classical IP and LAN emulation

IP switching over ATM has the following restrictions:

- Only one IP switching interface (`ips`) per host is supported.
- If using a driver for IP switching, you cannot use other ATM protocols on that driver.
- The `tcpdump` and `packetfilter` utilities are not supported on an `ips` interface.

Figure 4–3 shows a simple ATM network with an IP switch, IP switch gateway, some hosts, and a legacy LAN network. Host A (16.1.1.5), Host B (16.1.1.2), and the IP switch gateway (16.1.1.10) are on separate subnets (16.1.1.4/30, 16.1.1.0/30, and 16.1.1.8/30). The IP switch gateway runs a routing protocol and advertises routes to other subnets to hosts on the legacy LAN.

Figure 4–3: IP Switching over an ATM Network



ZK-1305U-AI

For the IP switching subnetworks, the recommended network mask length is 30 bits. This allows for two bits for each host address, one bit for the subnetwork address, and one bit for the broadcast address. Using large

netmasks helps to conserve IP address space on subnetworks that have a few hosts attached.

4.2 Planning ATM

This section describes the tasks you need to complete before configuring the ATM software.

4.2.1 Verifying That the ATM Subsets Are Installed

Verify that the ATM subsets are installed by entering the following command:

```
# setld -i | grep ATM
```

If all of the subsets are not installed, install them by using the `setld` command. For more information on installing subsets, see `setld(8)` or the *Installation Guide*.

Note

You do not have to install the `OSFATMBINOBJECT` subset.

4.2.2 Configuring ATM into the Kernel

After you install the ATM subsets, verify that the ATM support you require is in the kernel by issuing the following command:

```
# sysconfig -q atm
```

If `atm:` is not displayed, log in as superuser and complete the following steps:

1. Build a new kernel by issuing the `doconfig` command. If you are unfamiliar with rebuilding the kernel, see the *System Administration* manual.
2. When prompted, select one or more of the kernel options described in Table 4-1.

Note

If the ATM hardware is already installed, `options ATM` is automatically selected as a mandatory option.

3. Reboot your system with the new kernel by issuing the following command:

```
# shutdown -r now
```

This command immediately shuts down and automatically reboots the system.

Table 4–1: ATM Kernel Options

Option	Purpose
<code>options ATM</code>	For base ATM support (required)
<code>options UNI3X</code>	For ATM Forum signaling with either LANE or Classical IP
<code>options ATMILMI3X</code>	For ATM Forum Integrated Layer Management Interface (ILMI) support
<code>options ATMIP</code>	For Classical IP services
<code>options LANE</code>	For ATM Forum LAN Emulation (LANE)
<code>options ATMIFMP</code>	For IP switching

4.2.3 Preparing for the Configuration

After verifying that ATM support is in the kernel, you can configure ATM. To configure ATM, you need to configure an ATM adapter and one or more of the following interfaces:

- A Classical IP logical interface
- A LAN Emulation logical interface
- An IP switching logical interface

The type of information you need depends on the environment you want to set up and use.

4.2.3.1 Adapter Information

Figure 4–4 shows the ATM Setup Worksheet. The following sections explain the information you need to record on this worksheet. If you are viewing this manual on line, you can use the print feature to print a copy of the worksheet.

Figure 4–4: ATM Setup Worksheet

ATM Setup Worksheet	
Adapter name:	_____
ROM ESIs:	_____
More ESIs:	_____
Network layer:	SONET <input type="checkbox"/> SDH <input type="checkbox"/>
Flow control:	Yes <input type="checkbox"/> No <input type="checkbox"/>
ILMI:	Yes <input type="checkbox"/> No <input type="checkbox"/>
Signaling:	Yes <input type="checkbox"/> No <input type="checkbox"/>
VC accounting:	Yes <input type="checkbox"/> No <input type="checkbox"/>
UNI version:	3.0 <input type="checkbox"/> 3.1 <input type="checkbox"/>

Adapter name

The device names of the ATM network interfaces. For example, the `lta` network interface.

ROM ESIs

The ROM end system identifier (ESI) addresses of the adapter that you want to register with the system and the local switch. If you want to register all of the adapter's ROM ESI addresses, leave this blank.

Depending on the number of address prefixes assigned by the switch, you can create one or more ATM addresses. The driver can control up to 64 ROM ESI addresses, though adapters generally have only a few ROM ESI addresses.

More ESIs

Additional ESI addresses that you want to register with the system and the local switch. An ESI address has twelve hexadecimal digits.

Network layer

If you want to enable Synchronous Optical Network (SONET), on the adapter, check `SONET`. If you want to enable Synchronous Digital Hierarchy (SDH) mode on an ATM adapter that supports both SONET and SDH physical interfaces, check `SDH`.

Flow control

If you want to enable vendor-specific flow control on the adapter, check `Yes`; otherwise, check `No`. The adapter must support this type of flow control. Compaq adapters and switches support `FLOWmaster` vendor flow control.

ILMI

If you want to enable the Integrated Layer Management Interface (ILMI) on the adapter, check Yes; otherwise, check No. You must enable ILMI when using Classical IP over switched virtual circuits (SVCs).

Signaling

If you want to enable signaling on the adapter, check Yes; otherwise, check No. You must enable signaling when using Classical IP over SVCs.

VC accounting (signaling only)

If you want to enable logging of virtual circuit (VC) releases, check Yes; otherwise, check No.

UNI version (signaling only)

The signaling version to use on the adapter. If you want to use User-Network Interface (UNI) Version 3.0, check 3.0. If you want to use UNI Version 3.1, check 3.1. The default is 3.0.

4.2.3.2 Classical IP Information

Figure 4-5 shows the ATM Classical IP Worksheet. The following sections explain the information you need to record on this worksheet. If you are viewing this manual on line, you can use the print feature to print a copy of the worksheet.

ATM address (ARP client only)

The ATM address of the ATM ARP server, either a host name or alias that appears in the `/etc/atmhosts` file or a 40-digit ATM End System Address (AESA) with selector byte. The ARP server must also be on the ATM network.

Note

The ATM Forum now calls an NSAP-style address an AESA.

IP address (ARP client only)

The IP address of the ATM ARP server machine.

VCI (PVCs only)

The virtual channel identifier (VCI) for the PVC.

VPI (PVCs only)

The virtual path identifier (VPI) for the PVC.

Remote Classical IP (PVCs only)

If the remote host supports Classical IP as defined in RFC 1577, check Yes; otherwise, check No.

Remote IP address (PVCs only)

If the remote host does not support Classical IP, enter the remote host's IP address.

4.2.3.3 LAN Emulation Information

Figure 4–6 shows the ATM LAN Emulation Worksheet. The following sections explain the information you need to record on this worksheet. If you are viewing this manual on line, you can use the print feature to print a copy of the worksheet.

Mode

If you want to contact the default LAN Emulation Configuration Server (LECS), check Default LECS. The LEC contacts the LECS by using an ILMI MIB request to obtain the LECS address. If the request is unsuccessful, the LEC uses the well-known address for the LECS. If you want to contact a specific LECS, check Specific LECS. In either case, the LEC contacts a LECS to obtain a LES address.

If you want to contact the LES directly, check LES.

LECS name

The ATM address of the LECS, either a host name or alias that appears in the `/etc/atmhosts` file or a 40-digit ATM AESA address with selector byte. If you want to contact a specific LECS, enter the LECS address; you can specify up to four.

LES name

The ATM address of the LES, either a host name or alias that appears in the `/etc/atmhosts` file or a 40-digit ATM AESA address with selector byte. If you want the LEC to go directly to the LES and bypass the configuration phase, enter the LES address.

MTU size

The maximum transmission unit (MTU) size. The following MTU sizes are supported: 1516, 4544, 9234, and 18190. When specified with a virtual LAN name, the emulated LAN must already be configured on the ATM switch to support the specified MTU size. If it is not configured for the specified MTU size, the request is ignored.

4.2.3.4 IP Switching Information

Figure 4–7 shows the ATM IP Switching Worksheet. The following sections explain the information you need to record on this worksheet. If you are viewing this manual on line, you can use the print feature to print a copy of the worksheet.

ips number

The IP switching (`ips`) interface number. If you are using multiple adapters, each adapter is assigned a separate interface number.

SNAP VCI

The Virtual Channel Identifier (VCI) number that Ipsilon Flow Management Protocol (IFMP) uses as the default Subnetwork Attachment Point (SNAP) VCI. The default VCI is 15. This number must match the VCI number that IFMP uses on the destination host or switch associated with the point-to-point interface.

Routing

The method you use to update your internal routing tables. If you use the `gated` daemon, check `gated`. If you use the `routed` daemon, check `routed`. If you use static routes, check `static routes`.

Destination (static routes only)

The IP address of the destination subnetwork.

Gateway (static routes only)

The IP address of the IP controller on the IP switch.

Netmask (static routes only)

The netmask for the destination subnetwork.

4.3 Configuring ATM

After you complete the required ATM planning and you install the appropriate ATM hardware, you can configure the ATM software. Use the ATM Configuration application of the Common Desktop Environment (CDE) Application Manager to configure ATM. You can configure the following:

- ATM adapter
- Classical IP
- LAN Emulation
- IP Switching

To use the ATM Configuration application, invoke the SysMan Menu application as specified in Section 1.2.1, then see Section 4.3.1 for further instructions.

Optionally, you can use the `atmsetup` script that was available in previous releases by executing the `atmsetup -old` command. See the online help and `atmsetup(8)` for more information.

4.3.1 Configuring an ATM Adapter

Before you can configure ATM logical interfaces, you must configure an adapter. To configure an ATM adapter, do the following:

1. From the SysMan Menu, select Networking→Basic Network Services→Set up Asynchronous Transfer Mode (ATM) to display the ATM Configuration main window.

Alternatively, enter the following command on a command line:

```
# /usr/sbin/sysman atm
```

Or, enter:

```
# atmsetup
```

The ATM Configuration main window displays the unconfigured adapters, configured adapters, and configured logical interfaces.

2. Select an adapter from the Unconfigured Adapters field.
3. Select Configure. The Configure/Modify Adapter dialog box is displayed.
4. If you do not want to register all ROM Endpoint System Identifiers (ESIs) for the adapter, select Register ROM ESI. By default, all of the adapter's ROM ESI addresses are registered.
5. If you want to register additional ESIs (called soft ESIs) for the adapter, select Register Soft ESI.
6. If you want to set transmit Constant Bit Rate (CBR) or pacing options for the adapter, select Set CBR/Pacing Options. The Set CBR/Pacing Options dialog box is displayed. When you are finished, select OK to close the dialog box and save the changes.
7. Indicate the type of network physical layer you want the adapter to support: SONET or SDH.
8. Indicate whether you want to enable flow control (FLOWmaster) on the adapter.
9. Indicate whether you want to enable Integrated Local Management Interface (ILMI) on the adapter.
10. Indicate whether you want to enable signaling on the adapter.
11. Indicate whether you want to enable the logging of all virtual circuit (VC) releases.
12. Select a User-Network Interface (UNI) version.

13. Select OK to accept the configuration and close the Configure/Modify Adapter dialog box. You can now configure an ATM logical interface.

You can also modify your adapter configuration. See the online help and `atmsetup(8)` for more information.

4.3.2 Configuring Classical IP

Before you configure Classical IP, you must configure an ATM adapter. Configuring Classical IP on your host consists of the following steps:

1. Creating PVC mappings on your ATM switch (PVCs only)
2. Adding servers to the `atmhosts` file
3. Adding hosts to the `hosts` database
4. Running the ATM Configuration application
5. Configuring the Classical IP logical interface
6. Adding static routes (SVCs only)
7. Verifying the PVC Configuration (PVCs only)

The following sections describe these steps.

4.3.2.1 Creating PVC Mappings on Your ATM Switch

If you are going to use PVCs and your environment requires an ATM switch, you need to create PVC mappings on the switch. The method for creating these mappings depends on the type of ATM switch you use. See your ATM switch documentation for more information.

4.3.2.2 Adding Servers to the `atmhosts` File

You edit the `/etc/atmhosts` file to add the address of the ATM ARP server on your ATM network. The `/etc/atmhosts` file contains mappings of ATM host names to ATM hardware addresses. This file can also contain ATM ESIs and AESAs for specific services on the ATM network. Putting entries in this file enables you to specify the address or service by name instead of specifying a long hexadecimal string.

Entries in the `/etc/atmhosts` file can be one of the following:

- A comment, denoted by a pound sign (#) as the first character
- An address specification

The address specification is similar to that of IP addresses in the `/etc/hosts` file, and has the following format:

```
atm_addr hostname [ alias ... ]
```

The `atm_addr` parameter can consist of ESIs or AESAs.

The following table lists the address type and the number of hexadecimal address digits required for each type:

Address Type	Number of Address Digits
ESI	Twelve hexadecimal digits
AESA	Thirty-eight hexadecimal digits
AESA with selector byte	Forty hexadecimal digits

The `hostname` parameter can contain any printable character.

The following example shows entries in the `/etc/atmhosts` file:

```
08002b2fe740 myhost.esi [1]
47840f01020300002122313208002b2fe740 myhost [2]
47840f01020300002122313208002b2fe7403a myhost.ip [3]
```

- [1] Specifies an ESI to use in registering `myhost` with the switch.
- [2] Specifies the AESA of `myhost`. This is the network prefix and the ESI, and is the address that the network recognizes.
- [3] Specifies the AESA with selector byte of a service on `myhost` for the operating system's implementation of RFC 1577, *Classical IP and ARP over ATM*.

Note

By default, the `atmhosts` file contains an entry for PVCs. Do not delete or modify this entry.

4.3.2.3 Adding Hosts to the hosts Database

You add the IP addresses for all ATM hosts that will be on any Logical IP Subnet (LIS) to which the host will connect to the `hosts` database. Make sure you have the IP addresses for the local host and the ATM ARP server. Depending on your environment, host names and addresses can be in the local `/etc/hosts` file or in one of the files distributed with DNS or NIS.

You can enter these IP addresses in the `/etc/hosts` file either by editing the file itself or by running the SysMan Menu application of the CDE Application Manager. See Section 2.3.7 for more information.

4.3.2.4 Running the ATM Configuration Application

To configure Classical IP on your system, do the following:

1. From the SysMan Menu, select Networking→Basic Network Services→Set up Asynchronous Transfer Mode (ATM) to display the ATM Configuration main window.

Alternatively, enter the following command on a command line:

```
# /usr/sbin/sysman atm
```

Or, enter:

```
# atmsetup
```

The ATM Configuration main window displays the unconfigured adapters, configured adapters, and configured logical interfaces.

2. Select Add. The Add Interfaces dialog box is displayed.
3. Select Classical IP. The Add Interfaces dialog box closes. The Add/Modify Classical IP Interface dialog box is displayed.
4. Choose the adapter on which you want to add a Classical IP logical interface.
5. If you do not want to use the default logical interface number, enter a different number.
6. Indicate whether your system is to act as an ARP client or an ARP server.
7. If the system is to be an ARP client, enter the ARP server's ATM address or alias. Then, enter the ARP server's IP address.
8. If you are going to specify PVCs for the logical interface, select PVCs. The Add/Modify PVC dialog box is displayed. Do the following:
 - a. Enter a virtual path identifier (VPI) for the virtual circuit.
 - b. Enter a virtual channel identifier (VCI) for the virtual circuit.
 - c. Indicate whether the remote host entity supports Classical IP as defined in RFC 1577.
 - d. If the remote host does not support Classical IP, enter the remote host's IP address.
 - e. Select OK to accept the configuration and close the Add/Modify PVC dialog box.
9. Select OK to close the Add/Modify Classical IP Interface dialog box.
10. Select OK in the ATM Configuration main window to save the changes. If no ATM interface exists on the system, the Start ATM Now dialog box is displayed. If you want to start the ATM subsystem, select OK;

otherwise, select No. If you select No, you must reboot the system to start the ATM subsystem.

If an ATM interface exists on the system, the Reboot Required dialog box is displayed. Select OK to acknowledge the message. You must reboot the system to start the ATM subsystem.

You can also modify your adapter configuration. See the online help and `atmsetup(8)` for more information.

4.3.2.5 Configuring the Classical IP Logical Interface

After you run the ATM Configuration application and start the ATM components (either from within the application or by rebooting the system), you can configure the Classical IP (`lis`) interface. To configure the `lis` interface, see Section 2.3.1.

4.3.2.6 Adding Static Routes (SVC only)

Depending on your network topology and the number and configuration of logical IP subnetworks (LISs) in your network, you might need to add static routes to other hosts if you want a connection to a host that is on another LIS subnet. To add a static route to the routing tables, see Section 2.3.6.

4.3.2.7 Verifying the PVC Configuration (PVCs only)

After the PVC is configured, verify the configuration by issuing the `atmarp -a` command. Output similar to the following appears if the PVC is configured:

```
# atmarp -a
Number of entries : 1

IP Address :   atm66 (16.142.128.66)
ATM Address : PVC
Flags :       Complete Permanent
VCs :        vpi    vci    VC Type
              ---    ---    -
              0      999    PVC
```

4.3.3 Configuring LAN Emulation

Configuring LAN emulation on your host consists of the following steps:

1. Adding servers to the `atmhosts` file
2. Adding hosts to the `hosts` database
3. Running the ATM Configuration application

4. Configuring the LAN Emulation logical interfaces

The following sections describe these steps.

4.3.3.1 Adding Servers to the `atmhosts` File

You edit the `/etc/atmhosts` file only if you want to specify a LAN Emulation Server (LES) address or LAN Emulation Configuration Server (LECS) addresses on your ATM network. The `/etc/atmhosts` file contains mappings of ATM host names to ATM hardware addresses. This file can also contain ATM ESIs and AESAs for specific services on the ATM network.

See Section 4.3.2.2 for more information on editing the `/etc/atmhosts` file.

4.3.3.2 Adding Hosts to the `hosts` Database

You add the IP addresses for all ATM hosts that will be on any emulated LAN (ELAN) to which the host will connect to the `hosts` database. Make sure you have the IP addresses for the local host. Depending on your environment, host names and addresses can be in the local `/etc/hosts` file or in one of the files distributed with DNS or NIS.

You can enter these IP addresses in the `/etc/hosts` file either by editing the file itself or by running the SysMan Menu application of the CDE Application Manager. See Section 2.3.7 for more information.

4.3.3.3 Running the ATM Configuration Application

To configure LAN emulation on your system, do the following:

1. From the SysMan Menu, select Networking→Basic Network Services→Set up Asynchronous Transfer Mode (ATM) to display the ATM Configuration main window.

Alternatively, enter the following command on a command line:

```
# /usr/sbin/sysman atm
```

Or, enter:

```
# atmsetup
```

The ATM Configuration main window displays the unconfigured adapters, configured adapters, and configured logical interfaces.

2. Select Add. The Add Interfaces dialog box is displayed.
3. Select LAN Emulation. The Add Interfaces dialog box closes. The Add/Modify LAN Emulation Interface dialog box is displayed.
4. Choose the adapter on which you want to add a LAN Emulation logical interface.

5. If you do not want to use the default logical interface number, enter a different number.
6. If you want to join a specific emulated LAN, enter the name of the emulated LAN you want to join.
7. Choose the mode by which your system will be registered into the emulated LAN. If you choose to contact a specific LAN Emulation Configuration Server (LECS) (the second choice), also enter the LECS name or alias. If you choose to contact a LAN Emulation Server (LES) directly (the third choice), also enter the LES name or alias.
8. If you want to specify an MTU size other than the default 1516, choose another MTU size.
9. Select OK to close the Add/Modify LAN Emulation Interface dialog box.
10. Select OK in the ATM Configuration main window to save the changes. If no ATM interface exists on the system, the Start ATM Now dialog box is displayed. If you want to start the ATM subsystem, select OK; otherwise, select No. If you select No, you must reboot the system to start the ATM subsystem.

If an ATM interface exists on the system, the Reboot Required dialog box is displayed. Select OK to acknowledge the message. You must reboot the system to start the ATM subsystem.

Note

You can join an ELAN on an ATM switch only once for each adapter; do not join the same ELAN multiple times from the same adapter. If you want to join the same ELAN on the same switch, you must install another adapter and join the ELAN from it.

You can also modify your adapter configuration. See the online help and `atmsetup(8)` for more information.

4.3.3.4 Configuring the LAN Emulation Logical Interfaces

After you run ATM Configuration and start the ATM components (either from within the application or by rebooting the system), you configure the LAN Emulation (`elan`) interface. To configure the `elan` interface, see Section 2.3.1.

4.3.4 Configuring IP Switching

Configuring IP switching on your host consists of the following steps:

1. Adding IP addresses to the `hosts` file

2. Running the ATM Configuration application to create the IP Switching logical interface
3. Configuring the IP Switching logical interface
4. Adding routes to the routing table

The following sections describe these steps.

4.3.4.1 Adding IP Addresses to the hosts File

You edit the `/etc/hosts` file to add the IP addresses for each IP switching subnetwork to which the host will connect. For each subnet, add a pair of IP addresses for each end of the point-to-point link (host side and IP controller side), the IP address of the subnet, and the broadcast address of the subnet. For example, an `/etc/hosts` file for the configuration in Figure 4–3 is as follows:

```
# IP Switching subnet A
16.1.1.4  networka-net
16.1.1.5  hosta.corp.com          hosta          atm5
16.1.1.6  ipsctrlhosta.corp.com ipsctrlhosta  atm6
16.1.1.7  networka-broadcast
# IP Switching subnet B
16.1.1.0  networkb-net
16.1.1.1  ipsctrlhostb.corp.com ipsctrlhostb  atm1
16.1.1.2  hostb.corp.com       hostb          atm2
16.1.1.3  networkb-broadcast
# IP Switching subnet C
16.1.1.8  networkc-net
16.1.1.9  ipsctrlhostc.corp.com ipsctrlhostc  atm9
16.1.1.10 ipgwy.corp.com           ipgwy          atm10
16.1.1.11 networkc-broadcast
```

You can enter these IP addresses in the `/etc/hosts` file either by editing the file itself or by running the SysMan Menu application of the CDE Application Manager. See Section 2.3.7 for more information.

4.3.4.2 Running the ATM Configuration Application

Do the following to configure IP switching on your system:

1. From the SysMan Menu, select Networking→Basic Network Services→Set up Asynchronous Transfer Mode (ATM) to display the ATM Configuration main window.

Alternatively, enter the following command on a command line:

```
# /usr/sbin/sysman atm
```

Or, enter:

```
# atmsetup
```

The ATM Configuration main window displays the unconfigured adapters, configured adapters, and configured logical interfaces.

2. Select Add. The Add Interfaces dialog box is displayed.
3. Select IP Switching. The Add Interfaces dialog box closes. The Add/Modify IP Switching Interface dialog box is displayed.
4. Choose the adapter on which you want to add an IP Switching logical interface.
5. If you do not want to use the default logical interface number, enter a different number.
6. If you want to change the virtual channel identifier (VCI) information from the default, select Options. The Modify IP Switching Options dialog box is displayed. Do the following:
 - a. Enter a SNAP VCI value, if other than 15 (the default).

Note

This SNAP VCI number must match the VCI number that IFMP uses on the switch associated with the point-to-point interface.

- b. Enter a range of VCIs to use for transmitting and receiving connections.
 - c. Select OK to save the changes and close the Modify IP Switching Options dialog box.
7. Select OK to close the Add/Modify IP Switching Interface dialog box.
8. Select OK in the ATM Configuration main window to save the changes. If no ATM interface exists on the system, the Start ATM Now dialog box is displayed. If you want to start the ATM subsystem, select OK; otherwise, select No. If you select No, you must reboot the system to start the ATM subsystem.

If an ATM interface already exists on the system, the Reboot Required dialog box is displayed. Select OK to acknowledge the message. You must reboot the system to start the ATM subsystem.

You can also modify your adapter configuration. See the online help and `atmsetup(8)` for more information.

4.3.4.3 Configuring the IP Switching Logical Interfaces

After you run ATM Configuration and start the ATM components (either from within the application or by rebooting the system), you configure the IP Switching (`ips`) interface. To configure the `ips` interface, see Section 2.3.1.

4.3.4.4 Adding Routes

Depending on your network topology and the number of interfaces on your host, you might need to add routes to other hosts if your system has multiple interfaces and the default route is to another gateway on another network. Do either of the following:

- Run either the `gated` or the `routed` daemon to automatically update your system's routing tables.
- Add a static route to the routing tables for the destination network. Select Networking→Configuration→Static Routes from the SysMan Menu. This opens the Static Routes File dialog box. You need to specify the IP address of the destination subnetwork and address of the IP controller on the IP switch. For example, if you were configuring IP switching on Host A in Figure 4–3 and you wanted to route all traffic on all 16.1.1 networks through the IP switch, you would specify 16.1.1/24 as the destination address in Classless Inter-Domain Routing (CIDR) format and 16.1.1.6 as the gateway address.

Add entries for each additional network with which your system needs to communicate. See Section 2.3.6 for more information.

4.4 Managing the ATM Environment

Managing the ATM environment consists of managing the following components:

- ATM networking and displaying information about ATM networks
- Signaling module
- Classical IP environment
- LAN Emulation environment
- IP switching
- ATM subsystem messages

The following sections describe how to manage these components.

4.4.1 ATM Networking and Displaying Information About ATM Networks

To manage ATM networking and to display information about the ATM networks, you use the `atmconfig` command. The command controls only the base ATM modules and device drivers; it does not control specific convergence modules or signaling protocols. You can use the `atmconfig` command to do the following:

- Enable and disable device drivers
- Create and destroy PVCs
- Destroy SVCs
- Create and destroy ESIs
- Display the currently active VCs and driver status
- Process configuration batch files

See `atmconfig(8)` for more information.

4.4.2 Signaling Module

To manage ATM UNI signaling on the end system, you use the `atmsig` command. The `atmsig` command allows you to:

- Display state information about the signaling module
- Disable and enable the ILMI and signaling
- Read and modify the various timer values and statistics for Q.SAAL and Q.93B (2931)

The signaling module is associated with a specified interface at all times, which is identified by the driver name. If the interface is disabled, the signaling module is also disabled. The signaling module must be enabled again when the interface is brought back on line.

See `atmsig(8)` for more information.

4.4.3 Classical IP Environment

To manage Classical IP on an end system, you use the `atmarp` command. The `atmarp` command allows you to:

- Create a logical IP subnet (LIS) interface
- Create and delete entries in the ATM ARP table
- Display entries in the ATM ARP table
- Toggle the permanent flag for entries

- Display the local host's ATM configuration status
- Create and remove an association between an established VC and a remote IP entity that does not support Classical IP

See `atmarp(8)` for more information.

4.4.4 LAN Emulation Environment

Managing the LAN emulation environment consists of the following tasks:

- Managing LAN Emulation Clients (LECs)
- Displaying the LAN Emulation Address Resolution Protocol (LE-ARP) table

The following sections describe these tasks.

4.4.4.1 Managing LAN Emulation Clients

To manage LECs, you use the `atmelan` command. The `atmelan` command allows you to:

- Create and configure LECs as network interfaces
- Display counters, parameters, and the state of each LEC

See `atmelan(8)` for more information.

4.4.4.2 Displaying the LE-ARP Table

To display the LE-ARP table for each `elan` interface, you use the `learp` command. The command displays the address mappings for the emulated LAN. Each entry consists of the Media Access Control (MAC) address, state, ATM address, and flags. See `learp(8)` for more information.

4.4.5 IP Switching

To manage IP switching on an end system, you use the `atmifmp` command. The `atmifmp` command allows you to:

- Enable and disable IP switching
- Display IP switching configuration
- Display or clear IP switching statistics
- Display IP switching flow information

See `atmifmp(8)` for more information.

4.4.6 ATM Subsystem Messages

The ATM subsystem logs status and error messages in the `/var/adm/syslog.dated/date/kern.log` file. You can view the contents of this message file by using the Event Viewer that is part of the SysMan Menu utility. See Section 9.8 for more information about the Event Viewer.

By default, the ATM subsystem logs subsystem initialization messages, important state changes, and significant error conditions. To increase the message level displayed by all ATM subsystem components, enter the following command:

```
# sysconfig -r atm global_msg_level=2
```

You can also increase the message for individual subsystem components. For example, if you want to increase the message level for LANE to view session initialization information, enter the following command:

```
# sysconfig -r lane lane_msg_level=2
```

See `sys_attrs_atm(5)` for more information.

5

Dynamic Host Configuration Protocol

Dynamic Host Configuration Protocol (DHCP) enables you to centralize and automate IP address administration. Using a graphical application, you can configure several computers at once, ensuring that configurations are consistent and accurate. Even portable computers can be automatically configured each time they attach to the network.

This chapter describes:

- The DHCP implementation on Tru64 UNIX systems (Section 5.1)
- How to plan for your DHCP configuration (Section 5.2)
- How to configure a DHCP server by using the `xjoin` and SysMan Menu utilities (Section 5.3)
- How to manage DHCP client addressing (Section 5.4)

The implementation of DHCP in Tru64 UNIX is based on JOIN® Server Version 4.1 from JOIN Systems, Inc. For additional information about DHCP, see the `DHCP(7)` reference page and the *JOIN Server Administrator's Guide*. The latter is provided by JOIN Systems in HTML format, and it can be accessed by opening the following file with a web browser:

```
/usr/doc/join/TOC.html
```

For troubleshooting information, see Section 8.6.

Note

Starting with Tru64 UNIX Version 4.0F, DHCP database files were stored in a new format that is incompatible with older formats. An online document explains the reasons behind this change, lists the files that are affected, and provides instructions for converting the files to the new format. The document, `README-DB237`, and conversion utility, `conv185-237`, are located in the `/etc/join` directory.

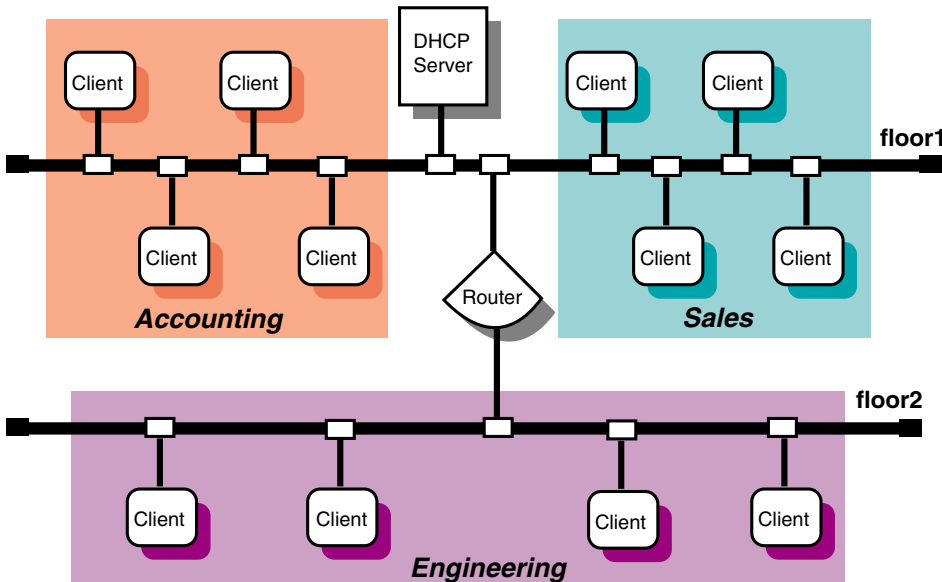
5.1 DHCP Environment

In the DHCP environment, systems can have the following roles:

- **Server** — A system that offers DHCP and BOOTP services to other systems on the network. Multiple servers can exist on a subnetwork, but each server's IP address range cannot overlap. If a cluster member is to support a DHCP server, there can be only one DHCP server for all of the cluster members using a common database with failover.
- **Client** — A system that requests configuration information from a DHCP server. A cluster member must never be a DHCP client. Use static addressing for cluster members.

Figure 5–1 shows a sample corporate local area network (LAN), named *acme-net*, in which a DHCP server is configured to supply IP addresses to clients in three different functional areas. In this configuration, the router must be configured to forward BOOTP packets. DHCP packets are BOOTP packets with DHCP extensions. See `bprelay(8)` for more information.

Figure 5–1: DHCP Configuration (*acme-net*)



ZK-1146U-AI

5.1.1 DHCP Parameter Assignment

In the DHCP environment, DHCP parameters can be assigned to the following named entities:

- **Groups** — Group parameters apply to all clients (nodes) on the network that share the same configuration values. By grouping these clients together, you can simplify the implementation and maintenance of your network configuration. You define a parameter once for a group instead of once for each individual node. After the group parameters are defined, you can use the settings for other subnetwork or node configurations. You can group nodes by logical area, by functional area, by physical area, or in any way you want. Groups can also be grouped together with other groups, subnetworks, and nodes.
- **Subnetworks** — Subnetwork parameters apply to all clients (nodes) on a subnetwork. A subnetwork can also be considered a group, but a group that also shares a common subnetwork address. Subnetworks can be grouped together with other subnetworks and nodes.
- **Nodes** — Node parameters apply to an individual client (node) in the network, and typically override subnetwork or group parameters.

These entities and their parameters have a hierarchical relationship to each other in your network. For example, Figure 5–1 shows a small business network named *acme-net*, comprising two subnetworks and three distinct groups, Accounting, Sales, and Engineering. A DHCP administrator might look at this network as one group named *acme-net*, consisting of two subnetworks, *floor1* and *floor2*, that contain the individual nodes.

The *acme-net* group, at the top level of the hierarchy, specifies those parameters that apply to all systems in the network. At the next level, the *floor1* subnetwork specifies those parameters that apply to all nodes on that subnetwork and the *floor2* subnetwork specifies those parameters that apply to all nodes on that subnetwork. If it were necessary to assign parameters on a group basis, the administrator could have the *floor1* subnetwork consist of the Accounting and Sales groups, with the individual nodes assigned to their respective groups. However, since these groups are on the same subnetwork, this is probably unnecessary.

If Figure 5–1 showed a single LAN with no subnetworks (no router), a DHCP administrator might look at this network as one group named *acme-net*, consisting of three groups (Accounting, Sales, and Engineering) that contain the individual nodes, respectively.

Groups can also be used to define a group of settings for one Ethernet or subnetwork number, allowing you to reuse the settings for other nodes or subnetwork configurations.

5.1.2 DHCP and Security

You can restrict client access to the DHCP server by creating a Media Access Control (MAC) address database. Only those clients with addresses in the

database are allowed to receive an IP address. See Section 5.4.4 for more information.

5.2 Planning DHCP

This section describes those tasks you need to do before configuring DHCP.

5.2.1 Verifying Installation of the DHCP Software

For a DHCP server system, verify that the DHCP server is installed by entering the following command:

```
# setld -i | grep OSFINET
```

If the subset is not installed, install it by using the `setld` command. For more information on installing subsets, see `setld(8)` or the *Installation Guide*.

For DHCP client systems, the DHCP client software is installed with the mandatory subsets.

5.2.2 Preparing for the Configuration

After you verify that the DHCP software is installed, you can configure DHCP by using the `xjoin` utility to:

- Specify server parameters
- Specify basic DHCP parameters for groups, subnetworks, and nodes

The information you need depends on how you define the DHCP environment. The following sections contain worksheets that you can use to record the information required to configure DHCP.

5.2.2.1 Server/Security Parameters

Figure 5–2 shows the DHCP Server/Security Parameters Worksheet. If you are viewing this manual on line, you can use the print feature to print this worksheet. The following sections explain the information you need to record on the worksheet.

Figure 5–2: DHCP Server/Security Parameters Worksheet

DHCP Server/Security Parameters Worksheet	
BOOTP address from pool:	True <input type="checkbox"/> False <input type="checkbox"/>
BOOTP compatibility:	True <input type="checkbox"/> False <input type="checkbox"/>
Default lease time:	_____
Name service:	/etc/hosts <input type="checkbox"/> DNS <input type="checkbox"/> NIS <input type="checkbox"/>
Ping timeout:	_____
Provisional time to live:	_____
Restrict to MAC address:	True <input type="checkbox"/> False <input type="checkbox"/>
IP ranges	
Subnetwork address:	_____
DHCP server:	_____
IP ranges:	_____

Host name lists	
Domain name:	_____
DHCP server:	_____
Host name prefix:	_____
Host names:	_____

BOOTP address from pool

If you want the DHCP server to allocate an address from the pool to BOOTP clients, check True. The address allocation is permanent. If you want the DHCP server to support BOOTP clients whose address is configured in the `/etc/bootptab` file (the usual method), check False; this is the default.

BOOTP compatibility

If you want the server to act as a BOOTP server in addition to a DHCP server when a client requests a BOOTP address, check True. For no BOOTP client support, check False. If you want to configure a BOOTP server only, see Section 5.4.6.

Default lease time

The default time (in days, hours, minutes, and seconds) of a client's DHCP lease, unless one is explicitly configured for the node, subnetwork, or group.

Name service

The name service to be used by the server. A name service must be configured for the DHCP server. The name service is used to authenticate, route, address, and perform naming-related functions for other systems on the network. The following types of name services can be used by the server:

- A Local Name Service updates the `/etc/hosts` file with information about dynamically assigned names and addresses.
- The Domain Name System (DNS) automatically translates host names to their numeric IP address.
- The Network Information Service (NIS) allows you to distribute host name information in a network.

Ping timeout

The time (in milliseconds) for the ping timeout. The ping command is used to determine if a client on your network is available. When the ping program sends a request to the client, the client responds to the request and includes its IP address in the response. The Ping timeout parameter is used to check that no other client is using an IP address prior to it being assigned by the server. After the timeout, the ping command stops checking.

Provisional time to live

The maximum time (in hours, minutes, and seconds) that an IP address remains on the provisionally allocated list before it can be allocated to another client. This prevents an IP address from being reused too quickly after a lease has expired.

Restrict to known MAC address

If you want to assign an IP address to a client's matching MAC address, check True; otherwise, check False. See Section 5.4.4 for additional information on restricting client access to the server.

IP ranges are those IP addresses available for assignment to clients on the network. Although multiple DHCP servers can reside on the same subnetwork, the IP address ranges administered by each server must not overlap. For IP ranges, supply the following information:

Subnetwork address

Subnetworks are logical subdivisions of a single TCP/IP network. The subnetwork IP number identifies one segment of the network. As the number of networks grows, routing IP addresses can get very

complicated. Using subnetworks allows more flexibility when assigning network addresses and simplifies the administration of network numbers. The IP address consists of the following information:

- Network address
- Subnetwork address
- Host address

The IP address is divided into four fields, each separated by a period. Each field represents an element of the address; for example, the following is a typical IP address:

```
128.174.139.47
```

In this example, 128.174 is the network address, 139 is the subnetwork address, and 47 is the host address; therefore, the full subnetwork address is 128.174.139.0.

DHCP server

The IP address of the DHCP server.

IP ranges

The group of unique IP addresses to be assigned to clients on the selected subnetwork. Using the previous subnetwork address as an example, if there are 25 clients on the subnetwork, the range of IP addresses is: 128.174.139.47 to 128.174.139.72.

A subnetwork address can have more than one corresponding IP Address Range.

The DHCP server can configure clients on more than one subnetwork as long as the routers between the server and the client forward BOOTP packets. See Section 5.2.2.2 and `bprelay(8)` for information about boot file and BOOTP parameters.

A host name list contains the names that are assigned clients when they are also assigned an IP address. For host name lists, supply the following information:

Domain name

A domain represents computers that are grouped together for administrative reasons. Domain names are usually assigned to a company, and make administering the domain easy. For example, if a domain is changed to have access to a new service on the network, each computer that is part of the domain automatically has access to the new service.

Write down the domain name exactly as it was assigned by the NIC Domain Registrar, and include its top-level domain extension; for example, `school.edu`, `company.com`, and `city.gov`.

DHCP server

The IP address of the DHCP server.

Host name prefix

A specific host name prefix that is assigned to a system when the system requests a host name and there are no host names available for assignment. For example, in the `company.com` domain, if the names in the Host name list box are all assigned and the host name prefix is `net12host`, the next computers to request host names will receive `net12host1`, `net12host2`, and so on as their host names.

Host names

The host names to be assigned to systems that request them.

5.2.2.2 Information for Basic DHCP Parameters

Figure 5–3 shows the Basic DHCP Parameters Worksheet. If you are viewing this manual on line, you can use the print feature to print this worksheet. The following sections explain the information you need to record on the worksheet.

Figure 5–3: Basic DHCP Parameters Worksheet

Basic DHCP Parameters Worksheet	
Configuration type:	Node <input type="checkbox"/> Subnet <input type="checkbox"/> Group <input type="checkbox"/>
Configuration name:	_____
Member of group:	_____
Group members:	_____
Net or subnetwork IP address:	_____
Hardware address:	_____
Hardware type:	_____
BOOTP Parameters	
Boot file:	_____
Boot file server address:	_____
Boot file size:	_____
DNS domain name:	_____
DNS server IP addresses:	_____

Home directory:	_____
Host IP address:	_____
Routers:	_____

Send client's host name:	True <input type="checkbox"/> False <input type="checkbox"/>
Subnetwork mask:	_____
TFTP root directory:	_____
Broadcast address:	_____
Subnetworks are local:	True <input type="checkbox"/> False <input type="checkbox"/>
Supply masks:	True <input type="checkbox"/> False <input type="checkbox"/>
DHCP rebinding time:	_____
DHCP renewal time:	_____
Lease time:	_____

Configuration type

For node configuration, check Node. For subnetwork configuration, check Subnet. For group configuration, check Group.

Configuration name

The name of the node, group, or subnetwork.

Member of group

For node, subnetwork, and group configurations, the name of a configuration from which to inherit DHCP parameter values. Parameters defined for that group also apply to this configuration.

Group members

For group configuration, the nodes, subnetworks, and groups that compose this group.

Net or subnetwork IP address

For subnetwork configuration, the IP address of the subnetwork. The IP address format is *ddd.ddd.ddd.ddd*. For example, if your subnetwork is 16.128, enter 16.128.0.0; you must include the trailing zeros.

Hardware address

For node configuration, the Ethernet address of the client node.

Hardware type

For node configuration, a descriptive name to identify the system.

For node, subnetwork, and group configuration, BOOTP parameters allow you to specify how to pass configuration information to hosts on the network. For BOOTP parameters, supply the following information:

Boot file

The fully qualified path name of the client's default boot image.

Boot file server address

The IP address of the server that stores the boot file. The IP address format is *ddd.ddd.ddd.ddd*.

Boot file size

The length, in 512-octet blocks, of the default boot image for the client. The file length is specified as a decimal number.

DNS domain name

The domain name the client uses when resolving host names using the Domain Name System.

DNS server IP addresses

A list of IP addresses of DNS name servers available to the client, in order of preference. The address format is *ddd.ddd.ddd.ddd*.

Home directory

The pathname for the boot file, if it is not specified in the boot file name.

Host IP address

The host IP address for BOOTP clients. The address format is *ddd.ddd.ddd.ddd*.

Routers

A list of IP addresses for routers. The address format is *ddd.ddd.ddd.ddd*.

Send client's host name

If you want to send the client's host name, check True. If you do not want to send the client's host name, check False.

Subnetwork mask

The client's subnetwork mask. A subnetwork mask allows the addition of subnetwork numbers to an address, and provides for more complex address assignments. If both the subnetwork mask and the router option are specified in a DHCP reply, the subnetwork mask option must be specified first. The subnetwork mask format is *ddd.ddd.ddd.ddd*.

TFTP root directory

The root directory for Trivial File Transfer Protocol (TFTP).

For subnetwork and group configuration, IP layer parameters affect the operation of the IP layer on a per-host basis. The required IP layer parameters are as follows:

Broadcast address

The broadcast address in use on the client's subnetwork. The address format is *ddd.ddd.ddd.ddd*.

Subnetworks are local

If all subnetworks of the IP network to which the client is connected use the same maximum transfer unit (MTU) as the subnetwork to which the client is directly connected, check True; otherwise, check False. It is recommended for the client to assume that some subnetworks of the directly connected network have smaller MTUs.

Supply masks

If the client responds to subnetwork mask requests using ICMP, check True; otherwise, check False.

For a list of additional parameters and a description of each, see the *JOIN Server Administrator's Guide* (`/usr/doc/join/TOC.html`).

For node, group, and subnetwork configuration, lease parameters allow you to specify information about IP lease times. Lease times determine the length of time an IP address is used. For the lease parameters, supply the following information:

DHCP rebinding time

The time interval (in seconds) from address assignment until the client requests a new lease from any server on the network.

DHCP renewal time

The time interval (in seconds) from address assignment until the client attempts to extend the duration of its lease with the original server.

Lease time

The amount of time (in months, days, hours, minutes, and seconds) the DHCP server will allow a DHCP client to use an IP address; for example, 2 months 5 days 45 minutes. The actual lease time is negotiated between the client and server.

5.3 Configuring a DHCP Server

Use the `xjoin` utility to configure a DHCP server. To start the utility, enter the following command:

```
# /usr/bin/X11/xjoin
```

You can configure the following server information:

- Server/Security parameters
- IP ranges
- Host names
- Subnetworks
- DHCP client nodes
- Groups

When you are finished making changes to these parameters, click on the Add/Update button in the lower right-hand side of the `xjoin` window to

update the server configuration files. To exit the utility, select File and Exit from the menu bar. See `xjoin(8)` and the *JOIN Server Administrator's Guide* (`/usr/doc/join/TOC.html`) for more information.

After you have configured the DHCP server with the `xjoin` utility, see Section 5.3.7 for information about enabling the DHCP server by starting the `joind` daemon.

5.3.1 Configuring Server/Security Parameters

To configure the Server/Security parameters, do the following:

1. Click on the Server/Security tab in the `xjoin` main window.
2. Select the Server item from the left side of the window.
3. Select Server/Security parameters from the pull-down menu.
4. Select a server parameter.
5. Select True or False, or enter a value.
6. Repeat steps 4 and 5 for all server parameters you want to configure.
7. Click on the Add/Update button to update the server with the new Server/Security parameters.

5.3.2 Configuring IP Ranges

To configure IP ranges, do the following:

1. Click on the Server/Security tab in the `xjoin` main window.
2. Select the Server item from the left side of the window.
3. Select IP Ranges from the pull-down menu.
4. Select the New IP Range item.
5. Enter the subnetwork address, server address, and IP range. For each IP range, do the following:
 - a. Enter the beginning of the IP Address Range for the subnetwork (network, subnetwork, and host address).
 - b. Press the Tab key to move to the next field.
 - c. Enter the end of the IP Address Range.
6. Repeat steps 4 and 5 for each new IP range.
7. Click on the Add/Update button to update the server with new IP ranges.

5.3.3 Configuring Host Name Lists

You configure host name lists only if the Accept Client Name server parameter is set to False. If the Accept Client Name server parameter is set to True, the server automatically accepts the name a client suggests for itself; do not configure host name lists.

To configure a host name list, do the following:

1. Click on the Server/Security tab in the `xjoin` main window.
2. Select the Server item from the left side of the window.
3. Select Hostname Lists from the pull-down menu.
4. Select the New Hostname List item.
5. Enter the domain name, DHCP server name, host name prefix, and host names for each host name list.
6. Repeat steps 4 and 5 for each host name.
7. Click on the Add/Update button to update the server with new host name lists.

5.3.4 Configuring a Subnetwork

To configure a subnetwork, do the following:

1. Click on the Subnets tab in the `xjoin` main window.
2. Select the New Record item from the left side of the window.
3. Select the Name parameter. Enter the name of the subnetwork configuration, for example, Subnet3.
4. Select the Member of Group parameter. Enter the name of the group of which the subnetwork will be a member.
5. Select the Net or Subnet IP Address parameter. Enter the Net or Subnet IP address that identifies the subnetwork portion of the network.
6. Select the Broadcast Address parameter. Enter the broadcast address for this subnetwork.
7. Enter information for basic DHCP parameters in the appropriate fields. See Section 5.2.2 and the *JOIN Server Administrator's Guide* (`/usr/doc/join/TOC.html`) for descriptions of these parameters.

Note that you do not have to change each parameter value in the Subnets tab; only those that describe your particular network configuration.

8. Click on the Add/Update button to update the server with new subnetwork configuration information.

9. Edit the `/etc/join/netmasks` file and add an entry for each subnetwork in your network. The format of each entry is as follows:

subnet_address subnet_mask

5.3.5 Configuring a DHCP Client Node

To configure a node, do the following:

Note

A cluster member must never be a DHCP client. Use static addressing for cluster members..

1. Click on the Nodes tab in the `xjoin` main window.
2. Select the New Record item from the left side of the window.
3. Select the Name parameter. Enter the name of the node configuration; for example, Client5.
4. Select the Hardware Type parameter. Enter the type of network to which the node is connected; for example, Token Ring, Ether3, Pronet, Arcnet, or 0.
5. Select the Hardware Address/Client ID parameter. Enter the hardware address or the client ID of the node. If the Hardware Type defined in the previous step is zero, enter the Client ID (an alphanumeric string that you define).

If you are using the hardware address (MAC address) of the node, enter it in the format *nn:nn:nn:nn:nn:nn* (for instance, 08:00:26:75:31:81). The hardware address is assigned when a workstation is manufactured, and is often displayed when the workstation is turned on or rebooted. The hardware address is also called the Ethernet address.

6. Select the Member of Group parameter. Enter the name of the group of which the node will be a member.
7. Enter information for basic DHCP parameters. See Section 5.2.2 and the *JOIN Server Administrator's Guide* (`/usr/doc/join/TOC.html`) for descriptions of these parameters.

Note that you do not have to change each parameter value in the Nodes tab, only those that describe your particular network configuration.

8. Click on the Add/Update button to update the server with new node configuration information.

Depending on the DHCP client, the MAC address field is not always the actual MAC address of the client's network adapter. The following Microsoft clients are known to modify the MAC address before sending it to the server:

- Windows 95
- Windows NT
- Windows for Workgroups with Microsoft TCP/IP

These clients prefix the MAC address with the hardware type. The MAC address type is 0 and the length is 7 (instead of 6). For example, if your Ethernet address is 11:22:33:44:55:66, you must specify the following for static IP mapping:

- MAC address: 01:11:22:33:44:55:66
- MAC type: 0
- MAC length: 7

If you do not specify the MAC address in this manner, the client will fail to collect an IP address from the DHCP server.

See the documentation for your Microsoft product for more information.

5.3.6 Setting Group Parameters

To define a group, do the following:

1. Click on the Groups tab in the `xjoin` main window.
2. Select the New Record item from the left side of the window.
3. Select the Name parameter. Enter the name of the group configuration; for example, Global.
4. Select the Member of Group parameter. If appropriate, enter the name of the group of which that the new group will be a member.
5. Select the Group Members parameter. Enter the names of subnetworks, nodes, or other groups that will be members of the group. Press the Tab key between entries.
6. Enter information for basic DHCP parameters. See Section 5.2.2 and the *JOIN Server Administrator's Guide* (`/usr/doc/join/TOC.html`) for descriptions of these parameters.

Note that you do not have to change each parameter value in the Groups tab, only those that describe your particular network configuration.

7. Click on the Add/Update button to update the server with new group configuration information.

5.3.7 Starting the DHCP Server (joind)

After you install the OSFINET optional subset, run the installation script, and configure the server, use the SysMan Menu application of the Common Desktop Environment (CDE) Application Manager to start the DHCP server and implement the new configuration. To invoke the SysMan Menu application, follow the instructions in Chapter 1.

To start the DHCP server, do the following:

1. From the SysMan Menu, select Networking→Additional Network Services→Set up the system as a DHCP Server (joind) to display the DHCP Server Daemon dialog box.

Alternatively, enter the following command on a command line:

```
# /usr/bin/sysman joind
```

The utility asks if you want this system to be a DHCP server.

2. Select the Yes radio button to enable the joind daemon.
3. Set the debugging level. The default is 0 for no debugging information. Higher numbers produce more detailed debugging information.
4. Set the Log Level by selecting the appropriate radio button.
5. Select OK to save the changes and enable the joind daemon. You are informed that the daemon is running.
6. Select OK to dismiss the message and close the DHCP Server Daemon dialog box.

The DHCP Server Daemon dialog box also allows you to disable and stop the joind daemon. See the SysMan Menu online help for additional information.

Caution

Do not use the `kill -9` command to stop the DHCP server daemon; it can corrupt your database files. Use the Configuring DHCP Server Daemon dialog box or the `kill -HUP` command instead.

See `joind(8)` for more information about the joind daemon.

5.4 Managing DHCP

This section describes how to perform the following DHCP tasks:

- Start the DHCP client

- Monitor DHCP client configuration
- Map client IP addresses permanently
- Restrict access to the DHCP server
- Configure a BOOTP client
- Disable DHCP address assignment

5.4.1 Starting the DHCP Client

When you configure a network interface with the SysMan Menu utility, you must specify how it is to obtain an IP address. If you intend to use a dynamically-assigned address from a DHCP server, select the radio button for Use DHCP in the appropriate dialog box, then start or restart network services. See Section 2.3.1 for more information about configuring network interfaces.

The DHCP client daemon subsequently starts and uses DHCP to obtain IP configuration information. From then on, the DHCP client automatically starts and obtains an IP address each time the client computer boots.

5.4.2 Monitoring DHCP Client Configuration

After the initial DHCP server configuration, you can check the status of a DHCP client by examining the contents of the `/var/join/log` file or by doing the following:

1. Log in as root to the DHCP server host.
2. Invoke the `xjoin` utility by entering the following command:


```
# /usr/bin/x11/xjoin
```
3. Click on the Server/Security tab in the `xjoin` main window.
4. Select Active IP Snapshot from the pull-down menu. The Active IP Snapshot window is displayed, listing each configured DHCP client.
5. Click on a record on the left side of the window to display all current configuration information for the client.

You can also use the `xjoin` utility to modify client configuration information, permanently map a hardware address to an IP address, import a file into the active IP database, and remove records from this window. See `xjoin(8)` and the *JOIN Server Administrator's Guide* (`/usr/doc/join/TOC.html`) for more information.

5.4.3 Mapping Client IP Addresses Permanently

Typically, a client is assigned the first available IP address from the pool of IP addresses. However, you might want to permanently assign an IP address to a client's hardware address or Media Access Control (MAC) address. The IP address mapped to a hardware address does not need to come from the IP addresses you have already defined. To permanently map an IP address to a client's hardware address, do the following:

1. Log in as root to the DHCP server.
2. Invoke the `xjoin` utility by entering the following command:

```
# /usr/bin/X11/xjoin
```
3. Click on the Server/Security tab in the `xjoin` main window.
4. Select Active IP Snapshot from the pull-down menu. The Active IP Snapshot window is displayed.
5. Select the New Record item.
6. Enter a value for each parameter. Press the Return or Tab key after each entry. Specify the integer `-1` for Lease Expiration to ensure that the IP address assignment is preserved in the DHCP database (it will never expire).
7. Click on the Add/Update button to add the new record to the database.
8. Repeat steps 5, 6, and 7 for each MAC address you want to permanently map.

5.4.4 Restricting Access to the DHCP Server

You restrict client access to the DHCP server only if you set the Restrict to Known MAC Address server parameter to True. (See Section 5.2.2.1.) If you set the Restrict to Known MAC Address server parameter to True, you must create a list of MAC addresses that can access and accept IP address assignments from the DHCP server. If you set the server parameter to False, do not create a list of MAC addresses.

To create a list of MAC addresses that can access the DHCP server, do the following:

1. Click on the Server/Security tab in the `xjoin` main window.
2. Select Preload MAC Addresses from the pull-down menu. The Preload MAC Addresses window is displayed.
3. Select the New Record item.
4. Enter a value for each parameter. Press the Return key after each entry.
5. Click on the Add/Update button to add the new record to the database.

6. Repeat steps 3, 4, and 5 for each MAC address that you want to access the DHCP server.

Alternatively, you can import a file into the MAC address database by using the `jdbmod` command. See `jdbmod(8)` for information on the imported file format.

To remove records from the MAC address database, select a MAC address from the left side of the window and click on the Delete button.

5.4.5 Configuring a BOOTP Client

To register a client to use BOOTP only, do the following:

1. Log in as root.
2. Invoke the `xjoin` utility by entering the following command:

```
# /usr/bin/x11/xjoin
```
3. Click on the Nodes tab in the `xjoin` main window.
4. Enter BOOTP client information, including the boot file name, host IP address, subnetwork mask, and any other required information. The basic BOOTP parameters are grouped together below the Key parameters in the middle column. To display additional parameters, click on the Basic DHCP Parameters pull-down menu and select DHCP Parameters.
5. Click on the File/Update button to update the server with the BOOTP client information.

5.4.6 Disabling DHCP Address Assignment

In some cases, you might want to disable DHCP address assignment and use the BOOTP and DHCP server daemon (`/usr/sbin/joind`) to respond to BOOTP requests only. To disable all DHCP address assignment features in the DHCP and BOOTP server, do not specify an IP address range for any subnetwork (this is the default). If no IP address ranges are defined, the server never sends a DHCP reply in response to a DHCP client request.

If DHCP address assignment is disabled, DHCP clients that have previously registered with this server continue to operate until their leases timeout; the server will fail to renew the client lease.

Point-to-Point Connections

The Tru64 UNIX system supports point-to-point connections using the Serial Line Internet Protocol (SLIP) and the Point-to-Point Protocol (PPP).

This chapter describes how to plan for and configure dial-in and dial-out systems for:

- SLIP connections (Section 6.1)
- PPP connections (Section 6.2)
- General modem connections (Section 6.3)

For troubleshooting information, see Section 8.7 for SLIP and Section 8.8 for PPP.

6.1 Serial Line Internet Protocol (SLIP)

The Serial Line Internet Protocol (SLIP) is a protocol used to run IP over serial lines between two hosts. You can connect the two hosts either directly or over telephone circuits using modems. TCP/IP commands (such as `rlogin`, `ftp`, and `ping`) can be run over the SLIP connection.

6.1.1 SLIP Environment

In the SLIP environment, systems can be directly connected to each other, if they are in close proximity, or connected through modems and a telephone network, if they are not. Figure 6–1 shows both of these simple SLIP configurations. Figure 6–2 shows a SLIP connection between two systems with host B acting as a gateway system.

Figure 6–1: Sample Simple SLIP Configuration

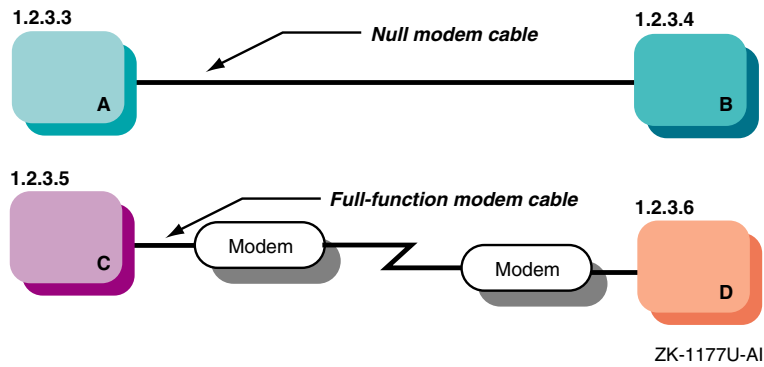
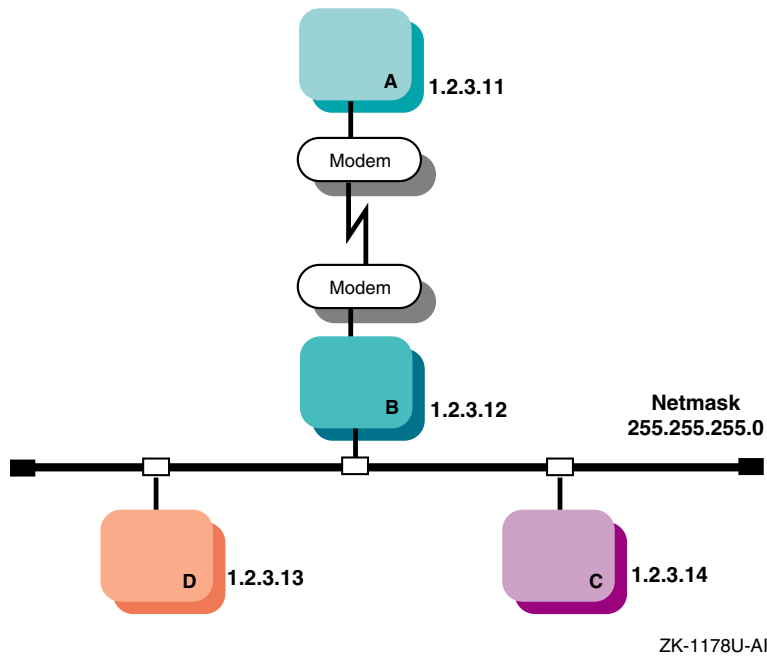


Figure 6–2: SLIP Configuration with Gateway System



6.1.2 Planning SLIP

This section describes those tasks you must complete before configuring SLIP.

6.1.2.1 Verifying the Hardware

When you verify the hardware, you need to verify both the cables and modems, if used.

Make sure you use the correct cable to connect to the serial port of your system. If you do not, you might experience signal degradation and the software will fail to function properly.

If the two systems are in close proximity to each other, use one of the null modem cables listed in Table 6–1.

Table 6–1: Types of Null Modem Cable

Cable Number	Description
BC22D-xx ^a	Asynchronous null modem cable (male DB25 pin to female DB25 pin cable)
BC22R-xx ^a	RS-232 null modem cable (male DB25 pin to female DB25 pin cable)
BC24C-xx ^a	25-wire null modem cable (male DB25 pin to female DB25 pin cable)
BC29Q-xx ^a	Male DB9 pin to female DB9 pin cable

^a xx denotes the cable length. For example, BC29Q-10 is a ten-foot cable.

If the two systems are connected through modems and telephone lines, see Table 6–7 for a list of modem cables to use.

When using modems with SLIP, adhere to the following guidelines:

- Use modems that can handle a serial port speed of 38,400 bits per second (bps). If the modems you plan to use cannot handle a serial port speed of 38,400 bps, set them to the highest speed to which they can be set.
- Use modems that are V.34bis compliant with V.42bis compression. Alternatively, you can use modems that support the Microcom Network Protocol (MNP) because both V.42bis and MNP implement a subset of the other protocol.
- Set the modems to 8 bits, no parity, and connect them to the telephone network.
- Use hardware flow control, if possible. High-speed modems often fall back to a lower data rate when line degradation occurs.

Note

Do not use software flow control (XON/XOFF). It will corrupt the data stream causing the TCP layer over IP to issue retransmit requests for overruns.

6.1.2.2 Preparing for the Configuration

After you verify the communication hardware, you can set up the system to run SLIP.

Figure 6–3 shows the SLIP Setup Worksheet, which you can use to record the information that you need to configure SLIP. The following sections explain the information you need to record on this worksheet. If you are viewing this manual on line, you can use the print feature to print the worksheet.

Figure 6–3: SLIP Setup Worksheet

SLIP Setup Worksheet	
Type of connection:	<input type="checkbox"/> Hardwired <input type="checkbox"/> Modem
Type of system:	<input type="checkbox"/> Dial-in <input type="checkbox"/> Dial-out
Local IP address:	_____
Network mask:	_____
Destination IP address:	_____
Terminal name:	_____
Speed:	_____
SLIP login information:	_____
Dialout systems	
startslip subcommands:	_____

Dialin systems	
slhosts file options:	_____
Gateway:	<input type="checkbox"/> Yes <input type="checkbox"/> No

Type of connection

Check Hardwired if the two systems are connected by a null modem cable, such as BC22D-xx. Check Modem if the two systems are connected by modem cables, modems, and a telephone network.

Type of system

Check dial-in if the system is to answer calls from remote systems. Check dial-out if the system is to place calls to a remote system.

Local IP address

Your system's SLIP interface IP address. Each SLIP interface must have an IP address. For more information on SLIP, see the *Technical Overview* and `startslip(8)`.

Network mask

Your network's subnetwork mask. This must be the same for both systems. See Section 2.2 for more information on the network mask.

Destination IP address

The destination system's SLIP interface IP address.

Terminal name

The name of a valid terminal device in the `/dev` directory that has a cable connection. This can be either the full path name (for example, `/dev/tty00`) or the name in the `/dev` directory (for example, `tty00`). For more information on the terminal line specification, see `startslip(8)`. If you are unsure of the terminal device, see `port(7)`.

Speed

The serial port speed used to connect the systems to each other or a system and the modem. The default speed is 9600 bps. For more information on the speed, see `startslip(8)`.

SLIP login information

The login information for the SLIP connection. This includes user name, password, and login sequence; for example, the login prompt used on dial-out connections.

startslip subcommands

For dial-out systems, Table 6-2 shows the mandatory `startslip` subcommands that you specify when you create a setup script file. Table 6-3 shows the optional `startslip` subcommands.

Table 6-2: Mandatory `startslip` Subcommands

Subcommand	Information Required
<code>myip</code>	Your system's IP address.
<code>dstip</code>	The destination system's IP address.
<code>netmask</code>	The network mask for the subnetwork.

Table 6–2: Mandatory startslip Subcommands (cont.)

Subcommand	Information Required
hardwired	None. Specifies that the two systems are connected by a null modem cable.
modemtype	The type of modem used, unless you have a direct connection.
opentty	The serial line and line speed.
dial	The telephone number to dial.
expect	The information that you expect to receive on the serial line; for example, login sequences.
send	The information that you want to send on the serial line.
connslip	Configures the network interface and attaches the serial line to the network interface.

Table 6–3: Optional startslip Subcommands

Subcommand	Description
debug	Generates debugging messages to the log file specified.
gateway	Specifies that the destination system is a gateway to another system on a LAN.
icmpsup	Suppresses Internet Control Message Protocol (ICMP) traffic. ICMP traffic (such as that generated by the ping command) cannot be sent over the SLIP connection. This frees line bandwidth for more critical traffic.
tcpauto	Specifies that the local system compress TCP headers when it detects that the remote system is compressing them. This option can be useful if you do not know whether the remote system is doing TCP header compression. Note: If the tcpauto option is enabled on both systems, TCP header compression does not occur. One of the two systems must explicitly enable TCP header compression.
tcpcomp	Compresses TCP headers before they are sent over the SLIP connection. Compressing the TCP header allows for faster data transfers. The remote system must support this option to decompress the headers when they arrive at the remote end.

See `startslip(8)` for a complete list of the `startslip` subcommands.

sllhosts file options

For dial-in systems, Table 6–4 shows a list of options for each SLIP link specified in the `/etc/sllhosts` file.

Table 6–4: sllhosts File Options

Option	Description
<code>debug</code>	Generates debugging messages to the <code>daemon.log</code> file.
<code>icmpsup</code>	Suppresses Internet Control Message Protocol (ICMP) traffic. ICMP traffic (such as that generated by the <code>ping</code> command) cannot be sent over the SLIP connection. This frees line bandwidth for more critical traffic.
<code>tcpauto</code>	Specifies that the local system compress TCP headers when it detects that the remote system is compressing them. This option can be useful if you do not know whether the remote system is doing TCP header compression. This is the default.
<code>tcpcomp</code>	Compresses TCP headers before they are sent over the SLIP connection. Compressing the TCP header allows for faster data transfers. The remote system must support this option to decompress the headers when they arrive at the remote end. Do not specify the <code>tcpcomp</code> and <code>tcpauto</code> options together.

See `sllhosts(4)` for more information.

Gateway

For dial-in systems, if your system is to act as a gateway for a dial-out system to access the LAN, check Yes; otherwise, check No.

6.1.3 Configuring SLIP

To configure SLIP, you must have verified the communications hardware and completed the configuration worksheet.

A system in a SLIP environment can have one of the following roles:

- Dial-in system
- Dial-out system

You edit system files and use the `startslip` program to configure both dial-in connections and dial-out connections.

6.1.3.1 Configuring a Dial-In System

To configure a dial-in system, log in as root and complete the following steps:

1. Set up your modem for dial-in access. See Section 6.3.2 for more information.

Note

Use a `getty` process for SLIP dial-in access.

2. Edit the `/etc/passwd` file and create a dedicated entry for a SLIP user. For the login shell field, specify `/usr/sbin/startslip`. The login name you specify here is used to find an entry in the `/etc/slhosts` file, for example:

```
slip1:password:10:20:Remote SLIP User:/usr/users/guest:/usr/sbin/startslip
```

3. Edit the `/etc/slhosts` file and create an entry for the login name using the information from the worksheet. An `/etc/slhosts` file entry has the following syntax:

login_name remote_ip local_ip netmask option

For example, if host D is the dial-in system in Figure 6–1, the entry is as follows:

```
slip1 1.2.3.6 1.2.3.5 255.255.255.0 nodebug
```

See `slhosts(4)` for more information.

4. Edit the `/etc/inittab` file and create an entry for each terminal device that is to run SLIP. An `/etc/inittab` file entry has the following syntax:

Identifier:Runlevel:Action:Command

For example:

```
modem:3:respawn:/usr/sbin/getty /dev/tty00 M38400 vt100
```

See `inittab(4)` for more information.

5. Issue the `init q` command to start the `getty` process immediately.
6. If the dial-in system will be a gateway for the dial-out system to reach other systems on the LAN, the dial-in system must be configured as an IP router and must also run the `gated` daemon. See Chapter 2 for basic network setup information.

If problems occur while using SLIP, see Section 8.7.

6.1.3.2 Configuring a Dial-Out System

To configure a dial-out connection, log in as root and complete the following steps:

1. Verify that there is an entry for your modem name in the `/etc/acucap` file. If your modem does not have an entry in the `/etc/acucap` file, do the following:
 - a. Copy an entry similar to that of your modem.
 - b. Modify the modem attributes to match your modem's attributes. Set up the modem for dial-out access by including the AT commands listed in Table 6–5 in the synchronization string (`ss`) of the entry. The other modem settings can remain as they are.

Table 6–5: Modem Commands for Dial-Out Access

Command	Description
<code>at&c1</code>	Normal Carrier Detect (CD) operation. Tells the modem to not raise Carrier Detect until it sees Carrier Detect from the other modem.
<code>at&d2</code>	Normal Data Terminal Ready (DTR) operation. This tells the modem to hang up the line when DTR drops; for example, when the user logs off the system.
<code>ate1</code>	Turns on echoing.
<code>atq0</code>	Displays the result codes.
<code>ats0=0</code>	Does not answer the phone.

In addition, include the debug option (`db`). With debugging turned on, the modem will provide you with additional information with which to tune the modem attributes in the file. See `acucap(4)` for more information.

2. If you use the `getty` command to provide access to the system from a modem and a `getty` process is already running, do the following:
 - a. Edit the `/etc/inittab` file and change the Action field of the modem entry from `respawn` to `off` as follows:

```
modem:23:off:/usr/sbin/getty /dev/tty00 M38400 vt100
```

See `inittab(4)` for more information.
 - b. Issue the `init q` command to terminate the `getty` process.
3. Check the `/usr/spool/locks` directory for `LCK. .tty n n` lock files. If any exist for the terminal device you are configuring for SLIP, remove them.

When you establish a connection over a terminal device, the system generates a lock file to prevent the connection from being disrupted by another application. If the connection terminates abnormally, the lock file might persist, preventing you from establishing new connections.

4. Create a file that contains `startslip` subcommands for SLIP dial-out connections by doing the following:
 - a. Copy the sample script file from `startslip(8)` to a new script file.
 - b. Use the `tip` command to dial out and log in to the remote system, writing down the exact prompt and login sequence on the worksheet.
 - c. Edit the script file, modify the `expect` subcommands with the prompt and login information, and modify other subcommands with information from the worksheet.

Note

The sample script file specifies the `debug` subcommand and a debug file name at the beginning of the file.

See `startslip(8)` for more information.

5. Invoke the `startslip` command with the `-i filename` option. The *filename* is the name of the file containing the `startslip` subcommands.

After making the connection, the `startslip` command runs in the background. The telephone number (if any) and the process ID are logged in the `/var/run/ttyxx.tel-pid` file.

If problems occur while using SLIP, see Section 8.7.

6.1.4 Terminating a SLIP Dial-Out Connection

To terminate a SLIP dial-out connection, do the following:

1. Determine the process ID of the `startslip` process to kill by using the following command:

```
# cat /var/run/ttyxx.tel-pid
phonenum 8021455 pid 821
```

In the previous command, `ttyxx` specifies the terminal line used for the SLIP connection. If multiple SLIP connections are active on your system, there will be multiple files in the `/var/run` directory.

2. Kill the `startslip` process by using the following command and specifying the process ID that you found in step 1:

```
# kill 821
```

Do not use a SIGKILL (`kill -9`) to terminate the process, as it might corrupt the tty files.

Alternatively, you can turn off your modem to terminate the dial-out connection.

6.2 Point-to-Point Protocol (PPP)

The Point-to-Point Protocol (PPP) provides a standard way to transmit datagrams over a serial link and a standard way for the systems at either end of the link (peers) to negotiate various optional characteristics of the link. Using PPP, a serial link can be used to transmit IP datagrams, allowing TCP/IP connections between the peers.

The Tru64 UNIX PPP subsystem is derived from public domain ppp-2.3.1, and supports IP datagrams. See RFC 1661, RFC 1662, RFC 1332, and RFC 1334 for more information about PPP.

Establishing a PPP connection between two systems basically involves setting up a serial link and running the `pppd` daemon on both ends of the link.

Systems in a PPP environment can have the following roles:

- Dial-out system
- Dial-in system

6.2.1 PPP Environment

Systems using PPP can be directly connected to each other if they are in close proximity, or connected through modems and a telephone network if they are not. Figure 6–4 shows two simple PPP configurations with PPP connections between two systems.

Figure 6-4: Simple PPP Configurations

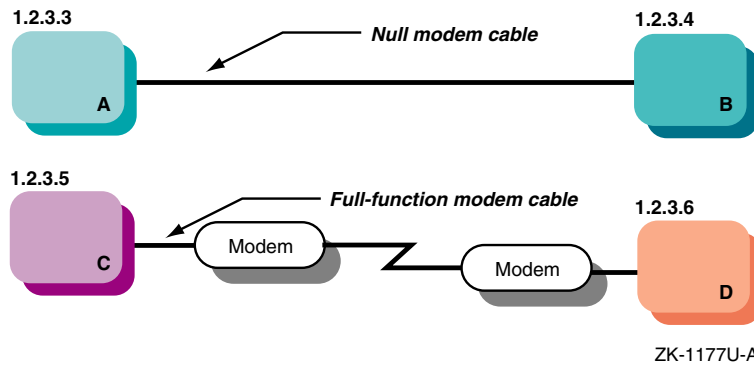
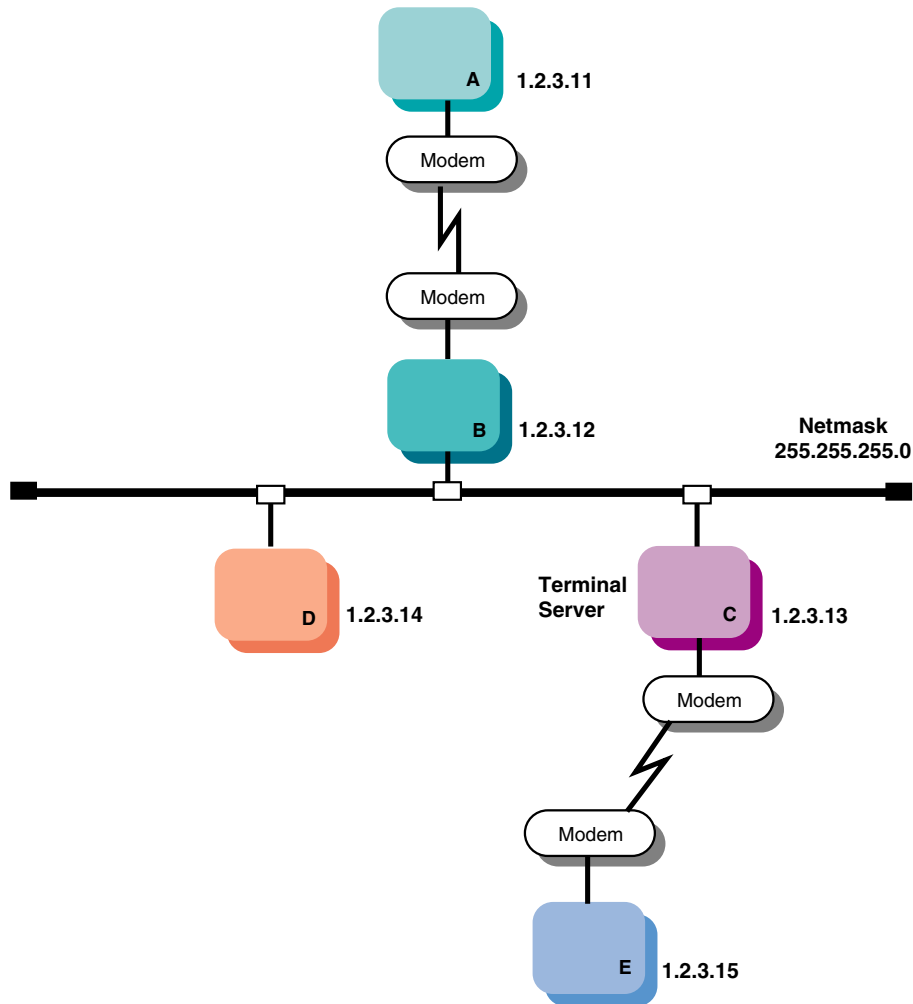


Figure 6-5 shows two PPP connections. The first is between host A and host B, with host B acting as a gateway system. The second is between personal computer E and host D through terminal server C. The latter configuration might be common for employees working at home and dialing in to a system at work.

Figure 6–5: Network PPP Configuration



ZK-1176U-AI

6.2.1.1 Chat Scripts

A chat script can be used to automate the dial-out process for a PPP connection. You can configure it to wait for output from a remote system and reply with responses that you specify.

Each entry in a chat script has the following format:

string_chat_expects string_chat_sends

For example, a chat script might contain the following information:

```

"" atdt2135476 [1]
CONNECT [2]
login: myname [3]
Password: "\qmypassword" [4]
"$ " "\qpppd" [5]
"\qpppd" local_addr: [6]

```

When this chat script is executed, the following steps are taken:

- [1] The chat program expects nothing and sends a dial command to the modem.
- [2] The chat program expects a CONNECT message and sends a carriage return (implied).
- [3] The chat program expects the login: string and sends the *myname* string.
- [4] The chat program expects the Password: string and sends the *mypassword* string. The \q prevents chat from logging the password when you use the -v option.
- [5] The chat program expects the shell prompt (\$) and sends pppd to start the pppd daemon on the remote machine. The \q cancels the effect of the previous \q.
- [6] If you want the local address of the PPP link to differ from the IP address for the local host's Ethernet or other broadcast interface, put the desired address on the pppd command line with a colon appended.

You can create a unique chat script for each remote system to which you connect. Once the scripts are created, you establish a PPP connection to a given system by executing the appropriate script with the chat command, as follows:

```
# chat /etc/ppp/chat-script
```

See chat(8) for more information on the chat command and chat scripts.

6.2.1.2 PPP Options

When you invoke the pppd daemon, you can specify options for it on the command line. These options allow you to configure basic settings such as the speed of the connection, the local and remote IP addresses, and the netmask for the network interface. They also allow you to configure advanced settings such as the types of flow control, authentication, and routing to use.

If you use certain settings each time you initiate a PPP connection, you can automatically enable these settings for each connection by editing the following files:

- `/etc/ppp/options` — This file contains system default options that are read before user default options and command line options. This file contains any options that you want the `pppd` daemon to use whenever it runs. If authentication is required, add the `auth` and `usehostname` options to this file.

Note

If the `/etc/ppp/options` file does not exist or is unreadable by `pppd`, the daemon will not run. Set the file permissions so that only root has write access.

- `/etc/ppp/options.tty.xx` — This file contains options specific to the serial port `/tty.xx`.
- `$HOME/.ppprc` — This file contains the user default options that are read before command line options.

Depending on your configuration, one options file might overrule another for certain parameters. For example, if you specify one set of values for parameters in the `/etc/ppp/options` file, then specify a different set of values for the same parameters in a `/etc/ppp/options.tty.xx` file, the settings in the latter file are used when you connect through the specified serial port.

See `pppd(8)` for a list of the `pppd` options. See Section 6.2.3.2 for information about how to use the SysMan Menu utility to create options files.

6.2.1.3 Authentication

PPP provides three protocols for authenticating hosts and for authenticating your host system to others:

- Password Authentication Protocol (PAP)
- Challenge Handshake Authentication Protocol (CHAP)
- Microsoft Challenge Handshake Authentication Protocol (MS-CHAP)

All protocols exchange secrets in order to complete the authentication process. PAP secrets are contained in the `/etc/ppp/pap-secrets` file; CHAP secrets are contained in the `/etc/ppp/chap-secrets` file. Set the file permissions on these files so that only root has read access.

The `/etc/ppp/pap-secrets` and the `/etc/ppp/chap-secrets` files for PAP and CHAP have the following format:

```
client server secret [ip_address...]
```

- *client* — Name of the machine to be authenticated

- *server* — Name of the machine requiring the authentication
- *secret* — Password or CHAP secret known by both client and server
- *IP address* — Zero or more IP addresses that the client can use (this field is only used on the server)

For example, if a LAN-connected host named *work* requires authentication, and a host named *home* connects to it and authenticates itself using CHAP, the `/etc/ppp/chap-secrets` file on each machine must contain an entry similar to the following:

```
home work "an unguessable secret" home.my.domain
```

Note

The `/etc/ppp` directory contains files of secrets used for authentication, and must not be in a partition that is exported using NFS and accessible by other hosts.

If authentication is required, the `/etc/ppp/options` file must contain the `auth` and `usehostname` options.

Note that the MS-CHAP protocol exchange secrets are located in the `/etc/ppp/chap-secrets` file. The format for this protocol is as follows:

username server secret

- *username* — User name of the user to be authenticated
- *server* — Name of the machine requiring the authentication
- *secret* — Password or CHAP secret known by both client and server

6.2.2 Planning PPP

This section describes the tasks you must complete before configuring PPP.

6.2.2.1 Verifying the Hardware

Verify that you have the hardware to connect to the serial port of your system. If the two systems are in close proximity to each other, use one of the null modem cables listed in Table 6-1.

If the two systems are connected through modems and telephone lines, see Table 6-7 for a list of modem cables to use. The modems are set to 8 bit, no parity, and are connected to the telephone network.

6.2.2.2 Verifying PPP Support in the Kernel

To verify that PPP is supported in the kernel, enter the following command:

```
# sysconfig -s | grep ppp
```

If PPP is not loaded and configured, do the following:

1. Log in as root.
2. Rebuild the kernel by running the `doconfig` utility and selecting the Point-to-Point (PPP) option.
3. Make a backup copy of the current `/vmunix` kernel file.
4. Copy the newly-created `/sys/HOSTNAME/vmunix` kernel file to the `/vmunix` file.
5. Reboot the system.

6.2.2.3 Preparing for Configuration

After you verify PPP support in the kernel, you can configure PPP.

Figure 6–6 shows the PPP Setup Worksheet, which you can use to record the information that you need to configure PPP. The following sections explain the information you need to record on this worksheet. If you are viewing this manual on line, you can use the print feature to print the worksheet.

Figure 6–6: PPP Setup Worksheet

PPP Setup Worksheet	
Type of system:	<input type="checkbox"/> Dial-in <input type="checkbox"/> Dial-out
Local IP address:	_____
Remote IP address:	_____
Network mask:	_____
Terminal name:	_____
Speed:	_____
Level of authentication:	_____
Type of authentication:	<input type="checkbox"/> PAP <input type="checkbox"/> CHAP
Options:	_____

Type of system

Check dial-in if the system is to answer calls from remote systems.
Check dial-out if the system is to place calls to a remote system.

Local IP address

The local system's IP address. For systems connected to a LAN, this address is already assigned if you configured your network software; it is the IP address of the LAN interface.

If you have a standalone system, you must assign it an IP address. If you are using PPP to link your system to a host that is connected to the Internet, assign the local system an address that is on the same subnetwork as the remote host. If the other host is not connected to the Internet, assign the local system any IP address.

Remote IP address

The remote system's IP address.

Network mask

Your network's subnetwork mask. This must be the same for both systems. See Section 2.2 for more information on the network mask.

Terminal name

The name of any valid terminal device in the `/dev` directory. This can be either the full path name (for example, `/dev/tty01`) or the name in the `/dev` directory (for example, `tty01`). If you are unsure of the terminal device, see `ports(7)` for more information.

Speed

The speed of the modem (or null modem) used to connect the systems and the terminal line specification. If your modem automatically senses the line speed or if you are using a null modem cable between hosts, you can specify any speed up to the maximum supported by the hosts. This is usually 38400 bps.

Level of authentication

The level of authentication required. In general, if your system is connected to a LAN, it is best to require the remote host to authenticate itself and to restrict the remote host's choice of IP address based on its identity. Otherwise, a remote host might impersonate another host on the local subnet.

Note

If you are configuring PPP for the first time, do not enable authentication until you can successfully establish a link.

Type of authentication

If you are using PAP authentication, check PAP. If you are using CHAP authentication, check CHAP.

Options

Table 6–6 describes some advanced options that are commonly configured. You can use the SysMan Menu utility to configure these options, as described in Section 6.2.3.2.

Table 6–6: shosts File Options

Option	Description
Asynchronous Character Conversion Map (asynctmap)	If the serial line is not completely 8-bit transparent, specify this option; <code>asynctmap 200a000</code> is appropriate if the serial link includes a <code>telnet</code> link.
Maximum Receive Unit (MRU) Negotiation	To improve performance for multiple IP connections, reduce the Maximum Receive Unit (MRU) on the local and remote system. It is best to set the MRU value to 296.
Hardware Flow Control(RTS/CTS)	Enables hardware flow control on the serial device. If the modem does not support hardware flow control, do not add this entry. See your modem documentation to verify this information.
LCP Echo-Request Interval (lcp-echo-interval)	Sends a Link Control Protocol (LCP) echo request frame to the remote system every 60 seconds. This determines whether the link to the remote system is still active.
Maximum LCP-Echo Requests (lcp-echo-failure)	If the local system does not receive a response from the remote system after five LCP echo request frames, the <code>pppd</code> daemon considers the link dead and tears down the connection.
Force peer to supply local IP address (noipdefault)	Specifies that the remote system (ISP) is to provide the local system an IP address, unless an IP address is specified explicitly on the command line or in an options file.
Enable debugging (debug)	Enables debugging. All messages are sent to the file specified in the <code>/etc/syslog.conf</code> file. After your connection is working correctly, remove this entry.

See `pppd(8)` for a complete list of `pppd` options.

6.2.3 Configuring a Dial-Out System with PPP

If the system will place calls to a remote system, you must establish a dial-out connection, which requires you to perform the following tasks:

- Set up initial communications
- Create options files

- Set up authentication
- Set up message logging
- Initiate and monitor the PPP connection

The following sections discuss these configuration tasks, and Section 6.2.3.6 describes additional steps you need to take if you are connecting to a Microsoft NT Remote Access Server (RAS).

6.2.3.1 Setting Up Initial Communications for a Dial-Out System

After you connect your modem to a serial port on your system, do the following:

1. Verify that you can communicate with the modem:
 - a. Edit the `/etc/remote` file and copy the `kdebug` entry.
 - b. Modify the new entry, providing a system name, the terminal device name (`tty00` or `tty01` depending on your system), the speed, and parity. See `remote(4)` for more information.
 - c. Check the `/usr/spool/locks` directory for `LCK..ttypn` lock files. If any exist for the terminal device you are configuring for PPP, remove them.

When you establish a connection over a terminal device, the system generates a lock file to prevent the connection from being disrupted by another application. If the connection terminates abnormally, the lock file might persist, preventing you from establishing new connections.

- d. Use the `tip` command to access the modem as follows:

```
% tip system_name
```

The *system_name* is stored in the `/etc/remote` file.

- e. If your modem is using the AT command language, enter the following command:

```
AT RETURN
```

If the modem is not in quiet mode, it responds with an OK message.

2. Contact the administrator of the remote system or your Internet Service Provider (ISP) and obtain the following information:
 - Your remote IP address and netmask, unless the remote system assigns the IP address dynamically
 - Characters that might need to be escaped
 - Instructions on how to log in and use the remote service

This information is used to create a `chat` script, which automates the dial-out process. A `chat` script is a file that contains a list of commands that the `chat` program uses to direct the modem. It contains the number to dial and the information to send to the remote system to start the `pppd` daemon.

Note

You can use the `tip` command to dial out and log in to the remote system to collect additional information about the process. Write down the exact prompt, login sequence, and `pppd` start-up sequence for use in the `chat` script.

3. Create a `chat` script, as described in Section 6.2.1.1, to automate the dial-out process.

6.2.3.2 Creating Options Files for a Dial-Out System

Use the SysMan Menu of the Common Desktop Environment (CDE) Application Manager to create PPP options files. To invoke the SysMan Menu application, follow the instructions in Section 1.2.1.

To create an options file for a dial-out system, do the following:

1. From the SysMan Menu, select `Networking`→`Additional Network Services`→`Serial Line Networking`→`Point-to-Point Protocol (PPP)`→`Create option files` to display the PPP Option Files dialog box.

Alternatively, enter the following command on a command line:

```
# /usr/bin/sysman ppp_options
```

2. Select a file from the list that is displayed and select `Modify`. Or, do the following to create a new options file:
 - a. Select the `New File...` option to display the Create PPP Options File dialog box.
 - b. Enter the new file name and select `OK`.

The `Modify PPP Options File` dialog box is displayed.

3. Select `Dial-Out Options` and select `Configure` to display the `Dial-Out Options` dialog box. Complete the fields using the information that you gathered on the PPP Setup Worksheet.

If your system is standalone and you are connecting to the Internet through the remote system, add a default route via the remote host. Under the `System Routing Tables` option, select the appropriate radio button.

See `pppd(8)` for a complete list of `pppd` options.

4. Select OK to close the Dial-Out Options dialog box.
5. Select Advanced PPP Options if you want to configure additional PPP options. Make the necessary changes, then select OK to close the associated dialog box.
6. Select OK in the Modify PPP Options File dialog box to save the changes and to close the dialog box.
7. Select Exit to close the PPP Option Files dialog box.

You can use the SysMan Menu utility to copy, modify, and delete option files. See the online help for more information.

6.2.3.3 Setting Up Authentication for a Dial-Out System

The `chap-secrets` and `pap-secrets` files contain entries that can be used for authentication purposes, as discussed in Section 6.2.1.3. The following sections describe how to create entries in these files.

6.2.3.3.1 Creating Entries in the PAP Secrets File

Use the SysMan Menu of the Common Desktop Environment (CDE) Application Manager to create entries in the `pap-secrets` file. To invoke the SysMan Menu application, follow the instructions in Section 1.2.1.

To create entries in the `pap-secrets` file, follow these steps:

1. From the SysMan Menu, select Networking→Additional Network Services→Serial Line Networking→Point-to-Point Protocol (PPP)→Modify `pap-secrets` file to display the Modify `pap-secrets` File dialog box.

Alternatively, enter the following command on a command line:

```
# /usr/bin/sysman pap
```

2. Select Add to display the Add `pap-secrets` Entry dialog box. Supply the requested information.
3. Select OK to save the current changes and close the dialog box. The Modify `pap-secrets` File dialog box displays the new entry.
4. Repeat steps 2 and 3 as many times as necessary.
5. Select Exit to close the Modify `pap-secrets` File dialog box.

You can also use the SysMan Menu utility to modify or delete entries in the PAP secrets file. See the online help for more information.

6.2.3.3.2 Creating Entries in the CHAP Secrets File

Use the SysMan Menu of the Common Desktop Environment (CDE) Application Manager to create entries in the `chap-secrets` file. To invoke the SysMan Menu application, follow the instructions in Section 1.2.1.

To create entries in the `chap-secrets` file, follow these steps:

1. From the SysMan Menu, select `Networking→Additional Network Services→Serial Line Networking→Point-to-Point Protocol (PPP)→Modify chap-secrets file` to display the Modify chap-secrets File dialog box.

Alternatively, enter the following command on a command line:

```
# /usr/bin/sysman chap
```

2. Select `Add` to display the Add chap-secrets Entry dialog box. Supply the requested information.
3. Select `OK` to save the current changes and close the dialog box. The Modify chap-secrets File dialog box displays the new entry.
4. Repeat steps 2 and 3 as many times as necessary.
5. Select `Exit` to close the Modify chap-secrets File dialog box.

You can also use the SysMan Menu utility to modify or delete entries in the CHAP secrets file. See the online help for more information.

6.2.3.4 Setting Up Message Logging

To set up message logging, complete the following steps:

1. Edit the `/etc/syslog.conf` file, as follows:

Note

Whitespace in the `/etc/syslog.conf` file must consist of tab characters. Spaces are not acceptable. See `syslogd(8)` for further information.

- a. Add the `local2` facility (used by the `pppd` daemon and the `chat` program) to the line that specifies `/dev/console` as the message destination, as follows:

```
kern.debug;local2.notice                               /dev/console
```

In this example, the `notice` severity level is specified. For more information about this severity level and logging system messages in general, see the *System Administration* manual.

- b. Add the following entry to the file to create a `ppp-log` file:

```
local2.debug                                /etc/ppp/ppp-log
```

- c. Save the edits and close the file.

2. Create the PPP log file by issuing the following command:

```
# touch /etc/ppp/ppp-log
```

3. Stop and restart the `syslogd` daemon by entering the following commands:

```
# /sbin/init.d/syslog stop
# /sbin/init.d/syslog start
```

6.2.3.5 Initiating and Monitoring a PPP Connection

Before initiating a PPP connection, note the following guidelines:

- Do not use the `ifconfig` command to configure the addresses of the `ppp` interface. The `pppd` daemon assigns addresses and identifies the interface as running.
- Whether you run the `pppd` daemon manually on the remote machine or use a script file on the local machine to run `pppd` on the remote machine, do not provide a device name to `pppd`; it uses the controlling `tty` by default.

Once you have configured your system for a PPP dial-out connection, initiate the connection as follows:

1. Invoke the `pppd` daemon on the local system to connect to the remote system. For example, the following command starts a link on `tty01` and specifies the `connect` option to run the `chat` program using the specified `chat` script file.

```
% pppd /dev/tty01 38400 connect 'chat -f /etc/ppp/chat-script'
```

2. Issue the following command to monitor the `ppp-log` file and to determine whether the PPP connection is active:

```
% tail -f /etc/ppp/ppp-log
```

After you have established a connection, you can use the `pppstats` and `netstat` commands to obtain current statistics for the PPP interface. For example:

```
% pppstats
IN  PACK VJCOMP VJUNC VJERR |   OUT  PACK VJCOMP VJUNC NON-VJ
9425 311   5     2     0 | 9574  308   1     3     304
% netstat -I ppp0
Name  Mtu   Network      Address          Ipkts Ierrs   Opkts Oerrs   Coll
ppp0  1500  <Link>       Link#6           305   0       302   0       0
ppp0  1500  10           10.1.2.2        305   0       302   0       0
```

For more information about the `pppstats` and `netstat` commands, see `pppstat(8)` and `netstat(1)`.

If problems occur while you are using PPP, see Section 8.8.

6.2.3.6 Connecting to a Microsoft NT Remote Access Server

This section describes how to establish a dial-out connection from a Tru64 UNIX system to a Microsoft NT Remote Access Server (RAS).

You will need to supply the following information in the `/etc/ppp/chap-secrets` file:

- NT login name and password
- NT domain name

For details on creating the `/etc/ppp/chap-secrets` file, refer to Section 6.2.3.3.2 and `pppd(8)`.

6.2.3.6.1 Configuring an NT RAS Server

To configure a Tru64 UNIX system to allow dial-out access to an NT RAS server, do the following:

1. Log in as root.
2. Create an `/etc/ppp/chap-secrets` file. For example, if you are dialing into a server named `money` with a username of `monopoly` and a password of `candlestick`, create the `chap-secrets` file as follows:

```
#
# secret for logging into an NT RAS server
#
monopoly money candlestick
```

3. Issue the `pppd` command with the user and remote name arguments to select the secret for the server `money`. For example:

```
# pppd tty00 38400 username monopoly remotename money
```

If the RAS server you dial out to is not a standalone server or a domain controller, you might need to prepend your NT domain name to your username. To do this from the command line, enter a command similar to the following in which `empire` is the domain name:

```
# pppd tty00 38400 user 'empire\\monopoly' remotename money
```

Note

Single quotes are required in the previous example to escape the backslash characters.

Alternatively, you can place this information in the `/etc/ppp/chap-secrets` file as follows:

```
#
# secret for logging into an NT RAS server
#
empire\\monopoly    money    candlestick
```

You can also use the `chat` program to automate any dialog that is required to establish a dial-out connection. See Section 6.2.1.1 for information on using the `chat` program.

During authentication, Microsoft Windows does not send its node name to the PPP peer. The peer must know beforehand the node name of the Microsoft Windows system to select the correct secret from the `chap-secrets` file. You can do this by setting the `remotename` option of the `pppd` daemon. If this is not done, authentication might fail and the PPP link will be disconnected.

6.2.3.6.2 Solving Microsoft CHAP Authentication Problems

Microsoft CHAP (MS-CHAP) returns error codes if authentication fails. To log the error messages, invoke the `pppd` command with the `debug` option. The error code format is as follows:

```
rcvd [CHAP Failure id=0x0 "E=NUM R=1"]
```

NUM is the error code that MS-CHAP returns.

Possible error codes include:

Error Code	Explanation
E=646	Your NT account has restricted log in hours. At this time of day you may not log on.
E=647	Your NT account has been disabled.
E=648	Your NT account password has expired. (Note that the <code>pppd</code> daemon cannot negotiate a change of password.)
E=649	You are not permitted to dial in.
E=691	The RAS server could not validate your username. You supplied an incorrect password, or you need to prepend your domain name to your username.

6.2.4 Configuring a Dial-In System with PPP

If the system will answer calls from remote systems, you must establish a dial-in connection, which requires you to perform the following tasks:

- Set up initial communications
- Create options files

The following sections discuss these configuration tasks.

6.2.4.1 Setting Up Initial Communications for a Dial-In System

To configure a dial-in system, complete the following steps after you connect your modem to a serial port:

1. Set up your modem for dial-in access. See Section 6.3.2 for more information.
2. Edit the `/etc/passwd` file and create a dedicated entry for a PPP user. For the login shell field, specify `/usr/sbin/startppp`, which starts the `pppd` daemon for dial-in connections. For example:

```
ppp1:password:10:20:Remote PPP User:/usr/users/guest:/usr/sbin/startppp
```

3. Edit the `/etc/inittab` file and create an entry for each terminal device that is to run PPP. For example:

```
modem:3:respawn:/usr/sbin/getty /dev/tty00 M38400 vt100
```

See `inittab(4)` for more information.

4. Issue the `init q` command to immediately start the `getty` process.
5. If the dial-in system will be a gateway for the dial-out system to reach other systems on the LAN, the dial-in system must be configured as an IP router and must run the `gated` daemon. Edit the `/etc/gated.conf` file and delete the `nobroadcast` option (if specified) in the `rip` statement. See Chapter 2 for basic network setup information and `gated.conf(4)` for `gated` options.

6.2.4.2 Creating Options Files for a Dial-In System

Use the SysMan Menu of the Common Desktop Environment (CDE) Application Manager to create PPP options files. To invoke the SysMan Menu application, follow the instructions in Section 1.2.1.

To create an options file for a dial-in system, do the following:

1. From the SysMan Menu, select `Networking→Additional Network Services→Serial Line Networking→Point-to-Point Protocol (PPP)→Create option files` to display the PPP Option Files dialog box.

Alternatively, enter the following command on a command line:

```
# /usr/bin/sysman ppp_options
```

2. Select a file from the list that is displayed and select Modify. Or, do the following to create a new options file:
 - a. Select the New File option to display the Create PPP Options File dialog box.
 - b. Enter the new file name and select OK.

The Modify PPP Options File dialog box is displayed.

3. Select Dial-In Options and select Configure to display the Dial-In Options dialog box. Complete the input fields using the information that you gathered on the PPP Setup Worksheet. By default, an entry is automatically added to the Address Resolution Protocol (ARP) table. If you do not want an entry to be added, set the appropriate radio button to the On position.
4. Select OK to close the Dial-In Options dialog box.
5. Select Advanced PPP Options if you want to configure additional PPP options. Make the necessary changes, then select OK to close the associated dialog box.
6. Select OK in the Modify PPP Options File dialog box to save the changes and to close the dialog box.
7. Select Exit to close the PPP Option Files dialog box.

You can also use the SysMan Menu utility to copy, modify, and delete option files. See the online help for more information.

6.2.5 Terminating PPP Connections

To terminate the PPP link, send a TERM or INTR signal to one of the pppd daemons by issuing the following command:

```
# kill `cat /etc/ppp/pppxx.pid`
```

In the previous command, `pppxx` specifies the pppd daemon used for the PPP connection. The pppd daemon specified in the command notifies other related pppd daemons to terminate, clean up, and exit.

If the pppd daemon is connected to a hardware serial port connected to a modem, it will receive a HUP signal when the modem hangs up, which causes it to clean up and exit. This action depends on the driver and its current settings.

Do not use a SIGKILL (`kill -9`) to kill the process. It might not allow the pppd daemons to terminate properly, which could corrupt the tty files.

6.3 Guidelines for Using Modems

The operating system software enables you to use a variety of modems for point-to-point connections to systems that are not in close proximity to each other. These connections can be Serial Line Internet Protocol (SLIP), Point-to-Point Protocol (PPP), and UNIX-to-UNIX Copy Program (UUCP) connections. In addition, these connections can be basic dial-out or dial-in connections; for example, you can log in to a remote system to perform remote system administration.

This section presents general guidelines for using modems on Tru64 UNIX systems for all types of connections. See Section 6.1.2.1 for specific information on SLIP and PPP connections and see *Network Administration: Services* for information about UUCP connections.

6.3.1 Using the Correct Modem Cables

You must use the correct cable to connect a modem to the serial port. Use of an incorrect cable might result in signal loss and associated software errors. Table 6–7 lists the cables you can use to connect modems. The cable connector is either 25-pin or 9-pin, depending on the type of serial port on your system. See the hardware documentation for your system if you are uncertain about the type of serial port.

Note

DECconnect cables do not provide a sufficient number of wires for full modem control; do not use them.

Table 6–7: Types of Modem Cable

Cable Number	Description
BC22E- <i>xx</i> ^a	16-wire modem cable (male DB25 pin to female DB25 pin cable)
BC22F- <i>xx</i> ^a	25-wire modem cable (male DB25 pin to female DB25 pin cable)
BC29P- <i>xx</i> ^a	Male DB25 pin to female DB9 pin cable
PC modem cable	Male DB25 pin to female DB9 pin cable

^a *xx* denotes the cable length. For example, BC22E-10 is a ten-foot cable.

6.3.2 Configuring a System for Dial-In Access

After you obtain the correct cable and connect your modem to it and the telephone network, do the following:

1. Edit the `/etc/remote` file and create an entry similar to the `kdebug` entry. For example, if your modem is connected to the `tty00` port and

you will use a speed of 38,400 bps to access the modem, create an entry similar to the following:

```
b38400:dv=/dev/tty00:br#38400:pa=none
```

Note

Some modems set their speed to the serial port rate. Be sure to access the modem using the same speed that you will specify to the `getty` or `uugetty` utility. Otherwise, you might not be able to log in because of the mismatch.

2. Check the `/usr/spool/locks` directory for `LCK..ttynn` lock files. If any exist for the terminal device you are configuring for use with the modem, remove them.

When you establish a connection over a terminal device, the system generates a lock file to prevent the connection from being disrupted by another application. If the connection terminates abnormally, the lock file might persist, preventing you from establishing new connections.

3. Use the `tip` command to access the modem as follows:

```
tip b38400
```

The `tip` utility responds with a `connected` message. You can now communicate with the modem.

4. If your modem uses the AT command set, a standard language for communication between terminals and modems, enter the following command to verify that the modem is ready and listening:

```
at Return
```

If the modem is not in quiet mode, it responds with an OK message.

5. Configure the modem for dial-in access as specified in Section 6.3.2.1.
6. Edit the `/etc/inittab` file and create an entry for the modem. If you want to use the modem line in nonshared mode, create an entry similar to the following:

```
modem:23:respawn:/usr/sbin/getty /dev/tty00 M38400 vt100
```

If you want to use the modem line in shared mode (for dial-out and dial-in connections), use the `uugetty` utility instead of the `getty` utility and create an entry similar to the following:

```
modem:23:respawn:/usr/lib/uucp/uugetty -r -t 60 tty00 38400
```

If you specify a speed greater than 9600 bps, you must edit the `/etc/uugettydefs` file and create an entry for the speed you want.

With the `uugetty` utility, you can use the `tip` and `cu` utilities, but differences in file locking might prevent the use of third-party utilities.

Note

If you want to use the `ugetty` utility, you must install the UNIX-to-UNIX Copy Facility subset.

7. As root, start the `getty` or `ugetty` process by entering the following command:

```
init q
```

The `getty` or `ugetty` process starts, then goes to sleep, waiting for someone to dial in to the system.

6.3.2.1 Setting Up a Modem for Dial-In Access

To configure your modem for dial-in access, you need to send various commands to the modem by using the AT command set. Table 6–8 lists the AT commands required. These command settings are generally the same as the default settings for most modems, but you can enter them again to verify that your modem is correctly configured.

Table 6–8: Modem Commands for Dial-In Access

Command	Description
<code>at&c1</code>	Normal Carrier Detect (CD) operation. Tells the modem not to raise Carrier Detect until it sees Carrier Detect from the other modem.
<code>at&d2</code>	Normal Data Terminal Ready (DTR) operation. This tells the modem to hang up the line when DTR drops. For example, when the user logs off the system.
<code>atq1</code>	Sets the modem to quiet mode. Result codes are not sent to the system.
<code>ate0</code>	Echo off. This prevents the modem from echoing the login prompt issued by the <code>getty</code> process.
<code>ats0=n</code>	Specifies the number of rings to wait before answering. If $n = 0$ (zero), the modem will not answer.
<code>at&w0</code>	Saves the current modem settings in NVRAM. Most modems contain user profiles where modem settings can be stored for future use. This command stores the settings in the default profile, 0.

You can enter these commands individually or as one command. For example:

```
at&c1&d2q1e0s0=n&w0 Return
```

Enter the following command to verify the results (these characters are not displayed on the screen because you turned echo off with the `e0` command):

```
at&v Return
```

The active profile and stored profile 0 will reflect the values you entered. The active (or current) profile is lost when you turn the modem off, but the stored profile will preserve the modem settings for future use.

In addition to the specified settings, configure the type of flow control to use for the connection between the computer and the modem. The operating system supports both hardware and software flow control. If your computer supports hardware flow control, set the modem and the serial line to use hardware flow control by using the appropriate commands. If hardware flow control is not supported, use software flow control. See the manuals for your computer and your modem for more information.

6.3.3 Configuring Your System for Dial-Out Access

After you obtain the correct cable and connect your modem to it and the telephone network, do the following:

1. Verify that there is an entry for the modem specified with the `modemtype` subcommand in the `/etc/acucap` file. If an entry does not exist, do the following:
 - a. Copy an entry similar to that of your modem. The following entry is for a US Robotics modem for use in shared mode with the `tip` command:

```
us|US|US Robotics (28.8 fax/data modem):\
:cr:hu:ls:re:ss=AT\rATE1Q0&C0X0&A0\r:sr=OK:\
:sd#250000:di=ATD:dt\r:\
:dd#50000:fd#50:os=CONNECT:ds=\d+++\dATZ\r\dATS0=2\r:\
:ab=\d+++\dATZ\r\dATS0=2:
```
 - b. Modify the modem attributes to match those of your modem and include the debug option (`db`). With debugging turned on, the modem will provide you with additional information with which to tune the modem attributes in the file. See `acucap(4)` for more information.
2. Create an entry in the `/etc/remote` file for the system you want to call, as specified in Section 6.3.3.1.

3. If you use the `getty` utility to provide access to the system from a modem and a `getty` process is already running, do the following:

- a. Edit the `/etc/inittab` file and change the Action field of the modem entry from `respawn` to `off` as follows:

```
modem:23:off:/usr/sbin/getty /dev/tty00 M38400 vt100
```

See `inittab(4)` for more information.

- b. Issue the `init q` command to terminate the `getty` process.

4. Check the `/usr/spool/locks` directory for `LCK..ttypn` lock files. If any exist for the terminal device you are configuring for use with the modem, remove them.

When you establish a connection over a terminal device, the system generates a lock file to prevent the connection from being disrupted by another application. If the connection terminates abnormally, the lock file might persist, preventing you from establishing new connections.

5. Use the `tip` command, specifying the `-baud_rate` flag and the telephone number to dial out as follows:

```
tip -38400 8881234
```

In this example, `tip` strips the minus sign (`-`) from the baud rate and concatenates the `tip` command name and the baud rate to create the string `tip38400`. Then, `tip` searches the `/etc/remote` file for the entry matching the string. The entry in the `/etc/remote` file points to the capability information in the `us38400` entry to initialize the modem.

You can specify the telephone number on the command line to share the same modem attributes for outgoing connections that have different telephone numbers.

When you log off the remote system and exit the `tip` utility, the saved settings are restored and the modem is ready for the next user. If used in shared mode, the modem is available for dial-in access.

6.3.3.1 Creating Entries in the `/etc/remote` File

The `/etc/remote` file stores information about the dial-out connections that you establish.

You can use this file to supply the terminal device name, connection speed, and the `/etc/acucap` file that defines your modem. For example, the following two entries are for the modem specified in step 1a of Section 6.3.3:

```
tip38400:tc=us38400 [1]
us38400|38400 Baud dial out via US Robotics modem:\ [2]
:el=^U^C^R^O^D^S^Q@:ie=#%$:oe=^D:\ [3]
```

```
:dv=/dev/tty00:br#38400:ps=none:at=us:du: 4
```

- 1 Points to the `us38400` entry specifying shared capabilities for modems
- 2 First line of the `us38400` entry
- 3 Defines end-of-line characters, and input and output end-of-file marks
- 4 Defines the device to open for the connection, the speed, the parity, the name of the `/etc/acucap` entry, and the dial-up line

You might use generic entries like these to connect to any number of remote systems.

Optionally, you can create an entry for each remote system you contact. Then you can include settings that are specific to those systems, for example, their phone numbers. See `remote(4)` for more information.

Local Area Transport Connections

The Local Area Transport (LAT) protocol supports communications between host computer systems and terminal servers with terminals, PCs, printers, modems and other devices over local area networks (LANs). The Tru64 UNIX LAT implementation is a STREAMS-based driver.

This chapter describes:

- The LAT implementation on Tru64 UNIX systems (Section 7.1)
- How to plan for your LAT configuration (Section 7.2)
- How to configure the LAT driver (Section 7.3)
- How to set up specific LAT connections (Section 7.4)

For additional introductory information on LAT, see `lat_intro(7)`. For troubleshooting information, see Section 8.9.

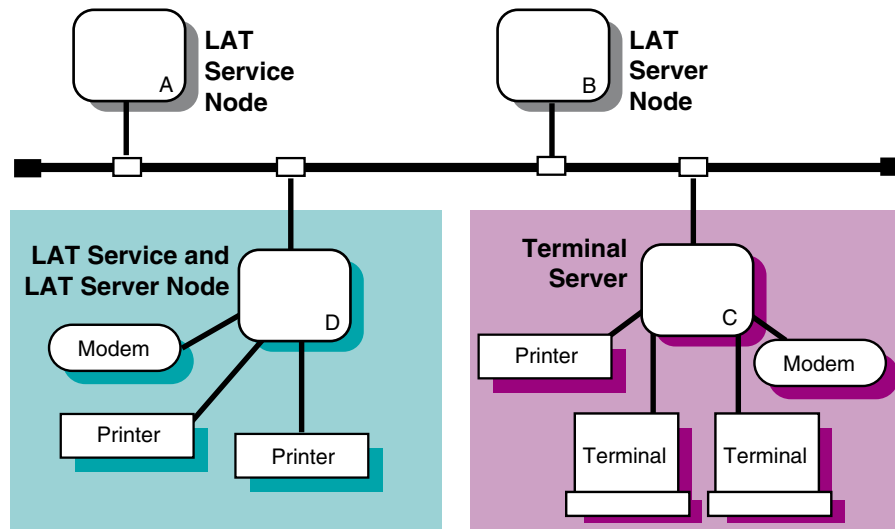
7.1 LAT Environment

In the LAT environment, systems can have the following roles:

- Service node — A system that offers LAT services to users on the LAN and accepts connections from server users.
- Server node — A terminal server or a system that is configured for outgoing connections. Server nodes enable users attached to the node to initiate LAT sessions through outgoing ports to LAT services offered by LAT service nodes.

Figure 7–1 shows a sample LAN with LAT server nodes and LAT service nodes.

Figure 7–1: Sample LAT Network Configuration



ZK-1179U-AI

The LAT software also permits host applications to initiate connections to server ports, designated as application ports, to access remote devices. The following sections describe:

- Types of LAT connections
- Access control in a LAT network
- Password specification for remote servers
- Load balancing

7.1.1 Types of LAT Connections

The following types of LAT connections are permitted:

- Terminal-to-host connections — The basic LAT connection in which a user at a terminal connected to a terminal server connects to a LAT service. For example, a user at a terminal connected to terminal server C and connecting to a service on host A in Figure 7–1 is using a terminal-to-host connection.
- Host-initiated connections — A connection in which a bit-serial, asynchronous device connected to a terminal server communicates with user-written applications on a LAT host. For example, a user who set up host A to use a printer on host D in Figure 7–1 is using a host-initiated connection.
- Outgoing connections — A connection in which a user on a LAT server node can connect to a LAT service by using the `llogin` command. For

example, a user on host B who connects to a LAT service on host A in Figure 7-1 is using an outgoing connection.

- Lattelnets gateway connections — A connection in which a user at a terminal connected to a terminal server connects to a remote host through an intermediate Tru64 UNIX host. For example, a user at a terminal connected to terminal server C who is connecting to the lattelnets service on host D in Figure 7-1 is using a lattelnets connection.

7.1.2 Controlling Access in a LAT Network

Because LAT networks are local in nature, you have a high degree of control over the LAT environment and who has physical access to LAT devices. In addition to controlling physical access, the following features enable you to control LAT access:

- LAT terminal server login password — You can require that users enter a password to gain access to terminal servers. (Refer to your terminal server documentation for more information.)
- LAT groups — You can establish LAT groups and restrict host communication to particular groups in the following cases:
 - On a LAT service node, by issuing a `latcp -g -a` command
 - On a LAT server node, by issuing a `latcp -u` command
 - On a terminal server (refer to your terminal server documentation for more information)

In general, groups are set up by the network manager, system manager, and server managers to partition the LAT network into logical subdivisions and to restrict message traffic between servers and service nodes. In addition, using groups can help you manage the size of the servers' LAT databases by limiting the number of service nodes for which the server keeps information.

Note

You can use groups to restrict access, but they are not intended as a security mechanism.

To establish a connection with a LAT service node, the group enabled on a terminal server port or an outgoing port on a LAT server node must match at least one group on the service node. Similarly, for a terminal server or server node to process messages from service nodes, the group enabled on a terminal server port or an outgoing port on the server node must match at least one group on the service node. Otherwise, the messages from the service nodes are ignored.

For more information on enabling LAT service node groups and outgoing port groups, refer to `latcp(8)`.

7.1.3 Specifying Passwords for Remote Services

The LAT protocol enables you to specify a password for access to remote services that are protected by a password. When password checking is enabled on a terminal server that offers a service that is password protected, you must specify the password when you map the application port; if you do not, all attempts to connect to the service from the terminal server are rejected. See `latcp(8)` for more information.

7.1.4 Load Balancing

When more than one node on a LAN offers the same service, the terminal server connects to the node with the highest rating for the service desired. The rating is based on the current load on the nodes that offer the service. This process is called load balancing.

Load balancing works in a heterogeneous environment. Therefore, service nodes with the same names may be running different operating systems.

7.2 Planning LAT

This section describes the tasks you must complete before configuring LAT.

7.2.1 Verifying That the LAT Subset Is Installed

Verify that the LAT subset is installed by entering the following command:

```
# setld -i | grep OSFLAT
```

If the LAT subset is not installed, install it by using the `setld` command. For more information on installing subsets, see `setld(8)` or the *Installation Guide*.

After the LAT subset is installed, reboot the system to load the LAT module into the kernel. The system is configured to dynamically load the LAT module into the kernel when the system boots.

7.2.2 Verifying DLB Support in the Kernel

After you install the LAT subset, verify that Data Link Bridge (DLB) support is in the kernel by issuing the following command:

```
# sysconfig -q dlb
```


If the `dlb:` prompt is not displayed, log in as superuser and complete the following steps:

1. Edit the configuration file and add the following entry to it:

```
options DLB
```

The default configuration file is `/sys/conf/HOSTNAME` where `HOSTNAME` is the name of your host processor, in uppercase letters.

2. Build a new kernel by issuing the `doconfig` command. If you are unfamiliar with rebuilding the kernel, see the *System Administration* manual.
3. Reboot your system with the new kernel by issuing the following command:

```
# shutdown -r now
```

This command immediately shuts down and automatically reboots the system.

7.2.3 Preparing for the Configuration

After you verify DLB support in the kernel, you can configure LAT by using the `latsetup` utility.

Figure 7–2 shows the LAT Setup Worksheet, which you can use to record the information required to configure LAT. If you are viewing this manual on line, you can use the print feature to print the worksheet. The following sections explain the information you need to record on the worksheet.

Figure 7–2: LAT Setup Worksheet

LAT Setup Worksheet	
Start LAT automatically at boot time:	Yes <input type="checkbox"/> No <input type="checkbox"/>
Type of tty devices:	_____
Number of LAT tty devices:	_____
Number of LAT entries (getty) in <code>/etc/inittab</code> :	_____

Start LAT automatically at boot time

By default, the `/sbin/init.d/lat` startup and shutdown script automatically starts LAT upon reaching run level 3 and stops LAT when exiting run level 3. If you do not want LAT to be started automatically, check No; otherwise, check Yes.

Type of tty devices

The type of terminal device (tty) for each LAT connection. Tru64 UNIX supports SVR4 and BSD device types. It is best to use SVR4 devices because the SVR4 format allows you to create more devices.

SVR4 device special files have the following format:

```
/dev/l $at$ / $n$ 
```

The value n is a number between 620 and 4370. For example, `/dev/l at /620`, `/dev/l at /777`, and `/dev/l at /4000` specify SVR4 devices.

BSD device special files have the following format:

```
/dev/t $ty$  $w$  $x$ 
```

The value w is a number from 0 to 9; x is an alphanumeric from 0 to 9, a lowercase a to z, or an uppercase A to Z. For example, `/dev/t ty 02`, `/dev/t ty 0e`, and `/dev/t ty 9f` specify BSD LAT terminal devices. However, all BSD terminal device names are not case sensitive. The device special files `/dev/t ty 9f` and `/dev/t ty 9F` are both converted to `TTY9F`.

This format enables you to specify up to 620 BSD terminal devices which are available to any serial devices (such as UUCP) running on the system. Therefore, fewer than 620 BSD devices might be available for LAT.

Number of LAT tty devices

The total of the desired number of simultaneous incoming LAT connections, the number of application ports, and the number of outgoing connections needed.

Number of LAT entries (getty) in /etc/inittab

The number of LAT `getty` entries to be added to the `/etc/inittab` file. This is the number of simultaneous incoming LAT connections desired.

7.3 Configuring LAT

This section describes how to perform the following tasks:

- Configure LAT with the `latsetup` utility
- Start and stop LAT manually
- Create a LAT startup file
- Customize the `inittab` file

- Run LAT over specific network adapters

7.3.1 Configuring LAT with `latsetup`

Use the `latsetup` utility to configure and administer LAT on your system. To use the `latsetup` utility, LAT and DLB must be configured into the running kernel, your system must be at run level 3 or 4, and you must be logged in as superuser. See `latsetup(8)` for more information.

The `latsetup` utility allows you to do the following:

- Create LAT device special files.
- Add or remove `getty` entries to or from the `/etc/inittab` file.
- Execute the `init q` command.
- Start or stop the LAT driver.
- Enable or disable LAT automatic startup and shutdown. When enabled, LAT starts automatically upon reaching run level 3.

You cannot configure LAT over NetRAIN virtual interfaces or the adapters that compose NetRAIN sets. LAT is not supported over NetRAIN.

From the SysMan Menu, invoke the `latsetup` utility by selecting Networking→Additional Network Services→Configure Local Area Transport (LAT). Alternatively, enter the following command on the command line:

```
# /usr/sbin/latsetup
```

If your terminal does not support curses, you must specify the `-nocurses` flag. This flag allows you to run the `latsetup` utility in command-line mode.

Note

Do not run multiple `latsetup` processes concurrently on the same machine. The `latsetup` user might receive erroneous information and the `/etc/inittab` file might become corrupted.

7.3.2 Starting and Stopping LAT

To manually start LAT, enter the following command:

```
# /sbin/init.d/lat start
```

To manually stop LAT, enter the following command:

```
# /sbin/init.d/lat stop
```

When you stop LAT from within a LAT session, the session will close.

7.3.3 Creating a LAT Startup File

If LAT automatic startup and shutdown are enabled, when the system reaches run level 3, it loads LAT into the kernel and executes the `/sbin/init.d/lat` script. This script reads and executes the `latcp` commands in the `/etc/latstartup.conf` file (if this file exists), then starts LAT. See `latcp(8)` for more information on the `latcp` command.

If you do not have an `/etc/latstartup.conf` file, LAT is started with the default values for its parameters. Table 7-1 lists the LAT parameters and their default values.

Table 7-1: LAT Parameters

Parameter	Default Value								
Node name	Host name								
Multicast timer	60 seconds								
Network adapter	All network adapters connected to broadcast media, except for NetRAIN virtual interfaces (<code>nr</code>) and those adapters that compose NetRAIN sets.								
Service name	From the LAT node name parameter. Each service has the following parameters:								
	<table border="1"><thead><tr><th>Parameter</th><th>Default Value</th></tr></thead><tbody><tr><td>Service description</td><td>"Compaq Tru64 UNIX Version <i>x.x</i> LAT SERVICE"</td></tr><tr><td>Rating</td><td>Dynamic</td></tr><tr><td>Group code</td><td>0</td></tr></tbody></table>	Parameter	Default Value	Service description	"Compaq Tru64 UNIX Version <i>x.x</i> LAT SERVICE"	Rating	Dynamic	Group code	0
Parameter	Default Value								
Service description	"Compaq Tru64 UNIX Version <i>x.x</i> LAT SERVICE"								
Rating	Dynamic								
Group code	0								
Agent status	Disabled								
Outgoing port groups	Group 0								
Maximum number of learned services	100								

If you want to customize LAT on your system, you can create and modify the `/etc/latstartup.conf` file to include `latcp` commands. For example, you can define a particular node name or add service names.

Note

If your system is a member of a cluster, you must create the `/etc/latstartup.conf` file as a Context-Dependent Symbolic Link (CDSL). See the *System Administration* manual for more information.

Example 7–1 shows a sample `/etc/latstartup.conf` file.

Example 7–1: Sample `/etc/latstartup.conf` File

```
/usr/sbin/latcp -n testnode      1
/usr/sbin/latcp -A -a lattelnet14 -i "LAT/telnet" -o 2
/usr/sbin/latcp -A -a testservice 3
/usr/sbin/latcp -g 0,21,52 -a testservice 4
/usr/sbin/latcp -A -a boundservice -p 620,621 5
/usr/sbin/latcp -c200 6
/usr/sbin/latcp -A -p 630 -O -V finance 7
/usr/sbin/latcp -u 0,1,41,97 8
/usr/sbin/latcp -e ln0 9
```

- 1 Changes the LAT node name.
- 2 Adds an optional service that can be used for LAT/Telnet connections. (See Section 7.4.4 for more information on the LAT/Telnet gateway.)
- 3 Adds an unbound interactive `testservice` service.
- 4 Adds groups 0, 21, and 52 to the `testservice` service.
- 5 Adds a bound service and binds to it two LAT devices: 620 and 621, which are SVR4-style LAT devices.
- 6 Increases the number of learned services to 200.
- 7 Maps an outgoing port to `finance` service.
- 8 Adds outgoing port groups 0, 1, 41, and 97.
- 9 Adds the `ln0` adapter.

A `latcp` command that adds a service must occur in the `latstartup.conf` file before you can issue a `latcp` command requiring the service name. Lines 3 and 4 in Example 7–1 illustrate this point.

7.3.4 Customizing the `inittab` File

You can modify the `/etc/inittab` file to use a program other than the `getty` program. For example, you can add the following entry to the `/etc/inittab` file to configure LAT device 620 to use the user-defined program `myownprogram`:

```
lat620:34:respawn:/usr/sbin/myownprogram /dev/lat/620
```

The previous example uses an absolute pathname for the device `/dev/lat/620`.

For more information on using user-defined programs with LAT, see Section 7.4.5. For more information on the `/etc/inittab` file and the `getty` utility, see `inittab(4)` and `getty(8)`.

You can also modify the `/etc/inittab` file to add LAT devices created manually after the initial configuration by adding an entry similar to the following:

```
lat621:34:respawn:/usr/sbin/getty lat/621 console vt100
```

The second field (34) specifies the run level in which the entries will be processed. In this example, the `getty` process is spawned at either run level 3 or 4. In addition, this example uses a relative pathname, `lat/621`.

7.3.5 Running LAT Over Specific Network Adapters

If your system is configured with multiple network adapters, by default the `latcp` program attempts to start the LAT protocol on all adapters that can support it (which excludes NetRAIN virtual interfaces and the adapters that compose NetRAIN sets). For adapters connected to different logical networks, this is probably desirable. However, for adapters connected to a single logical network, it is recommended that you run the LAT protocol over only one adapter. To specify the adapter, add the `latcp -e adapter` command to the `/etc/latstartup.conf` file. See `latcp(8)` for more information.

Use the `netstat -i` command to determine the adapters defined on your system.

7.4 Configuring LAT Connections

This section describes how to perform the following tasks:

- Set up printers to print through LAT
- Set up host-initiated connections
- Set up outgoing connections
- Set up the LAT/Telnet gateway
- Create dedicated or optional services
- Provide a dedicated tty device on a terminal

7.4.1 Setting Up Printers

The following sections describe how you can set up a printer to print through LAT. Once the printer is properly configured, local LAT hosts can access the printer through host-initiated connections, as described in Section 7.4.2.

This manual provides information on how to establish the LAT service. It does not contain all of the details of printer setup. For more information on setting up printers, see the *System Administration* manual, `printconfig(8)`, `lprsetup.dat(4)`, and `lprsetup(8)`.

In addition, before you start, you need to collect the following information:

- The name of the terminal server to which the printer will be attached
- Either or both of the following:
 - The name of the port to which the printer will be attached
 - The name of the service assigned for the remote printer
- Terminal server documentation
- Printer documentation

Note

The examples in this section use the DECserver 700 server. Please refer to the documentation supplied for your terminal server.

7.4.1.1 Setting Up the Printer on a Terminal Server

To set up a printer, do the following:

1. Connect the printer to a serial interface on a terminal server.
2. Use the terminal server commands specified in the terminal server documentation to set up the server to allow access to the attached remote printer through host-initiated requests from the service node. (Service node refers to the local Tru64 UNIX LAT host.)
3. Use the printer documentation to determine your printer's character size, flow control, parity, and speed.
4. Compare the printer's characteristics to the terminal server's port settings. You can display the settings on the terminal server console by entering a command similar to the following:

```
Local> SHOW PORT 7 CHARACTERISTICS
```

This command displays the characteristics for port 7. Minimally, the terminal server should have settings for the port similar to the following:

Character Size:	Printer's character size
Flow Control:	XON (or -CTS/RTS, for some printers)
Speed:	Printer's speed
Access:	Remote
Autobaud:	Disabled
Autoconnect:	Disabled

If the terminal server's port settings do not match the printer's characteristics, define the terminal server's port settings by using the `DEFINE` command. For example:

```
Local> DEFINE PORT 7 SPEED 9600
```

5. After you define the settings for the port, log out of that port to initialize the new settings. For example:

```
Local> LOGOUT PORT 7
```

7.4.1.2 Testing the Port Configuration

To verify that the printer characteristics match in the printer and in the terminal server port, use the `TEST PORT` command on the terminal server. For example, if the configuration is correct, the following command run on a DECserver 700 prints a test pattern of characters on a printer attached to port 7:

```
Local> TEST PORT 7
```

The printer prints 24 lines of test data unless you press the Break key at the terminal server console. If data does not print or if it is incorrect, the port or the printer is incorrectly set, or there is a hardware problem.

7.4.1.3 Setting Up a Service Node for the Printer

On the service node (local LAT host), use the `latcp` command to map an unused application port with the remote port or remote service on the terminal server. Use the terminal server name and either the name of the port or the name of the service for the printer from Section 7.4.1.1.

For example, the following command maps the local application port 621 for the server `LOC SER` to the remote printer port `port07`.

```
# latcp -A -p 621 -H LOC SER -R port07
```

The following command specifies the remote printer service name instead of the remote print port:

```
# latcp -A -p 621 -H LOC SER -V REMprinter07
```

For more information, see `latcp(8)`.

7.4.1.4 Setting Up the Print Spooler on the Service Node

To set up the print spooler for the remote printer, use the `lprsetup` command. The following symbols must be set in the `printcap` file for the service node (local LAT host) to access the remote printer through host-initiated connections:

- `ct` — Connection type

- `lp` — Device name to open for output

The following example shows an `/etc/printcap` entry for a LAT printer:

```
lp25|lp0:\
:af=/usr/adm/lpacct:\
:ct=LAT:\ 1
:lf=/usr/adm/lperr:\
:lp=/dev/lpt/621:\ 2
:mx#0:\
:of=/usr/lbin/lpf:\
:sd=/usr/spool/lpd:
```

- 1 Specifies LAT for the `ct` symbol.
- 2 Specifies the LAT application port (tty device) that was used in the `latcp` command to set up the service node. You must specify the full path name for the `lp` symbol.

7.4.1.5 Testing the Printer

After you set up the printer, print a file to ensure everything works properly. For example, if the printer name is `lp25` and `test` is a text file, you can test the printer by issuing the following command:

```
# lpr -Plp25 test
```

If the printer does not work, verify that all the settings are correct. If the `printcap` file entry has an `lf` symbol defined, you can check the corresponding log file for error information.

7.4.2 Setting Up Host-Initiated Connections

A host-initiated connection is one in which any bit-serial, asynchronous device connected to a terminal server can communicate with user-developed applications on an appropriately configured system. Examples of such devices are terminals, modems, communications ports on other host computer systems, and printers. Printer connections are discussed in Section 7.4.1.

This section describes how you set up a system for host-initiated connections and provides guidelines for developing applications to take advantage of these connections.

7.4.2.1 Setting Up the System for Host-Initiated Connections

To set up your system for LAT host-initiated connections, do the following:

1. Use the `latcp -A -p` command to map an application port (tty device) on the system with a remote port or service on a terminal server. In the

following example, 623 is the application port, T1301A is the terminal server name, and PORT_6 is the terminal port name.

```
# /usr/sbin/latcp -A -p 623 -HT1301A -R PORT_6
```

Alternatively, you can specify a service name instead of a port name in this example.

2. Make sure the protection bits, the owner, and the group of the tty device are set appropriately for the intended use of the connection. If ordinary users will open and read the tty device, make the device world readable.
3. Set up the server port characteristics to match the characteristics of the device connected to the port and to allow host-initiated connections. See your device and terminal server documentation for this information.

7.4.2.2 Program Interface

Applications that employ host-initiated connections are much like applications for any tty device, with the following exceptions:

- The programs communicate with the LAT driver through the device special file. When the host program issues an open call on the LAT tty device, the LAT driver attempts to establish a connection to the target port or service on the target server. The driver reports success and failure codes in the `errno` variable.
- When the open call is successful, the user program issues read and write system calls to handle data transfers, and normal `ioctl` processing for the device control information.
- A `close` system call on the device terminates the LAT connection.

The `dial.c` application program in the `/usr/examples/lat` directory is an example of a program that can be used with host-initiated connections. To access this example, you must install the `OSFEXAMPLES` optional subset.

The Tru64 UNIX LAT implementation is a STREAMS-based tty design. When a LAT tty device is opened, the POSIX line discipline module `ldterm` is pushed onto the stream above the LAT driver. If your application does not need the additional processing provided by `ldterm`, it must remove the module from the stream.

The `lined.c` application program in the `/usr/examples/lat` directory demonstrates how terminal (tty) line disciplines are changed in a Clist-based tty and a STREAMS tty environment. To access this example, you must install the `OSFEXAMPLES` optional subset. Additionally, you can use the `strchg` command to change the STREAMS configuration of the user's standard input.

For more information, see `autopush(8)` and `strchg(1)`.

7.4.3 Setting Up Outgoing Connections

An outgoing connection is one in which a local user can connect to a service on a remote host by using the `llogin` command. To accomplish this, a named service on the remote host is associated with a terminal device special file on the local host. See `llogin(1)` and the *Command and Shell User's Guide* for information on the `llogin` command.

7.4.3.1 Setting Up the System for Outgoing Connections

To set up your system for LAT outgoing connections, do the following:

1. Map an outgoing port (tty device) on the system with a port or service on a remote system by using the `latcp -A -p` command. In the following example, 621 is the outgoing port and `REMOTE_SERVICE` is the service name on the remote node:

```
# /usr/sbin/latcp -A -p 621 -O -V REMOTE_SERVICE
```

Alternatively, you can specify a remote node name and a port name, as in this example, where `titan` is the node and `PORT_1` is the port:

```
# /usr/sbin/latcp -A -p 621 -O -H titan -R PORT_1
```

2. Verify that the remote service is a learned service available to your system, by using the following command:

```
# /usr/sbin/latcp -d -l
```

If the service is not displayed, the maximum number of learned services has been reached; the service might still be available. When an outgoing connection is attempted, the local host determines whether the remote service is available. If it is available, the outgoing LAT connection is made.

To increase the maximum number of learned services, use the `latcp -c` command. See `latcp(8)` and `lat_intro(7)` for more information on learned services.

7.4.3.2 Program Interface

Applications developed to employ outgoing connections adhere to the same guidelines as applications developed for host-initiated connections. See Section 7.4.2.2 for more information.

The `getdate.c` application program in the `/usr/examples/lat` directory is an example of a program that can be used with outgoing connections. To access this example, you must install the `OSFEXAMPLES` optional subset.

7.4.4 Setting Up the LAT/Telnet Gateway

The LAT/Telnet gateway service enables a user on a LAT terminal server to connect to remote hosts running the Telnet protocol through an intermediate Tru64 UNIX host. The user does not have to log in to the local Tru64 UNIX system first. Optionally, if configured, you can use the `rlogin` command to connect directly to remote hosts.

To set up the LAT/Telnet gateway, perform the following steps:

1. Define the LAT/Telnet service by using the `latcp` command. For example:

```
# /usr/sbin/latcp -A -a lattelnet -i "LAT/telnet gateway" -o
```

The `-o` flag specifies that this is an optional service. Optional services are used with specialized applications that are written especially for LAT. These services are bound to LAT tty devices for the exclusive use of the specialized applications.

2. Edit the `/etc/inittab` file and modify the LAT device entries that you want to spawn the `lattelnet` service you created in step 1. The LAT terminals you select are dedicated to the gateway. The number of terminals selected determines the maximum number of simultaneous LAT/Telnet gateway sessions the system can deliver. For example, the following example shows LAT/Telnet gateway entries for three devices, which means that this system can deliver three simultaneous sessions:

```
lat624:34:respawn:/usr/sbin/lattelnet lat/624 lattelnet
lat625:34:respawn:/usr/sbin/lattelnet lat/625 lattelnet
lat626:34:respawn:/usr/sbin/lattelnet lat/626 lattelnet
```

If you want to use the `rlogin` command instead of Telnet, specify `/usr/bin/rlogin` as the third argument to the `lattelnet` program in the `/etc/inittab` entry. For example:

```
lat624:34:respawn:/usr/sbin/lattelnet lat/624 lattelnet /usr/bin/rlogin
```

3. Use the `init` program to read the `inittab` file and start the gateway by using the `init q` command.
4. Verify that the `lattelnet` process has started by using the `ps` command.

The `lattelnet` program uses the `syslogd` daemon to log messages to the `/var/adm/syslog.dated/date/daemon.log` file. Check this file to verify that no error messages were generated.

5. Connect to the gateway from the LAT terminal server by entering the `CONNECT` command. For example, to connect to a remote node named `REMOTE` by using a local node named `LOCAL` as a gateway, enter:

```
Local> CONNECT LATTELNET NODE LOCAL DEST REMOTE
```

You can use this command line for either Telnet or rlogin.

Alternatively, if connecting for Telnet, you can enter the service name LATTELNET and wait to be prompted for the remote node desired. The following example shows what occurs when a user on a terminal server connects to the service LATTELNET and waits for a login prompt from remote node MYTRIX:

```
Local> CONNECT LATTELNET
LAT to TELNET gateway on printf
telnet> OPEN MYTRIX
Trying...
Connected to mytrix.
Escape character is '^]'.
mytrix login:
```

7.4.5 Creating Dedicated or Optional Services

Dedicated services can be used in combination with your own specialized applications. The following specialized application programs are provided in the `/usr/examples/lat` directory:

- `latdate.c` — Provides a user with the date and time
- `latdlogin.c` — Provides a LAT/DECnet gateway for logging in over DECnet

Setting up a dedicated service is similar to setting up the LAT/Telnet gateway. (See Section 7.4.4.) To set up a dedicated service, complete the following steps:

1. Log in as root.
2. After you enter and compile the application code, copy the executable to the directory of your choice.
3. Add the service by using the `latcp -A -a` command. For example:

```
# /usr/sbin/latcp -A -a showdate -i "LAT/date service" -o
```

The `-o` specifies that this is a dedicated service.

4. Edit the `/etc/inittab` file and add the dedicated tty device entries. For example:

```
lat630:3:respawn:/usr/sbin/latdate lat/630 showdate
```

Note

You need an entry in the `/etc/inittab` file for every simultaneous service you want to run. The previous example

allows for only one user of the `latdate` service at any one time.

5. Use the `init` program to read the `inittab` file and start the service by using the `init q` command.

To use the service at a LAT terminal, issue the `CONNECT` command. For example:

```
Local> CONNECT SHOWDATE
```

A Tru64 UNIX host can also offer bound interactive and unbound interactive services. See `lat_intro(7)` for more information. For information on the commands used to create these services, see `latcp(8)`.

7.4.6 Providing a Dedicated tty Device on a Terminal

A terminal connected to a terminal server port can offer a dedicated tty device on a given Tru64 UNIX LAT host. This configuration is useful when the terminal user needs access to a specific application (for example, a database) on the host, but must not be allowed to access other applications or hosts for security reasons.

Once configured, the terminal will always be connected to the specified tty device on the LAT host. The user at the terminal cannot switch sessions or connect to different hosts or different tty devices on that host.

7.4.6.1 Setting Up a Dedicated tty Device

To set up a dedicated tty device on a terminal, perform the following steps:

1. Determine the name of the terminal server and the port name on which the terminal is connected. The following terminal server commands display the name of the server and the port name, respectively:

```
Local> SHOW SERVER
Local> SHOW PORT number
```

The *number* variable is the number of the port on the terminal server.

2. On the LAT host, map an application port (tty device) to the port on the terminal server by using the `latcp -A -p` command. For example, the following command maps an SVR4 device (application port 630 to port 2 on the terminal server LATTERM):

```
# latcp -A -p630 -H LATTERM -R PORT_2
```

For more information, see `latcp(8)`.

3. On the LAT host, add a `getty` entry to the `/etc/inittab` file for the tty device that was mapped as an application port. For example:

```
lat630:34:respawn:/usr/sbin/getty          lat/630 console vt100
```

4. On the terminal server, define the port's access to be **REMOTE** and log out from the port. For example:

```
Local> DEFINE PORT 2 ACCESS REMOTE
Local> LOGOUT PORT 2
```

5. Press Return on the terminal connected to the terminal server port that you just set up. When the system prompt is displayed, the terminal is connected to the dedicated tty device.

If you need to repeat the procedure, remove the `getty` entry from the `/etc/inittab` file, issue the `init q` command, and start the procedure from the beginning.

7.4.6.2 Removing a Dedicated tty Device

To remove a dedicated tty device from a terminal port and allow the terminal connected to the port to connect to any host, do the following:

1. Log in to another terminal on the same server.
2. Set the port's access to **LOCAL** and log out from the port. For example:

```
Local> DEFINE PORT 2 ACCESS LOCAL
Local> LOGOUT PORT 2
```

3. Unmap the application port and remove the `getty` entry from the `/etc/inittab` file.

8

Solving Network and Network Services Problems

This chapter contains a diagnostic map to help you solve problems that might occur when you use the network and network services software. Use this chapter together with the appropriate Compaq documentation to solve as many problems as possible at your level.

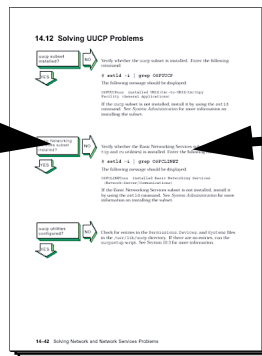
Section 8.1 and Section 8.2 provide information about how to use the diagnostic map and where in the map to start for certain problems. The sections that follow contain portions of the diagnostic map. They describe how to solve problems related to the following types of connections:

- IPv4 (Section 8.3)
- IPv6 (Section 8.4)
- ATM (Section 8.5)
- DHCP (Section 8.6)
- SLIP (Section 8.7)
- PPP (Section 8.8)
- LAT (Section 8.9)

8.1 Using the Diagnostic Map

Network and network service problems can occur for a number of reasons. The diagnostic map in this chapter and a similar diagnostic map in *Network Administration: Services* help you to isolate the problem. The following figure explains how to use the diagnostic map:

The left-hand column asks questions about the status of specific events that occur as you use the system.



The right-hand column diagnoses negative responses to those questions.

After you isolate the problem, the map refers you to other chapters for instructions on using the various problem solving tools and utilities. The map also refers you to other manuals for more complete diagnostic information for particular devices and software products.

You could experience problems that are not documented in this manual when you use base system network and network services software with other layered products. See the documentation for the other products for additional information.

8.2 Getting Started

Before you start problem solving, ensure that the communications hardware is ready for use. Verify the following:

- The system's physical cable connections (the Ethernet connection and the transceiver connection) are properly installed. See the documentation for your system and communications hardware device.
- Event logging is enabled in order to monitor network events. See the *System Administration* manual for information on starting event logging and for descriptions of the event messages.

Also see the product release notes for up-to-date information on known problems.

For solving IPv6 network problems, you must also be familiar with the following terms before you start problem solving:

on-link node

An on-link node is attached to the same subnetwork as your system. This subnetwork can be a LAN, a serial connection running PPP, or an IPv6 over IPv4 configured tunnel. There are no IPv6 routers between your system and the on-link node. For the configured tunnel, the on-link node is the node at the destination end of the tunnel.

off-link node

An off-link node is not attached to the same subnetwork as your system. There is at least one IPv6 router between your system and the off-link node.

In Figure 3–4, if your system were Host A, Host B is an on-link node, and Host C and Host D are off-link nodes.

Table 8–1 helps you identify a starting point in the diagnostic map.

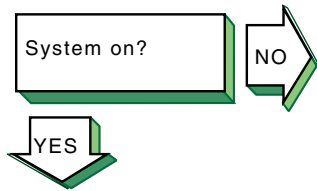
Table 8–1: Problem Solving Starting Points

If your problem is:	Start here:
uucp command error	Solving UUCP Problems section in <i>Network Administration: Services</i> .
Network command error	Section 8.7, if using a SLIP connection Section 8.8, if using a PPP connection Section 8.3 Section 8.4
Connecting to an ATM network	Section 8.5 Section 8.5.1, if using Classical IP Section 8.5.2, if using LANE Section 8.5.3, if using IP switching Section 8.3 Section 8.4
Obtaining an IP address using DHCP	Section 8.6 Section 8.3 Section 8.4
Correcting system time when you are using NTP	Solving NTP Problems section in <i>Network Administration: Services</i> .
Getting host name information	Solving DNS Client Problems section in <i>Network Administration: Services</i> , if you are using DNS/BIND. Solving NIS Client Problems section in <i>Network Administration: Services</i> , if you are using NIS.
Accessing files	Solving NFS Client Problems section in <i>Network Administration: Services</i> , if you are using NFS. Section 8.3 Section 8.4
Connecting to a host using LAT	Section 8.9
Unknown errors	Section 8.3

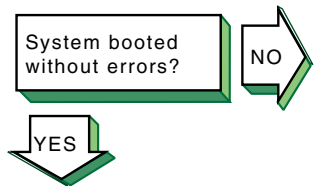
Table 8–1: Problem Solving Starting Points (cont.)

If your problem is:	Start here:
Unknown IPv6 errors	Section 8.4
Sending or receiving mail	Solving Mail Problems section in <i>Network Administration: Services</i> . Solving POP/IMAP Problems section in <i>Network Administration: Services</i> , if you are using POP or IMAP mail.

8.3 Solving IPv4 Network Problems



Turn on the power to your system. See the system manual for your system's startup procedure and any problem solving information.



If you are running Network Information Service (NIS) and your system hangs after the NIS daemons are started and before it mounts remote file systems, no NIS server is available to respond to the `ybind` request. If you know there is an NIS server for your domain, wait until the server responds; the boot procedure will continue.

If there is a Local Area Transport (LAT) problem, the following message is displayed:

```
getty: cannot open "/dev/ttyxx"
```

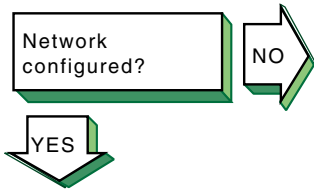
See the steps for solving LAT problems in Section 8.9.

If your system is a Network File System (NFS) client and it hangs while mounting a remote file system or directory, complete the following steps:

1. Inspect the cable and connection between your system and the network.
2. Wait until all the servers listed in the `/etc/fstab` file are available on the network; your system will then continue booting.
3. If you want your system to continue booting even if an NFS server is down, do the following:
 - a. Halt the system.
 - b. Boot the system to single-user mode and run the `fsck` command on the local file systems.
 - c. Edit the `/etc/fstab` file and add the `bg` (background) option to the server entries. See `fstab(4)` and `mount(8)` for more information.
 - d. Reboot the system with the following command:

```
# /sbin/reboot
```

If the `bg` option is specified in the `fstab` file entry, the remote file system or directory is automatically mounted when the server is running and begins functioning as an NFS server.

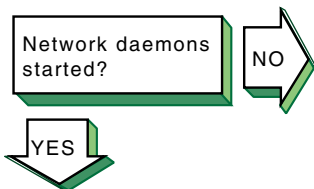


Follow these steps to see if your network is configured:

1. If your system is new to this environment and you recently configured it for use on a network, verify that the network adapter mode is set correctly at the console level. For example, if you have a 10base2 Ethernet network and your system is configured to use 10baseT Ethernet, your system fails to see the network until you set the appropriate console variable. See the prerequisite tasks for a full installation in the *Installation Guide* for more information.
2. Use the `rcmgr` utility to display the value of the `NUM_NETCONFIG` entry in the `/etc/rc.config` file:

```
# rcmgr get NUM_NETCONFIG
```

If the value is 0, run the SysMan Menu utility to configure your network. See Section 2.3 for more information.

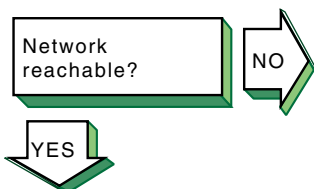


Verify that the network daemon (`inetd`) is running. Enter the following command:

```
# ps -e | grep inetd
```

If no `inetd` daemon is running, start it, using the following command:

```
# /sbin/init.d/inetd start
```

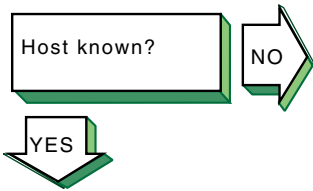


If a remote host's network is not reachable, the following message is displayed:

```
network is unreachable
```

Complete the following steps:

1. Ensure that the network devices are configured properly on the local host, using the `netstat -i` command. See Section 2.3 for information on configuring network devices.
2. Verify that the routing tables on the local host are correct, using the `netstat -r` command.
3. Trace the path looking at each Internet Protocol (IP) router's routing tables to find an entry for the remote host's network. Repair the incorrect IP router's routing tables. (This step requires a thorough knowledge of your topology.)
4. Verify that the local host's address-to-name translation for the remote host is correct. See the solutions for Host known?.
5. Inspect the routers along the path to the remote host to determine whether they have security features enabled that prevent you from reaching the remote host.

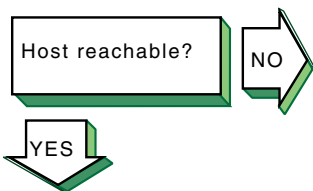


If a remote host is not known, the following message is displayed:

```
unknown host
```

Complete the following steps:

1. Verify that the user is trying to reach the remote host using a valid host name.
2. Verify that the remote host is in another name domain and that the user specified the full domain name.
3. If your site uses the Domain Name System (DNS) for name-to-address translation, look in the `/etc/svc.conf` file to see if `bind` is specified as a service for the `hosts` database entry. If it is not, edit the file and add it. Also, verify that the DNS service has information about the remote host. See the steps for solving DNS/BIND client problems in *Network Administration: Services*.
4. If your site uses NIS name service for name-to-address translation, look in the `/etc/svc.conf` file to see if `yp` (NIS) is specified as a service for the `hosts` database entry. If it is not, edit the file and add it. Also, verify if the NIS service has information about the remote host. See the steps for solving NIS client problems in *Network Administration: Services*.
5. If your `/etc/svc.conf` file lists `local` as the only name-to-address translation mechanism, the `/etc/hosts` file does not have information on the remote host. See *System Administration* for more information.



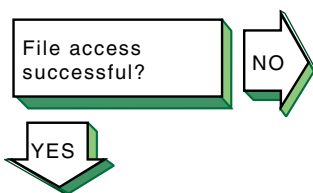
If a remote host is not reachable, the following message is displayed:

```
host is unreachable
```

Complete the following steps:

1. Inspect the cabling between the local host and the network.
2. Verify that the remote host is running, using the `ping` command.
3. Make sure that the network devices are configured properly on the local host, using the `netstat -i` command. See Section 2.3 for information on configuring network devices.
4. Verify that the routing tables on the local host are correct, using the `netstat -r` command. Use the `ping` command to determine whether the IP router is reachable.
5. Verify that the local host's address-to-name translation for the remote host is correct. See the solutions for Host known?.

6. Inspect the routers along the path to the remote host to determine whether they have security features enabled that prevent you from reaching the remote host.



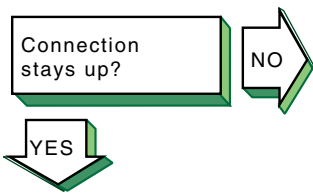
If a file cannot be accessed using the `rcp` or `rsh` commands, the following message is displayed:

```
permission denied
```

Complete the following steps:

1. Verify that the user is intended to have access to the remote host. The remote host might be intentionally preventing remote access.
2. Verify that the correct host and user definitions exist in the user's `.rhosts` file on the remote host.
3. Verify that the `/etc/hosts.equiv` file is set up correctly.
4. Verify that the directory and file protection on the files to be copied or the `.rhosts` file on the remote system are correct.

If you are using NFS, see *Network Administration: Services* for NFS troubleshooting information.



If the connection is broken, the following message is displayed:

```
connection timed out
```

Complete the following steps:

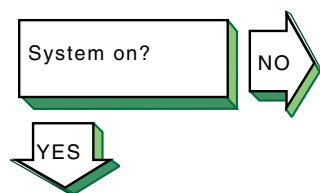
1. Test the network to determine whether the problem is on the local host, remote host, or a host on the path between the two. See Chapter 9 for more information on testing the network.
2. After you identify the host with the problem, do the following:
 - a. Confirm that the network device is properly configured. Verify that the broadcast address and address mask for the local host are correct. See Section 2.3 for information on configuring network devices.
 - b. Make sure the local host's `/etc/hosts` file has the correct IP address for the local host.
 - c. Make sure the cabling from the local host to the network is intact and properly connected.
 - d. If connected over a local area network (LAN), verify that the Address Resolution Protocol (ARP) entries are correct and that the system is properly connected to the LAN.



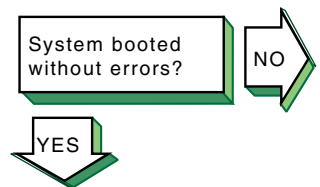
Problem still exists?
Report it to your service representative. See Chapter 10.

- e. If connected over a wide area network (WAN), verify that the system is properly connected to the WAN and that the modems are working properly.

8.4 Solving IPv6 Network Problems



Turn on the power to your system. See the system manual for your system's startup procedure and any problem solving information.



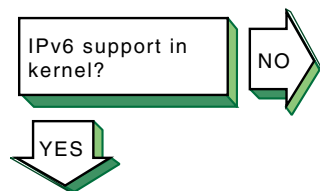
If network-related errors or warnings are displayed during boot, complete the following steps:

1. Disable IPv6 during system boot by issuing the following command:
2. Reboot the system. If the problems persist, go to Section 8.3.
3. Start IPv6 by issuing the following command:

```
# rcmgr set IPV6 "no"
```

```
# /usr/sbin/rcinet inet6
```

If the problems reappear, look in the `/etc/rc.config`, `/etc/ip6rtrd.conf`, and `/etc/routes` files for possible errors.



Verify that the IPv6 support you want is configured in the kernel. Enter the following command:

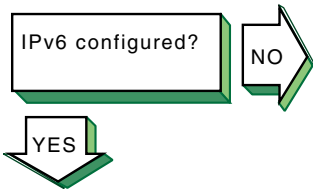
```
# sysconfig -s ipv6 | grep configured
```

If nothing is displayed, the IPv6 option is not configured in the kernel. Reconfigure the kernel by using the `doconfig` command. See Section 3.5.1 for more information.

If you want to use configured tunnels, verify that the IP tunneling support is configured in the kernel. Enter the following command:

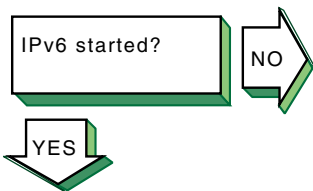
```
# sysconfig -s iptunnel | grep configured
```

If nothing is displayed, the `IPTUNNEL` option is not configured in the kernel. Reconfigure the kernel by using the `doconfig` command. See Section 3.5.1 for more information.



Verify that IPv6 is configured to start on system boot by issuing the `rcmgr get IPV6` command. If IPv6 is configured, the word `yes` is displayed.

If IPv6 is not configured, use the `ip6_setup` utility. See Section 3.6 for information on setting up an IPv6 host or router.



Verify that IPv6 was started by issuing the following command:

```
# ping ::1
```

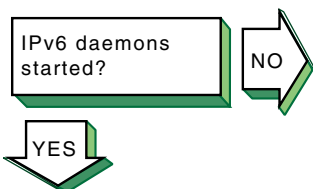
If the `host is unreachable` message is displayed, start IPv6 by issuing the following command:

```
# /usr/sbin/rcinet start inet6
```

This creates and brings up the IPv6 interfaces, and starts the IPv6 daemons.

Go to Section 8.4.1 for IPv6 host problems or Section 8.4.2 for IPv6 router problems.

8.4.1 Solving IPv6 Host Problems



Verify that the `nd6hostd` daemon is running by issuing the following command:

```
# ps ax | grep nd6hostd
```

If the daemon is not running, verify that your system is configured as an IPv6 host by issuing the following command:

```
# rcmgr get ND6HOSTD
```

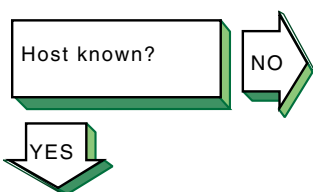
If the word `yes` is not displayed, run the `ip6_setup` utility and configure your system as an IPv6 host. Then, restart IPv6 with the following command:

```
# /usr/sbin/rcinet restart inet6
```

If the word `yes` is displayed, enable debugging for the `nd6hostd` daemon with the following command:

```
# rcmgr set ND6HOSTD_FLAGS "-d -1 /usr/tmp/nd6hostd.log"
```

Then, restart IPv6.



If a remote node is not known, the following message is displayed:

```
unknown host
```

Complete the following steps:

1. Verify that the user is using a valid node name to reach the remote node.

2. Verify that the remote node is in another name domain and that the user specified the full domain name.
3. If your site uses the DNS/BIND name service for name-to-address translation, look in the `/etc/svc.conf` file to see if `bind` is specified as a service for the `hosts` database entry. If it is not, configure your system as a DNS/BIND client. See *Network Administration: Services* for more information.

Verify that your system is running IPv4. If it is not, use the local `/etc/ipnodes` file for name-to-address translations.

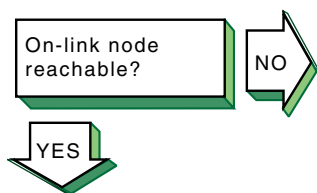
Also, verify that the DNS/BIND service has information about the remote node. See the steps for solving DNS/BIND client problems in *Network Administration: Services*.

4. If your site uses only NIS name service for name-to-address translation, you need to use another service for node names because NIS does not support IPv6 addresses.

Edit the `/etc/svc.conf` file and add either `bind` (DNS/BIND) or `local` (`/etc/ipnodes` file) as the service for the `hosts` database, depending on which service has the information about the remote node.

5. If your `/etc/svc.conf` file lists `local` as the only name-to-address translation mechanism, edit the `/etc/ipnodes` file and verify that the node name and address are present and accurate. Make any necessary additions or corrections.

Also, verify that there are no formatting errors in previous lines in the file. Beginning with the first entry, issue the `ping` command to each node to locate any formatting errors.



If an on-link node is unreachable, one of the following messages is displayed:

```

host is unreachable
network is unreachable
timeout
  
```

Verify that an on-link host or router, if one exists, is reachable by using the `ping` command. If the command fails or if there are frequently dropped packets, complete the following steps:

1. If the node is attached to a LAN, look at the datalink counters by using the `netstat -I device -s` command. The counters to examine and possible problems are as follows:
 - Zero blocks sent or received can indicate a network hardware failure or wiring problem.

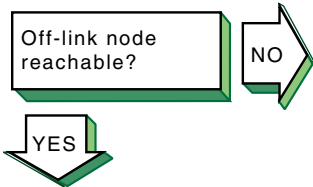
- High collision rates can indicate an improperly wired network or a node sending excessive message traffic.
 - Data overrun and buffer unavailable errors indicate your system is misconfigured.
2. Look at the IPv6 and ICMPv6 counters with the `netstat -p ipv6` and `netstat -p ipv6-icmp` commands, respectively. The counters and their possible causes:
 - Packets discarded due to error or errors generated due to ICMP errors indicate another node generating invalid messages. Other counters show more specific information.
 - Allocation errors can indicate excessive message traffic, a misconfigured system, or a program that repeatedly allocates memory without freeing it.
 3. Verify that IPv6 network interfaces exist, are running, and have `inet6` addresses by using the `ifconfig -a` command. If they do not, verify that the configuration variables in the `/etc/rc.config` file are correct. Run the `ip6_setup` utility to correct any errors.

Also, look for `nd6hostd` errors in the `/var/adm/syslog.dated/current/daemon.log` file. See Section 9.8 for more information.

If your interface does not have a global or site-local address, contact your network administrator to verify that your local router is advertising a prefix on the link. If there is no local router, you can define a prefix by using the `ifconfig` command (see Section 3.7.5).
 4. Contact the administrator of the on-link system and verify that the on-link system is up and running, that it is configured for IPv6 correctly, and that the address you are using is enabled on the node's interface.
 5. Issue the `ping` command to the on-link node's IPv4 address, if IPv4 is configured on both systems. If this succeeds, verify the IPv6 configuration on both systems. If the command fails, see the steps for solving IPv4 network problems in Section 8.3.
 6. Issue the `ping` command to other nodes on the link to determine whether the failure is confined to just one node or multiple nodes. Partial connectivity might indicate a faulty network device or cable on the link.
 7. If the link is a configured tunnel, do the following:
 - a. Verify the tunnel source and destination addresses by using the `ifconfig -a` command. Contact the administrator of the tunnel destination node and verify

that your source and destination addresses match the destination and source addresses on that node.

- b. Issue the `ping` command to the tunnel destination address. If the command fails, see the steps for solving IPv4 network problems in Section 8.3.



If an off-link node is not reachable, one of the following messages is displayed:

```
host is unreachable
network is unreachable
timeout
```

Verify that an off-link node is reachable by issuing the `ping` command. If there is 100% packet loss, complete the following steps:

1. Verify connectivity between your system and an on-link router by using the `ping` command. If the command fails or shows frequently dropped packets, follow the steps for `On-link node reachable?`. If you do not know the address to a router, issue the following command:

```
# ping -I interface ff02::2
```

2. Verify that the interface over which you are sending messages has a global or site-local unicast address enabled by using the `ifconfig -a` command. If it does not, contact your network administrator to verify that your local router is advertising a prefix on the link.

If the link is a configured tunnel and the router is not advertising an address prefix, manually define one for the tunnel by using the `ip6_setup` utility. See Section 3.6.1 for more information.

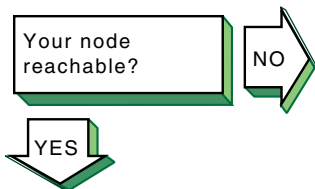
3. Contact the remote system's administrator to verify that the system is up and running, that it is configured for IPv6, and that the IPv6 address on its interface is the same one you are using. If the address is different, look in your system's `/etc/ipnodes` file or have the remote system administrator verify that the DNS entry is correct.
4. Verify that there is a default route (with U and G flags set) to a router on the network by issuing the `netstat -rf inet6` command. If there is not, contact the router administrator to verify that the router is advertising itself as a default router.

Also, look at other routers to see if your messages are being directed on the wrong path.

5. Trace the path to the off-link node by using the `traceroute` command. See Section 9.5 and `traceroute(8)` for more information.

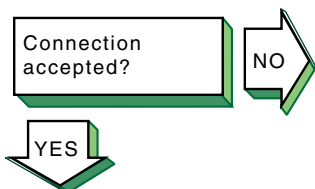
If there are frequently dropped packets, the problem might be network congestion or an intermittent routing problem. Do the following:

1. Verify connectivity between your system and an on-link router by using the `ping` command.
2. Trace the path to the off-link node by using the `traceroute` command. See Section 9.5 and `traceroute(8)` for more information.



If someone reports a problem reaching your node from another node, complete the following steps:

1. Verify that their node is reachable by issuing the `ping` command. If the command fails, follow the steps for `On-link node reachable?` or `Off-link-node reachable?`, depending on the location of the node.
2. If they are using a name from the DNS database, verify that the address for your node in the DNS database matches one of the addresses configured on your system's interfaces. Use the `nslookup -type=AAAA node-name` command to retrieve the address from DNS and the `ifconfig -a` command to display addresses for your system.
3. If they are using an address defined in their local `/etc/ipnodes` file, compare that address with the addresses configured on your system's interfaces. Use the `ifconfig -a` command.

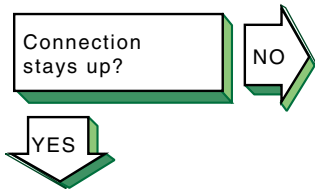


If a remote node is not configured to accept a connection from your application, the following message might be displayed:

```
connection refused
```

Contact the administrator of the remote node and ask if the correct socket-based service definitions are defined in the `/etc/services` and `/etc/inetd.conf` files. They might be missing or commented out.

Verify that the service in the local `/etc/inetd.conf` file has either `tcp6` or `udp6` in the protocol field.

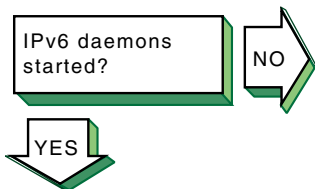


Problem still exists?
Report it to your service representative. See Chapter 10.

If the connection terminates abnormally or a network application appears to hang, complete the following steps:

1. Verify that there is network connectivity to the remote node by using the `ping` command immediately after the failure.
If the `ping` command fails or shows a high rate of packet loss, follow the steps for either `On-link node reachable?` or `Off-link node reachable?`, depending on the location of the remote node.
2. If your application transfers a large amount of data over the network, verify if large or fragmented messages are being handled correctly by using the `ping -s 2000 nodename` command.
If the `ping` command fails, trace the path to the remote node with 1200-byte packets by using the `traceroute nodename 1200` command. All IPv6 links must support message sizes of at least 1280 bytes. This command might show the location of the problem in the network. See Section 9.5 and `traceroute(8)` for more information.
3. Run the application with different client and server nodes located on different links in the network.

8.4.2 Solving IPv6 Router Problems



Verify that the `ip6rtrd` daemon is running by issuing the following command:

```
# ps ax | grep ip6rtrd
```

If the daemon is not running, verify that your system is configured as an IPv6 router by issuing the following command:

```
# rcmgr get IP6RTRD
```

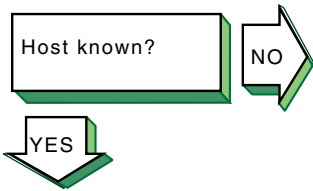
If the word `yes` is not displayed, run the `ip6_setup` utility and configure your system as an IPv6 router. Then, restart IPv6 with the following command:

```
# /usr/sbin/rcinet restart inet6
```

If the word `yes` is displayed, enable debugging for the `ip6rtrd` daemon with the following command:

```
# rcmgr set IP6RTRD_FLAGS "-d -l /usr/tmp/ip6rtrd.log /etc/ip6rtrd.conf"
```

Then, restart IPv6.



If a remote node is not known, the following message is displayed:

```
unknown host
```

Complete the following steps:

1. Verify that the user is using a valid node name to reach the remote node.
2. Verify that the remote node is in another name domain and that the user specified the full domain name.
3. If your site uses the DNS/BIND name service for name-to-address translation, look in the `/etc/svc.conf` file to see if `bind` is specified as a service for the `hosts` database entry. If it is not, configure your system as a DNS/BIND client. See *Network Administration: Services* for more information.

Verify that your system is running IPv4. If it is not, use the `/etc/ipnodes` file for name-to-address translation.

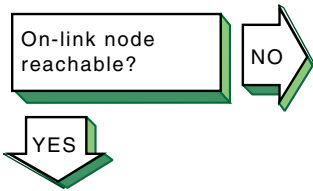
Also, verify that the DNS/BIND service has information about the remote node. See the steps for solving DNS/BIND client problems in *Network Administration: Services*.

4. If your site uses only NIS name service (`yp`) for name-to-address translation, you need to use another service for node names as NIS does not support IPv6 addresses.

Edit the `/etc/svc.conf` file and add either `bind` (DNS/BIND) or `local` (`/etc/ipnodes` file) as the service for the `hosts` database, depending on which service has the information about the remote node.

Also, verify that there are no formatting errors in previous lines in the file. Beginning with the first entry, issue the `ping` command to each node to locate any formatting errors.

5. If your `/etc/svc.conf` file lists `local` as the only name-to-address translation mechanism, edit the `/etc/ipnodes` file and verify that the node name and address are present and accurate. Make any necessary additions or corrections.



If an on-link node is not reachable, one of the following messages is displayed:

```
host is unreachable
network is unreachable
timeout
```

Verify that an on-link host or router, if one exists, is reachable by using the `ping` command. If the command fails or if there are frequently dropped packets, complete the following steps:

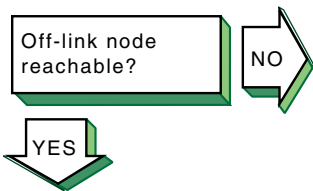
1. If the node is attached to a LAN, look at the datalink counters by using the `netstat -I device -s` command. The counters to examine and possible problems are as follows:
 - Zero blocks sent or received can indicate a network hardware failure or wiring problem.
 - High collision rates can indicate an improperly wired network or a node sending excessive message traffic.
 - Data overrun and buffer unavailable errors indicate your system is misconfigured.
2. Look at the IPv6 and ICMPv6 counters with the `netstat -p ipv6` and `netstat -p ipv6-icmp` commands, respectively. The counters to examine and possible problems are as follows:
 - Packets discarded due to error or errors generated due to ICMP errors indicate another node generating invalid messages. Other counters show more specific information.
 - Allocation errors can indicate excessive message traffic, a misconfigured system, or a program that repeatedly allocates memory without freeing it.
3. Verify that IPv6 network interfaces exist, are running, and have `inet6` addresses by using the `ifconfig -a` command. If they do not, verify that the `/etc/rc.config` and `/etc/ip6rtrd.conf` files are correct.

Also, look for `ip6rtrd` errors in the `/var/adm/syslog.dated/current/daemon.log` file. See Section 9.8 for more information.

Run the `ip6_setup` utility to correct any errors.
4. Contact the administrator of the on-link system and verify that the on-link system is up and running, that it is configured for IPv6 correctly, and that the address you are using is enabled on the node's interface.
5. Issue the `ping` command to the on-link node's IPv4 address, if IPv4 is configured on both systems. If this succeeds, verify the IPv6 configuration on both systems. If the command

fails, see the steps for solving IPv4 network problems in Section 8.3.

6. Issue the `ping` command to other nodes on the link to determine whether the failure is confined to just one node or multiple nodes. Partial connectivity might indicate a faulty network device or cable on the link.
7. If the link is a configured tunnel, do the following:
 - a. Verify the tunnel source and destination addresses by using the `ifconfig -a` command. Contact the tunnel destination node's administrator and verify that your source and destination addresses match the destination and source addresses on that node.
 - b. Issue the `ping` command to the tunnel destination address. If the command fails, see the steps for solving IPv4 network problems in Section 8.3.



If an off-link node is not reachable, one of the following messages is displayed:

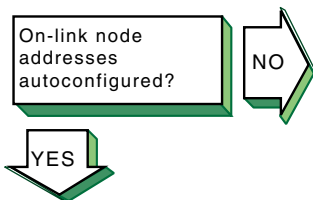
```
host is unreachable
network is unreachable
timeout
```

Verify that an off-link node is reachable by issuing the `ping` command. If there is 100% packet loss, complete the following steps:

1. Verify connectivity between your system and the next router in the path to the off-link node by using the `ping` command. If the command fails or shows frequently dropped packets, follow the steps for On-link node reachable?.
2. Verify that the interface over which you are sending messages has a global or site-local unicast address enabled by using the `ifconfig -a` command. If it does not, verify that the interface address prefixes defined in the `/etc/ip6rtrd.conf` file (see `ip6rtrd.conf(4)`) are correct. Run the `ip6_setup` utility to correct any prefix errors.
3. Contact the administrator of the remote system to verify that the system is up and running, that it is configured for IPv6, and that the IPv6 address on its interface is the same as the address that you are using. If the address is different, look in the hosts database.

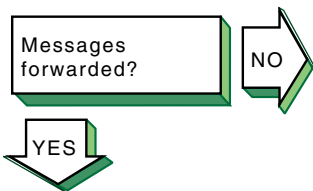
If there are frequently dropped packets, there might be network congestion or an intermittent routing problem. Do the following:

1. Verify connectivity between your system and an on-link router by using the `ping` command.
2. Trace the path to the off-link node by using the `traceroute` command. See Section 9.5 and `traceroute(8)` for more information.



IPv6 hosts generate their global and site-local unicast addresses automatically using address prefixes provided by a router on the link. If an on-link node cannot autoconfigure its addresses, complete the following steps:

1. Verify that the host is reachable from your router by using the `ping` command and specifying the host's link-local address. If the command fails or shows a high rate of packet loss, follow the steps for `On-link node reachable?`.
2. Edit the `/etc/ip6rtrd.conf` file and verify that the router is configured to advertise the correct prefixes and that the timers are reasonable. See Section 3.7.11 and `ip6rtrd.conf(4)` for more information.



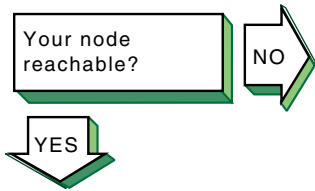
If another network user reports that message transmission appears to be failing at your router, complete the following steps:

1. Obtain the source and destination addresses of the message that your router is not forwarding. Then, verify that your router can reach each node by using the `ping` command. If either command fails or shows a high rate of packet loss, follow the steps for `On-link node reachable?` or `Off-link node reachable?`, as applicable.
2. If your router is running the RIPng protocol, verify that the IPv6 router daemon is running by issuing the following command:

```
# ps ax | grep ip6rtrd
```

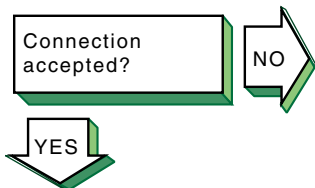
If it is running, edit the `/etc/ip6rtrd.conf` file and verify that the RIPng protocol is enabled on each IPv6 link. If it is not, your node might not be propagating routes correctly.

3. Make sure that you are not using manual routes on some interfaces and RIPng routes on other interfaces. Manual routes defined in the `/etc/routes` file do not get propagated to other routers with RIPng.



If someone reports a problem reaching your node from another node, complete the following steps:

1. Verify that their node is reachable by issuing the `ping` command. If the command fails, follow the steps for `On-link node reachable?` or `Off-link-node reachable?`, depending on the location of the node.
2. If they are using a name from the DNS database, verify that the address for your node in the DNS database matches one of the addresses configured on your system's interfaces. Use the `nslookup -type=AAAA node-name` command to retrieve the address from DNS and the `ifconfig -a` command to display addresses for your system.
3. If they are using an address defined in their local `/etc/ipnodes` file, compare that address with the addresses configured on your system's interfaces. Use the `ifconfig -a` command.

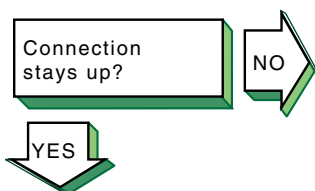


If a remote node is not configured to accept a connection from your application, the following message might be displayed:

```
connection refused
```

Contact the administrator of the remote node and ask if the correct socket-based service definitions are defined in the `/etc/services` and `/etc/inetd.conf` files. They might be missing or commented out.

Verify that the service in the local `/etc/inetd.conf` file has either `tcp6` or `udp6` in the protocol field.



If the connection terminates abnormally or a network application appears to hang, complete the following steps:

1. Verify that there is network connectivity to the remote node by using the `ping` command immediately after the failure. If the `ping` command fails or shows a high rate of packet loss, follow the steps for either `On-link node reachable?` or `Off-link node reachable?`, depending on the location of the remote node.
2. If your application transfers a large amount of data over the network, verify if large or fragmented messages are being handled correctly by using the `ping -s 2000 nodename` command.

If the `ping` command fails, trace the path to the remote node with 1200-byte packets by using the `traceroute nodename 1200` command. All IPv6 links must support message sizes of at least 1280 bytes. This command might

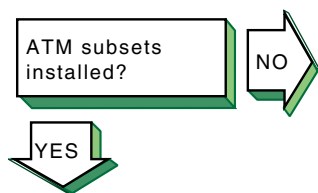


Problem still exists?
Report it to your service representative. See Chapter 10.

show the location of the problem in the network. See Section 9.5 and `traceroute(8)` for more information.

3. Run the application with different client and server nodes located on different links in the network.

8.5 Solving ATM Problems



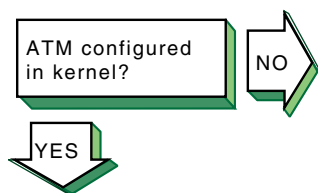
Verify that the ATM subsets are installed. Enter the following command:

```
# setld -i | grep OSFATM
```

The following messages are displayed:

```
OSFATMnnn installed ATM Commands  
  (Network-Server/Communications)  
OSFATMBINnnn installed ATM Kernel  
  Modules (Kernel Build Environment)  
OSFATMBINCOMnnn installed ATM Kernel  
  Header and Common Files  
  (Kernel Build Environment)  
OSFATMBINOBJECTnnn installed ATM Kernel  
  Objects (Kernel Software Environment)
```

If the OSFATM, OSFATMBIN, and OSFATMBINCOM subsets are not installed, install them by using the `setld` command. See the *Installation Guide* for information on installing the subset.

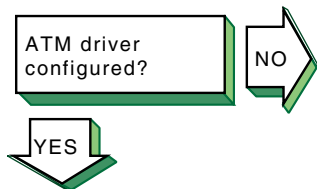


Verify that the ATM support you want is configured in the kernel. Enter the following command:

```
# sysconfig -q atm
```

If nothing is displayed, ATM is not configured in the kernel. Reconfigure the kernel with the ATM option and additional ATM options as needed. See Section 4.2.2 for a list of ATM kernel options and for information on reconfiguring the kernel.

If ATM is configured in the kernel, use the `sysconfig -q` command to verify that other ATM kernel options are configured. Reconfigure the kernel with additional options as needed.



Go to Section 8.5.1 for Classical IP, go to Section 8.5.2 for LAN Emulation, or go to Section 8.5.3 for IP switching.

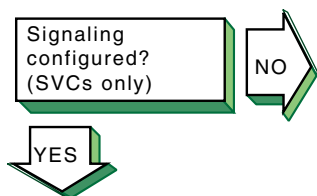
Verify that the driver is configured by using the `atmconfig drvlist` command. If the driver is configured, information similar to the following is displayed:

```
Name: lta0      Type: STS-3      State: UP
Driver ID: 1   ESIs: 8     PPAs: 9   VCs: 6
```

If an entry for the driver does not exist, use the `genvmunix` kernel to reboot the system and run the `doconfig` utility to build a kernel with the required driver.

If the driver state is not UP, run the `atmsetup` utility for the ATM service you want. See Section 4.3.2.4, Section 4.3.3.3, and Section 4.3.4.2 for information on configuring the driver for Classical IP (CLIP), LAN emulation (LANE), and IP switching, respectively.

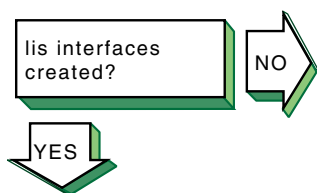
8.5.1 Solving CLIP Problems



Verify that signaling is configured. Enter the following command:

```
# atmstatus driver=driver_name
```

If the UNI version number is not displayed or the ILMI state is Unknown, run the `atmsetup` utility and configure signaling. See Section 4.3.2.4 for information.

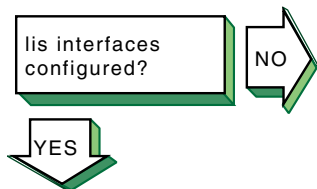


Verify that the CLIP `lis` interfaces are created. Enter the following command:

```
# atmarp -h
```

If a `lis` interface is created, the status of all created LISs and data indicating whether the host is an ARP client or ARP server is displayed.

If no LISs are created, run the `atmsetup` utility and configure CLIP. See Section 4.3.2.4 for more information.



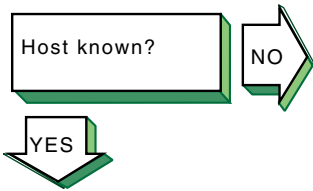
Verify that a `lis` interface is configured. Enter the following command:

```
# ifconfig lisx
```

If a `lis` interface is configured, information similar to the following is displayed:

```
lis0: flags=c23<UP,BROADCAST,NOTRAILERS,MULTICAST,SIMPLEX>
      inet 10.140.120.52 netmask ffffffff broadcast 10.140.120.255
      ipmtu 1500
```

If a `lis` interface is not configured, run the `netconfig` utility to configure one or use the Interfaces application from the SysMan Menu. See Section 4.3.2.5 for more information.



If a remote host is not known, the following message is displayed:

```
unknown host
```

Complete the following steps:

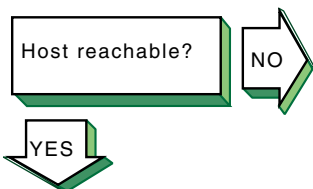
1. Verify that the user is using a valid host name to reach the remote host.
2. Verify that the remote host is in another name domain and that the user specified the full domain name.
3. If your site uses DNS for name-to-address translation, look in the `/etc/svc.conf` file to see if `bind` is specified as a service for the `hosts` database entry. If it is not, edit the file and add it.

Also, verify that DNS has information about the remote host. See the steps for solving DNS/BIND client problems in *Network Administration: Services*.

4. If your site uses NIS name service for name-to-address translation, look in the `/etc/svc.conf` file to see if `nis` is specified as a service for the `hosts` database entry. If it is not, edit the file and add it.

Also, verify that the NIS service has information about the remote host. See the steps for solving NIS client problems in *Network Administration: Services*.

5. If your `/etc/svc.conf` file lists `local` as the only name-to-address translation mechanism, the `/etc/hosts` file does not have information on the remote host. See *System Administration* for more information.

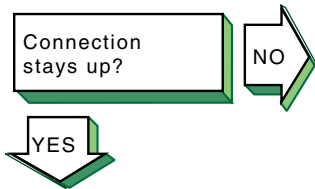


If a remote host is not reachable, the following message is displayed:

```
host is unreachable
```

Complete the following steps:

1. Verify that the cabling between the local host and the switch is properly installed and undamaged.
2. Verify that there is network connectivity to the IP controller on the switch by using the `ping` command. If the command fails, it might be because the `ifconfig` command parameters are wrong, or the IP controller is down or has an interface problem. Contact the switch administrator.
3. Verify that there is network connectivity to the target remote host by using the `ping` command. If the command fails, use the `traceroute` command to verify the route to the remote host.

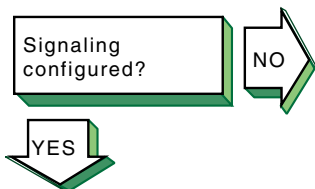


Problem still exists?
Report it to your service representative. See Chapter 10.

If the connection terminates abnormally, complete the following steps:

1. Test the network to determine whether the problem is on the local host, remote host, or a host on the path between the two. See Section 8.3.
2. After you identify the host with the problem, do the following:
 - a. Confirm that the network device is properly configured. Verify that the broadcast address and address mask for the local host are correct. See Section 2.3 for information on configuring network devices.
 - b. Make sure the local host's `hosts` database has the correct IP addresses.
 - c. Make sure the cabling from the local host to the network is intact and properly connected.
 - d. If connected over a LAN, verify that the ARP entries are correct and that the system is properly connected to the LAN.

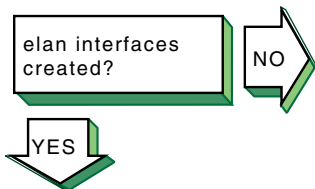
8.5.2 Solving LANE Problems



Verify that signaling is configured. Enter the following command:

```
# atmsig status driver=driver_name
```

If no User-Network Interface (UNI) version number is displayed or the Integrated Layer Management Interface (ILMI) state is Unknown, run the `atmsetup` utility and configure signaling. See Section 4.3.3.3 for information.



Verify that an `elan` interface is created. Enter the following command:

```
# atmelan show
```

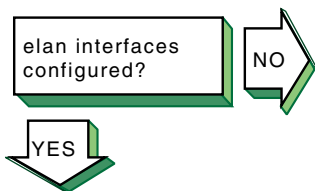
If an `elan` interface is created, information similar to the following is displayed:

```
⋮
control state: S_OPERATIONAL
⋮
```

If the control state is not `S_OPERATIONAL`, do the following:

1. Increase the message logging level for the `lane` subsystem. See Section 4.4.6 for more information.

2. Verify that the UNI version on the switch matches the UNI version on your system.
3. Verify that the LAN Emulation Server (LES) on the switch is configured correctly.



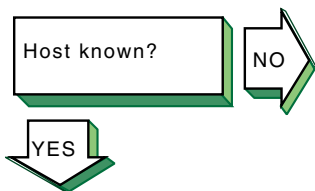
Verify that an `elan` interface is configured. Enter the following command:

```
# ifconfig elanx
```

If an `elan` interface is configured, information similar to the following is displayed:

```
elan0: flags=c23<UP,BROADCAST,NOTRAILERS,MULTICAST,SIMPLEX>
      inet 10.140.120.52 netmask ffffffff broadcast 10.140.120.255
      ipmtu 1500
```

If an `elan` interface is not configured, run the `netconfig` utility to configure one or use the Interfaces application from the SysMan Menu. See Section 4.3.3.4 for more information.



If a remote host is not known, the following message is displayed:

```
unknown host
```

Complete the following steps:

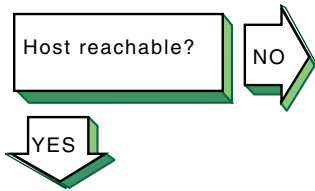
1. Verify that the user is using a valid host name to reach the remote host.
2. Verify that the remote host is in another name domain and that the user specified the full domain name.
3. If your site uses DNS for name-to-address translation, look in the `/etc/svc.conf` file to see if `bind` is specified as a service for the `hosts` database entry. If it is not, edit the file and add it.

Also, verify that DNS has information about the remote host. See the steps for solving DNS/BIND client problems in *Network Administration: Services*.

4. If your site uses NIS name service for name-to-address translation, look in the `/etc/svc.conf` file to see if `nis` is specified as a service for the `hosts` database entry. If it is not, edit the file and add it.

Also, verify that the NIS service has information about the remote host. See the steps for solving NIS client problems in *Network Administration: Services*.

5. If your `/etc/svc.conf` file lists `local` as the only name-to-address translation mechanism, the `/etc/hosts` file does not have information on the remote host. See *System Administration* for more information.

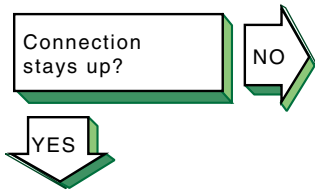


If a remote host is not reachable, the following message is displayed:

```
host is unreachable
```

Complete the following steps:

1. Verify that the cabling between the local host and the switch is properly installed and undamaged.
2. Verify that the addresses on the link are correct by using the `ifconfig elanx` command.
3. Verify that there is network connectivity to the target remote host by using the `ping` command. If the command fails, use the `traceroute` command to verify the route to the remote host.



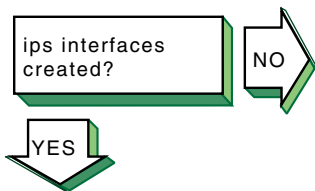
If the connection terminates abnormally, complete the following steps:

1. Test the network to determine whether the problem is on the local host, remote host, or a host on the path between the two. See Section 8.3.
2. After you identify the host with the problem, do the following:
 - a. Confirm that the network device is properly configured. Verify that the broadcast address and address mask for the local host are correct. See Section 2.3 for information on configuring network devices.
 - b. Make sure the local host's `hosts` database has the correct IP addresses.
 - c. Make sure the cabling from the local host to the network is intact and properly connected.
 - d. If connected over a LAN, verify that the ARP entries are correct and that the system is properly connected to the LAN.



Problem still exists?
Report it to your service representative. See Chapter 10.

8.5.3 Solving IP Switching Problems



Verify that an IP switching `ips` interface is created. Enter the following command:

```
# atmifmp showips
```

If an `ips` interface is created, information similar to the following is displayed for each created `ips` interface:

```
ips0:
    Attached to driver lta0
    Default (SNAP) VC = 32
    IP Traffic VC = 1850 (Unused - peer does
        not support Flow Type 0)
```

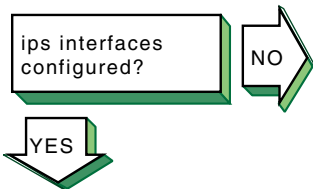
```

Min Tx VC = 1
Max Tx VC = 2048
Min Rx VC = 1
Max Rx VC = 2048
Driver Min Tx VC = 1
Driver Max Tx VC = 2048
Driver Min Rx VC = 1
Driver Max Rx VC = 2048
Peer does not support Flow Type 0

```

This example shows that the `ips0` interface was created and is attached to driver `lta0`.

If no `ips` interfaces are found, create one or more `ips` interfaces. See Section 4.3.4 for more information.



Verify that an `ips` interface is configured. Enter the following command:

```
# ifconfig ipsx
```

If an `ips` interface is configured, information similar to the following is displayed:

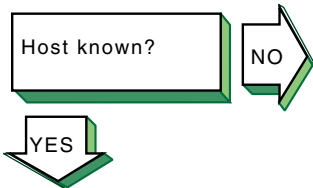
```

ips0: flags=4d1<UP,POINTOPOINT,RUNNING,NOARP,MULTICAST>
      inet 16.142.128.129 --> 16.142.128.130 netmask ffffffff ipmtu 1500

```

The example shows that the interface is up and running and that addresses are configured for each end of the point-to-point link.

If an `ips` interface is not configured, run the `netconfig` utility to configure one or use the Interfaces application from the SysMan Menu. See Section 4.3.4.3 for more information.



If a remote host is not known, the following message is displayed:

```
unknown host
```

Complete the following steps:

1. Verify that the user is using a valid host name to reach the remote host.
2. Verify that the remote host is in another name domain and that the user specified the full domain name.
3. If your site uses the DNS for name-to-address translation, look in the `/etc/svc.conf` file to see if `bind` is specified as a service for the `hosts` database entry. If it is not, edit the file and add it.

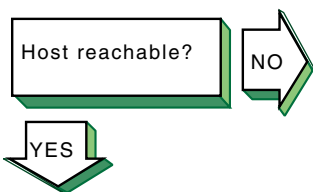
Also, verify that DNS has information about the remote host. See the steps for solving DNS/BIND client problems in *Network Administration: Services*.

4. If your site uses NIS name service for name-to-address translation, look in the `/etc/svc.conf` file to see if `nis` is

specified as a service for the `hosts` database entry. If it is not, edit the file and add it.

Also, verify that the NIS service has information about the remote host. See the steps for solving NIS client problems in *Network Administration: Services*.

5. If your `/etc/svc.conf` file lists `local` as the only name-to-address translation mechanism, the `/etc/hosts` file does not have information on the remote host. See *System Administration* for more information.

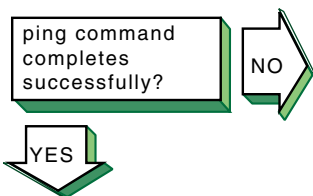


If a remote host is not reachable, the following message is displayed:

```
host is unreachable
```

Complete the following steps:

1. Verify that the addresses on the point-to-point link to the switch are correct by using the `ifconfig ipsx` command.
2. Verify the connection to the IP controller on the switch by using the `ping` command. If the command fails, the local host's `ifconfig` command parameters might be incorrect. On the switch, the problem might be that the IP controller is down or has an interface problem. Contact the switch administrator.
3. Verify that there is an `ips` route to the remote host's subnet by using the `netstat -r` command.

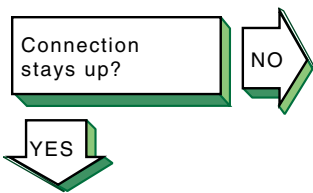


If the `ping` command fails, complete the following steps:

1. Verify that the cabling between the local host and the switch is properly installed and undamaged.
2. Verify that the default Subnetwork Attachment Point (SNAP) virtual circuit (VC) specified on the local host matches the default SNAP VC on the switch.
3. Contact the administrator of the remote system and verify that the remote system is up and running and that it is configured correctly for IP switching.
4. Verify the route to the remote host by using the `traceroute` command. If the first hop in the output shows the default network interface and not the IP controller, add a static route to the remote subnet through the IP controller to your routing table. Use the `netstat -r` command to verify the change.

If the route reaches the IP controller but goes no further, contact the administrator of the remote system to verify

that the system is configured correctly and that the routing tables are correct.



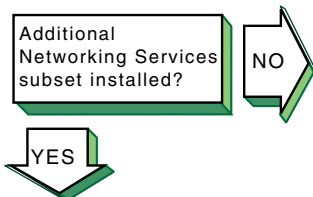
If the connection terminates abnormally, complete the following steps:

1. Test the network to determine whether the problem is on the local host, remote host, or a host on the path between the two. See Section 8.3.
2. After you identify the host with the problem, do the following:
 - a. Confirm that the network device is properly configured. Verify that the broadcast address and address mask for the local host are correct. See Section 2.3 for information on configuring network devices.
 - b. Make sure the `hosts` database on the local host has the correct IP addresses.
 - c. Make sure the cabling from the local host to the network is intact and properly connected.



Problem still exists?
Report it to your service representative. See Chapter 10.

8.6 Solving DHCP Problems



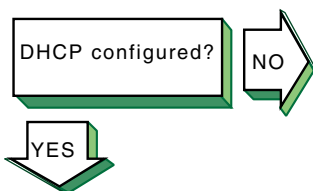
Verify that the Additional Networking Services subset is installed. Enter the following command:

```
# setld -i | grep OSFINET
```

If the subset is installed, the following message is displayed:

```
OSFINETnnn installed Additional Networking Services  
(Network-Server/Communications)
```

If the subset is not installed, install it by using the `setld` command. See the *Installation Guide* for more information on installing the subset.



Complete the following steps to verify that Dynamic Host Configuration Protocol (DHCP) has been configured on both server and client:

1. Use the `rcmgr` utility to display the value of the `JOIND` entry in the `/etc/rc.config.common` file on the DHCP server:

```
# rcmgr get JOIND
```

If nothing is returned, run the SysMan Menu utility to configure your DHCP server. See Section 5.3.7 for more information.

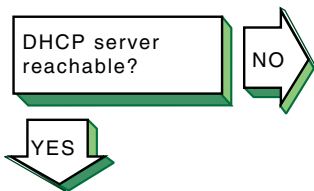
2. Use the `rcmgr` utility to display the value of the `IFCONFIG_n` entry in the `/etc/rc.config` file on the DHCP client. For example:

```
# rcmgr get IFCONFIG_0
```

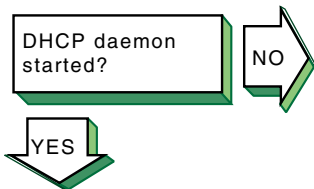
A value similar to the following is displayed:

```
DYNAMIC netmask n.n.n.n
```

If a similar value is not returned, run the SysMan Menu utility to configure your DHCP client. See Section 2.3 for more information.



Verify that the DHCP server is running and reachable, using the `ping` command.



Verify that the DHCP daemon (`joind`) is running on the server. Enter the following command:

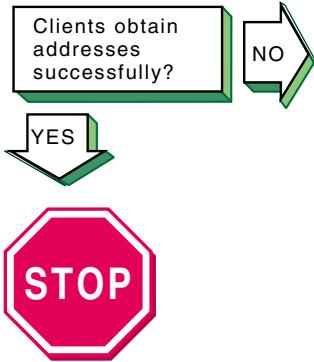
```
# ps -e | grep joind
```

Alternatively, you can use the SysMan Menu utility to view the status of the DHCP daemon. You can skip directly to the status dialog box by entering the following command:

```
# /usr/sbin/sysman dmnstatus
```

If the DHCP daemon is not running, start it by entering the following command:

```
# /usr/sbin/joind
```



Problem still exists?
Report it to your service
representative. See
Chapter 10.

If a DHCP client has problems obtaining DHCP information from the server, do the following:

1. Verify the Media Access Control (MAC) address you entered for the client. Users of Microsoft clients specifically must see Section 5.3.5, which explains how these clients modify their MAC addresses before sending them to the DHCP server.
2. Run the `joind` daemon with the debugging flag by doing the following:
 - a. Stop the `joind` daemon with the `kill -HUP` command.

Caution

Never use the `kill -9` command to stop the DHCP server daemon; it can corrupt your database files.

- b. Restart the `joind` daemon with the debug flag as follows:

```
# /usr/sbin/joind -d4
```

If you are running `joind` from the `/etc/inetd.conf` file, do the following:

- i. Edit the `/etc/inetd.conf` file and add the `-d4` flag.
- ii. Stop the `joind` daemon with the `kill -HUP` command.
- iii. Stop the `inetd` daemon with the `inetd -h` command. This forces the `inetd` daemon to reread the `/etc/inetd.conf` file.

Alternatively, you can run the SysMan Menu utility to configure your DHCP server with the debug option. See Section 5.3.7 for more information.

3. Review the `/var/join/log` file for information about the cause of any DHCP client problems.

The following example shows a `/var/join/log` file message that indicates a DHCP discover message arrived at the server system, but the IP subnetwork address range is not defined:

```
DHCPDISCOVER from HW address 08:00:2b:96:79:b6 :
network not administered by server
```

This problem can also occur if an address range is defined, but the `/etc/join/netmasks` file is missing the subnetwork mask definition for this IP network. In this case, edit the `netmasks`

file, add an entry for the subnetwork, and restart the DHCP server, `/usr/sbin/joind`.

8.7 Solving SLIP Problems

SLIP supported in kernel?

NO

YES

Verify that the correct number of Serial Line Internet Protocol (SLIP) pseudodevices are supported in the kernel by using the `netstat -in` command. If SLIP is supported, information similar to the following is displayed for each interface:

```
s10* 296 <Link> 0 0 0 0 0
```

The `s1` prefix indicates that SLIP is supported on the system. In this example there is one SLIP interface.

If you need additional SLIP interfaces, specify them by adding the `nslip=x` attribute under the `net:` subsystem in the `/etc/sysconfigtab` file. See *System Administration* for information on adding more SLIP interfaces.

On systems with 24 MB of memory, SLIP is not configured into the kernel. To add SLIP into the kernel, edit the system configuration file (`/usr/sys/confhostname`) and add the following entry:

```
options SL
```

See *System Administration* for more information.

Network hardware configured?

NO

YES

Configure the network hardware as follows:

- Verify that you are using the correct hardware. See Section 6.1.2.1 for more information.
- Make sure the modem is configured as follows:
 - Use 8-bit characters with no parity.
 - Software flow control (XON/XOFF) is disabled.
 - For dial-in systems, follow the guidelines in Section 6.1.3.1.
 - For dial-out systems, follow the guidelines in Section 6.1.3.2.

Dial in successful? (dial-in systems)

NO

YES

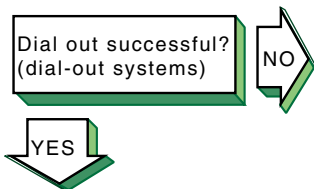
If a remote system cannot dial in to your system successfully, complete the following steps:

1. Check the `/usr/spool/locks` directory for `LCK..tty n` lock files. If any exist for the terminal device you are using for SLIP, remove them.

When you establish a connection over a terminal device, the system generates a lock file to prevent the connection from being disrupted by another application. If the connection

terminates abnormally, the lock file might persist, preventing you from establishing new connections.

2. Edit the `/etc/slhosts` file and include the debug option in the login entry for the host that cannot log in. See `slhosts(4)` for more information.
3. Instruct the remote user to dial in again.
4. Look in the `/var/adm/syslog.dated/current/daemon.log` file for information on SLIP problems on the dial-in system. See Section 9.8 for more information.

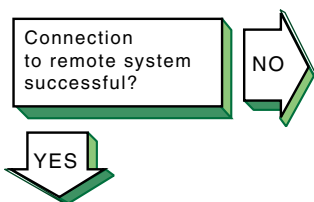


If you cannot dial out to the remote system, complete the following steps:

1. Check the `/usr/spool/locks` directory for `LCK..tty n` lock files. If any exist for the terminal device you are using for SLIP, remove them.

When you establish a connection over a terminal device, the system generates a lock file to prevent the connection from being disrupted by another application. If the connection terminates abnormally, the lock file might persist, preventing you from establishing new connections.

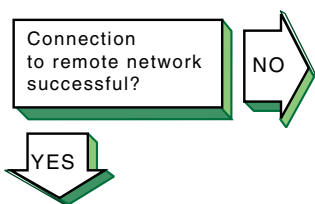
2. Verify that the modem is working correctly.
Edit the `/etc/acucap` file and include the `db` option in your modem's entry. This option displays useful information for debugging a new entry. See `acucap(4)` for more information.
3. Verify SLIP setup. Do the following:
 - a. Edit the `startslip` dial-out script file and specify the debug subcommand and a debug log file.
 - b. Try to dial out again.
 - c. Look in the debug log file for information about SLIP dial-out problems.



If you cannot communicate with the remote host and none of the debug messages shows an error, complete the following steps:

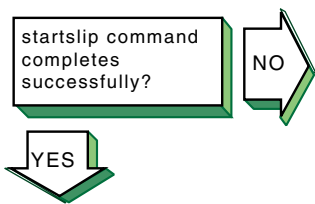
1. Verify that the IP addresses and netmasks are correct on both ends of the connection.
2. Examine the following SLIP configuration parameters at each end of the connection:
 - Internet Control Message Protocol (ICMP) traffic suppression — If enabled at either end of the connection, the `ping` command will fail.

- TCP header compression — If enabled at one end, TCP header compression must be enabled or autoenabled on the other end.



If you can communicate with the remote host but not the network connected to the remote host, complete the following steps:

1. If your local system is using the remote system as a gateway system, issue the `netstat -rn` command on the local system to verify that the remote SLIP address is the default gateway.
2. On the gateway system (remote system), issue the `iprsetup -d` command to see if the `ipforwarding` and `ipgateway` variables are on. If the variables are off, use the `iprsetup -s` command to turn them on.
3. On the gateway system, verify that the `gated` daemon is running. See `gated(8)` for more information.



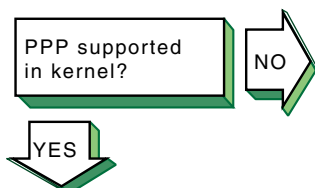
If the `startslip` command does not complete successfully, complete the following steps:

1. Build your kernel with the `PACKETFILTER` option.
2. Use the `tcpdump` command to examine packets sent and received through the SLIP interface. See `tcpdump(8)` for more information.



Problem still exists?
Report it to your service representative. See Chapter 10.

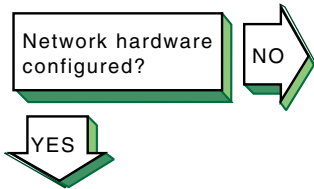
8.8 Solving PPP Problems



Verify that the Point-to-Point Protocol (PPP) is supported in the kernel by using the `sysconfig -s | fgrep ppp` command. If PPP is supported, information similar to the following is displayed:

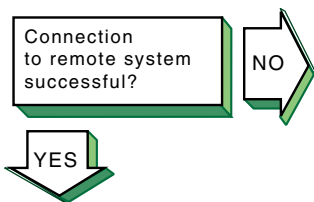
```
ppp: loaded and configured
```

If PPP is not supported, add options `PPP` into the `/sys/conf/MACHINE` system configuration file and rebuild the kernel.



Configure the network hardware as follows:

- Direct connections to remote host — Use a null modem or modem eliminator cable to connect your system to the remote host.
- Phone line connection to remote host — Use a cable to connect your system to a modem and another cable to connect your modem to a phone line. The modem you use must be compatible with the modem at the remote host. Make sure the modem is configured as follows:
 - Use 8-bit characters with no parity.
 - All flow control is disabled.



If you are logging messages to the console and the link comes up successfully, the following messages are displayed on the console:

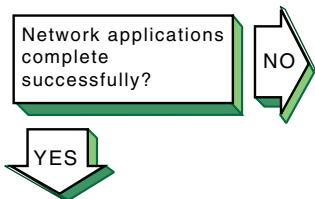
```
Local IP address: xx.xx.xx.xx  
Remote IP address: yy.yy.yy.yy
```

If the link does not come up, look at the following:

- Check the `/usr/spool/locks` directory for `LCK..ttypn` lock files. If any exist for the terminal device you are using for PPP, remove them.

When you establish a connection over a terminal device, the system generates a lock file to prevent the connection from being disrupted by another application. If the connection terminates abnormally, the lock file might persist, preventing you from establishing new connections.

- Verify that the serial connection is set up successfully. Use the `chat -v` command to log the characters the `chat` program sends and receives.
- Verify that the `pppd` daemon starts on the remote system. Use the `chat -v` command to log the characters the `chat` program sends and receives.
- Examine the PPP negotiation between the two peers. Use the `pppd` command with the `debug` option to log the contents of all control packets sent and received.



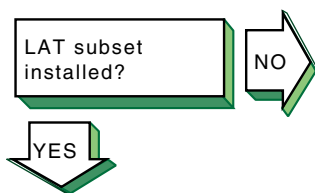
If network applications do not work successfully, this might indicate a problem with assigning IP addresses or routing. Do the following:

1. Use the `netstat -i`, `netstat -r`, `ping`, and `traceroute` commands to diagnose the problem.
2. If you can communicate with the peer machine but not with machines beyond that in the network, there is a routing problem. For instances where the local machine is connected to the Internet through the peer, do the following:
 - a. Assign the local machine an IP address on the same subnet as the remote machine.
 - b. Run the local `pppd` daemon with the `defaultroute` option.
 - c. Run the remote `pppd` daemon with the `proxyarp` option.
 - d. On the peer system (remote system), issue the `iprsetup -d` command to determine if the `ipforwarding` and `ipgateway` variables are on. If these variables are off, use the `iprsetup -s` command to turn them on.



Problem still exists?
Report it to your service representative. See Chapter 10.

8.9 Solving LAT Problems



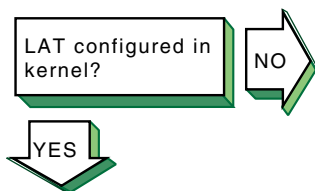
Verify that the Local Area Transport subset is installed. Enter the following command:

```
# setld -i | grep OSFLAT
```

If the subset is installed, the following message is displayed:

```
OSFLATnnn installed Local Area Transport (LAT)
  (General Applications)
```

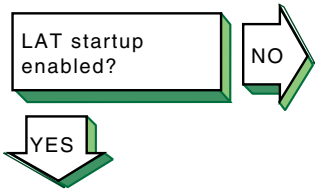
If the subset is not installed, install it by using the `setld` command. See the *Installation Guide* for information on installing the subset.



Verify that Local Area Transport is configured in the kernel. Enter the following command:

```
# sysconfig -q lat
```

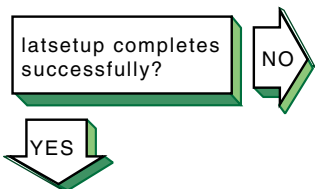
If no information is displayed, LAT is not configured in the kernel. Reconfigure the kernel with the LAT option. See *System Administration* for information on reconfiguring the kernel.



Use the `rcmgr` utility to display the value of the `LAT_SETUP` entry in the `/etc/rc.config` file:

```
# rcmgr get LAT_SETUP
```

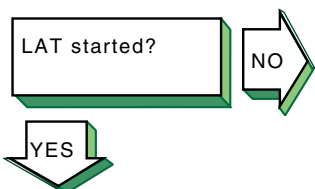
If 0 is returned, run the `latsetup` utility. See Section 7.3.1 for more information.



If the `latsetup` utility fails while creating new LAT ttys, verify that the `/usr/sbin` directory is included in the search path. Enter the following command:

```
# echo $PATH
```

If it is not, include it in your `PATH` environment variable. Then, create new LAT ttys using the `latsetup` command.



Verify that LAT has been started. Enter the following command:

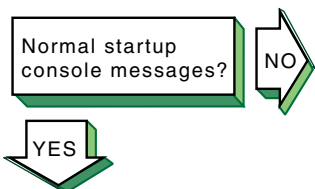
```
# latcp -d
```

If LAT is running, the following line is displayed:

```
LAT Protocol is active
```

If LAT was not started, start it. Enter the following command:

```
# latcp -s
```



If LAT starts and messages are continually displayed on the system console, look for the following messages and perform the required steps:

Message 1

```
getty: cannot open "/dev/lat/xx".  
errno: 2
```

This means a LAT terminal device file (tty) does not exist and the `/etc/inittab` file contains an entry for this file. The `latsetup` utility will also report that no LAT entries are available. Do the following:

1. Edit the `/etc/inittab` file and remove the LAT `getty` entries.
2. If LAT terminal devices are required, create the LAT terminal device files and corresponding entries in the `/etc/inittab` file by using the `latsetup` command. See `latsetup(8)` for information.

Message 2

```
getty: cannot open "/dev/lat/xx".  
errno: 19
```

This means the kernel was not configured with the LAT option and the `/etc/inittab` file contains at least one LAT `getty` entry. Do either of the following:

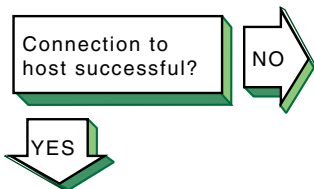
- Configure LAT into the kernel. See *System Administration* for information on configuring LAT into the kernel.
- Remove the LAT `getty` entries from the `/etc/inittab` file, either manually or by using the `latsetup` command.

Message 3

INIT: Command is respawning too rapidly.

The following meanings are possible:

- You are using an optional service name, such as `lattelnet`, and it is incorrectly defined. Do the following:
 1. Verify that the optional service name defined by the `latcp -A` command is correct by using the `latcp -d` command.
 2. Edit the `/etc/inittab` file and verify that a LAT entry has the optional service name specified correctly.
- An attempt was made to use a nonexistent LAT terminal device (`tty`). Do the following:
 1. Edit the `/etc/inittab` file and remove the entry with the nonexistent terminal device name.
 2. If LAT terminal devices are required, create the LAT terminal device files and corresponding entries in the `/etc/inittab` file by using the `latsetup` command. See `latsetup(8)` for more information.



If the user cannot connect to or display a service from a terminal server via LAT, complete the following steps on the system:

1. Verify that the service name is correct by using the `latcp -d` command. If the service name is incorrect, delete the service with the incorrect name. Enter the following command:

```
# latcp -D -aservice_name
```

Then, add a service with the correct name. Enter the following command:

```
# latcp -A -aservice_name
```

See `latcp(8)` for more information.
2. Display the group codes for the service to which the user is attempting to connect, using the `latcp -d` command. Check if any group code matches a group displayed by using

the `show port` command at the terminal server. If no group code matches, do either of the following:

- Add at least one group displayed by the port to the service. Enter the following command:

```
# latcp -glist -aservice_name
```

- Change the port characteristics at the terminal server by adding a group that matches the service.

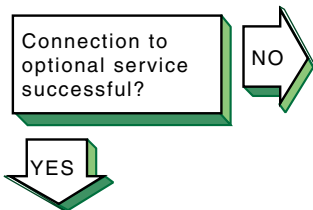
See `latcp(8)` for more information.

3. Check if LAT is started on the system. If it is not, start it. Enter the following command:

```
# latcp -s
```

4. If the problem persists, restart LAT. Enter the following command:

```
# latcp -s
```



If problems occur when using an optional service, complete the following steps:

1. Verify that the service was added as an optional service. Enter the following command:

```
# latcp -d
```

Look for the following line:

```
Service name: name (Optional)
```

If `Optional` is not displayed, the optional service was not defined with the `-o` option. Delete the service. Enter the following command:

```
# latcp -D -aservice_name
```

Then, add the service with the correct name and the `-o` option. Enter the following command:

```
# latcp -A -aservice_name -o
```

See `latcp(8)` for more information.

2. Verify that the optional service name matches the name defined in the `/etc/inittab` file. If it does not, do either of the following:

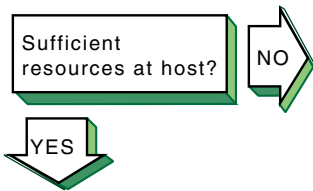
- Edit the `/etc/inittab` file and specify the optional service name.
- Delete the service. Enter the following command:

```
# latcp -D -aservice_name
```

Then, add the service with the correct name and the `-o` option. Enter the following command:

```
# latcp -A -aservice_name -o
```

See `latcp(8)` for more information.



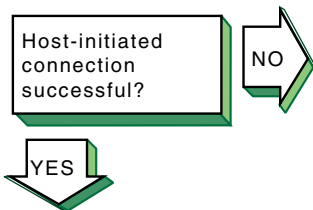
If the user cannot connect to a host using LAT, the following messages are displayed:

```
Connection
to node-name not established.
Service in use.
```

The `/etc/inittab` file does not contain a sufficient number of `getty` entries. Create more LAT terminal devices (`ttys`) and add their corresponding entries into the `/etc/inittab` file by using the `latsetup` command. Then, restart LAT to advertise the available services. Enter the following command:

```
# latcp -s
```

See Section 7.3.1 for information.



If a host-initiated connection fails, verify that the port, host, and service names are specified correctly. Enter the following command:

```
# latcp -d -P -L
```

If these names are not specified correctly, delete the application ports with the incorrect names. Enter the following command:

```
# latcp -D -pport_name
```

Then, add the application ports, using correct spelling. To create the application port by specifying the remote port to which the LAT terminal device is to be mapped, use the following command:

```
# latcp -A -plocal_port -Hnode -Rrem_port
```

Or, to create the application port by specifying the remote service name to which the LAT terminal device is to be mapped, use the following command:

```
# latcp -A -plocal_port -Hnode -Vsvc_name
```

See `latcp(8)` for information.

Note

When you delete an application port for a LAT printer, any print operations that are currently

executing continue until the printer buffer is empty.
The print job might not be complete.

Printing on LAT port
successful?



If you print a file to a printer attached to a LAT application port, the printer is online, and no printing occurs, look at the status of the print queue. Enter the following command:

```
# lpc status
```

The following line might be displayed:

```
waiting for printer to become ready (offline ?)
```

If this line is displayed, verify that LAT has been started. Enter the following command:

```
# latcp -d
```

If LAT has not been started, start it. Enter the following command:

```
# latcp -s
```



Connection using
LAT/Telnet gateway
successful?



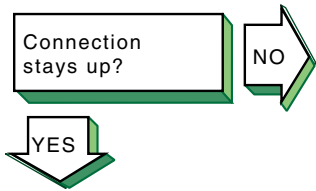
If problems are encountered with the LAT/Telnet gateway, look in the `/var/adm/syslog.dated/current/daemon.log` file for error messages. Use the error messages to diagnose the problem. See Section 9.8 for more information on viewing the `daemon.log` file.

The `lattelnet` utility uses the syslog message priority of `LOG_INFO`. For example, if you edit a LAT terminal entry in the `/etc/inittab` file, reassign it to `lattelnet` while a `getty` process is still active for the terminal, and a user tries to connect to LAT/Telnet, the connection will fail. The following error message is posted in the `daemon.log` file:

```
No such file or directory
```

Terminate the `getty` process for the terminal port.





Problem still exists?
Report it to your service
representative. See
Chapter 10.

If the LAT connection terminates abnormally, complete the following steps:

1. Examine the LAT terminal device (ttys) files for duplicate minor numbers. Enter the following command:

```
# ls -l /dev/lat/*
```

If any exist, remove the duplicate device files, leaving the original file.

2. Look in the `/etc/inittab` file for duplicate LAT entries. Remove the duplicate entries, leaving the original entry.

Using the Problem Solving Tools

To help you resolve problems with network connections and network hardware, the operating system provides problem solving tools you can use to complete the following tasks:

- Detect network interface failures (Section 9.1)
- Test access to network hosts on the Internet network (Section 9.2)
- Display network statistics (Section 9.3)
- Display and modify the Internet-to-Ethernet translation tables (Section 9.4)
- Display a datagram's route to a network host (Section 9.5)
- Display headers of packets on the network (Section 9.6)
- Display the error log file (Section 9.7)
- Display the `syslogd` daemon message files (Section 9.8)

The following sections contain information about using the tools associated with these tasks. For information about additional tools you can use to diagnose network services, see *Network Administration: Services*.

9.1 Detecting Network Interface Failures

You can use the Network Interface Failure Finder (NIFF) daemon, `niffd`, to detect and report possible failures in network interfaces or their connections.

When you enable monitoring for a particular network interface, the system begins tracking changes in the interface's packet counters. As long as the counters continue to increase, the system assumes that the network interface is functioning. If the counters do not increment within a given period of time, the `niffd` daemon verifies connectivity by generating its own traffic over the interface. If the daemon itself cannot get the counters to increment, signifying that the interface is functioning, it reports the problem to the Event Manager (EVM) subsystem.

You can review the associated log entries with the Event Viewer, or monitor connectivity problems in real time by using other EVM utilities.

This section describes how to manually configure NIFF to monitor individual interfaces; it does not describe how to provide failover for these interfaces.

Although NIFF provides the mechanism that the Redundant Array of Independent Network Adapters (NetRAIN) uses to determine when interfaces have failed, NIFF itself does not provide failover. To configure a NetRAIN set for automatic failover between network interfaces, see Section 2.1.1.2.

9.1.1 Configuring and Deconfiguring NIFF

Use the `niffconfig` command to enable monitoring for an interface, as follows:

```
# niffconfig -a interface-id
```

Replace *interface-id* with the device name for the network interface you want to monitor, for example, `tu0`. If necessary, you can specify more interfaces separated by spaces.

In addition, if you want the `niffd` daemon to continue monitoring an interface if you reboot your system, enter the following commands to enable the daemon in the `rc.config` file:

```
# rcmgr set NIFFD "YES"  
# rcmgr set NIFFC_FLAGS "-a interface-id"
```

You can display a list of the interfaces that the `niffd` daemon is currently monitoring by entering the `niffconfig` command with no options:

```
# niffconfig  
Interface:  tu0, status: UP
```

If necessary, you can later disable monitoring for an interface by entering the following command:

```
# niffconfig -r interface-id
```

Then, if you have configured the system to continue monitoring when you reboot your system, use the `rcmgr` command to update the `NIFFC_FLAGS` parameter. Or, to disable monitoring altogether, enter the following commands:

```
# rcmgr delete NIFFD "YES"  
# rcmgr delete NIFFC_FLAGS
```

See `niffconfig(8)` and `niffd(8)` for more information about configuring NIFF.

9.1.2 Viewing NIFF Events

Once NIFF is enabled for one or more interfaces, you can use the Event Viewer to view events related to those interfaces by doing the following:

1. From the SysMan Menu, select Monitoring and Tuning→View events to display the Event Viewer.

Alternatively, enter the following command on a command line:

```
# /usr/bin/sysman event_viewer
```

By default, the Event Viewer lists all events in the logs generated by the `syslogd` daemon. There could be hundreds or thousands of events; therefore, you will need to suppress everything but the events that NIFF generates.

2. Select Filter... to create a filter for NIFF events. The Filter dialog box is displayed.
3. Select the Event Name check box and the associated `equal to` check box.
4. Enter the `sys.unix.hw.net.niff.*` string into the Event Name text field. This string specifically identifies events that NIFF generates.
5. Optionally, if you want to suppress NIFF informational messages and alerts and view only interface failures, filter the events by priority, as follows:
 - a. Select the Priority check box, the associated `equal to` check box, and the Range check box.
 - b. Specify a range of 600–700 in the Range text field.
Failures are reported with a priority of 600. Informational messages and alerts are reported with a priority of 200; therefore, they will be hidden.
6. Select OK to save and apply the specified filters.

If NIFF has generated any events, the Event Viewer displays them.

The Event Viewer displays events that have already been reported. You must select the Refresh option to view additional events as they are reported. Or, you can use the following procedure to send connectivity alerts directly to a terminal on your local console when the `niffd` daemon reports them to EVM:

1. Open a new terminal (for example, `dtterm` or `xterm`) on your local console.
2. Execute one of the following commands in the new terminal.

To display all NIFF events (informational messages, alerts, and failures):

```
# evmwatch | evmshow -f "[name sys.unix.hw.net.niff.*]"
-t "@timestamp [@priority] @@"
```

To display only failures:

```
# evmwatch -f "[priority >= 600]" | evmshow -f "[name
sys.unix.hw.net.niff.*]" -t "@timestamp [@priority] @@"
```

The terminal will display events as the `niffd` daemon reports them. The terminal will appear dormant until an event of the appropriate priority is reported.

You cannot execute additional commands in this terminal until you abort the process by typing `Ctrl/c`.

Note that if the system running the `niffd` daemon contains only one network interface, you cannot use this method to monitor that network interface's connectivity via a remote host. You must be on the local console.

See *System Administration* for more information about EVM.

9.2 Testing Access to Internet Network Hosts

Use the `ping` command to test your system's ability to reach a host on the Internet network. The `ping` command has the following syntax:

```
/usr/sbin/ping [options... ] hostname
```

Table 9–1 describes some of the `ping` command options.

Table 9–1: Options to the ping Command

Option	Function
<code>-c count</code>	Specifies the number of ECHO RESPONSE packets to send and receive.
<code>-I interface</code>	Specifies the interface over which to send packets.
<code>-R</code>	Includes the RECORD_ROUTE option in the packet and displays the route buffer on returned packets.
<code>-r</code>	Executes the <code>ping</code> command for a host directly connected to the local host. With this option, the <code>ping</code> command bypasses normal routing tables and sends the request directly to a host on an attached network. If the host is not on a directly attached network, the local host receives an error message.
<code>-V</code>	Specifies the IP version number (4 or 6) of the address returned by the resolver when a host name has both IPv4 and IPv6 addresses. By default, the <code>ping</code> command tries to resolve host names as an IPv6 address then IPv4 address.

The `ping` command sends an Internet Control Message Protocol (ICMP) echo request to the host specified. When the request is successful, the remote host sends the data back to the local host. If the remote host does not respond to the request, the `ping` command does not display any results.

To terminate the `ping` command output, press `Ctrl/C`. When terminated, the `ping` command displays statistics on packets sent, packets received, the percentage of packets lost, and the minimum, average, and maximum round-trip packet times.

You can use the output from the `ping` command to help determine the cause of direct and indirect routing problems such as an unreachable host, a timed-out connection, or an unreachable network.

When using the `ping` command for fault isolation, first test the local host to verify that it is running. If the local host returns the data correctly, use the `ping` command to test remote hosts farther and farther away from the local host.

If you do not specify command options, the `ping` command displays the results of each ICMP request in sequence, the number of bytes received from the remote host, and the round-trip time on a per-request basis.

The following example shows the output from a `ping` command to a host named `host1`:

```
% ping host1
PING host1.corp.com (16.20.32.2): 56 data bytes
64 bytes from 16.20.32.2: icmp_seq=0 ttl=255 time=11 ms
64 bytes from 16.20.32.2: icmp_seq=1 ttl=255 time=3 ms
64 bytes from 16.20.32.2: icmp_seq=2 ttl=255 time=7 ms
64 bytes from 16.20.32.2: icmp_seq=3 ttl=255 time=3 ms
64 bytes from 16.20.32.2: icmp_seq=4 ttl=255 time=7 ms
64 bytes from 16.20.32.2: icmp_seq=5 ttl=255 time=3 ms
Ctrl/C
---host1.corp.com PING Statistics---
6 packets transmitted, 6 packets received, 0% packet loss
roundtrip (ms) min/avg/max = 3/5/11 ms
```

The `ping` command accepts an IPv4 address, IPv6 address, or node name on the command line. The following example specifies an IPv6 address:

```
# ping -c 2 5F00:2100:108C:4000:8C40:800:2B2D:2B2
PING (5F00:2100:108C:4000:8C40:800:2B2D:2B2): 56 data bytes
64 bytes from 5F00:2100:108C:4000:8C40:800:2B2D:2B2: icmp6_seq=0
    hlim=58 time=17 ms
64 bytes from 5F00:2100:108C:4000:8C40:800:2B2D:2B2: icmp6_seq=1
    hlim=58 time=17 ms
---5F00:2100:108C:4000:8C40:800:2B2D:2B2 PING Statistics---
2 packets transmitted, 2 packets received, 0% packet loss
round-trip (ms) min/avg/max = 17/17/17 ms
```

The command sends appropriate ECHO_REQUEST packets based on the address family being used. In some cases, a single node name might resolve to both an IPv4 and IPv6 address. Use the -v4 or -v6 option specify which address to use.

You can also use the -I flag to force the use of a specific interface. For example:

```
# ping -I ln0 FE80::800:2B2D:2B2
```

See ping(8) for more information on this command and its options.

9.3 Displaying Network Statistics

Use the netstat command to display network statistics for sockets, interfaces, and routing tables. You can select several forms of display; each allows you to specify the type of information you want to emphasize.

Table 9-2 shows the netstat command options.

Table 9-2: Options to the netstat Command

Option	Function
-A	Displays the address of any associated protocol control blocks.
-a	Includes information for all sockets.
-f <i>address_family</i>	Includes statistics or address control block reports for the specified address family, for example, inet (IPv4) or inet6 (IPv6).
-I <i>interface</i>	Displays information about the specified interface.
-i	Provides status information for autoconfigured interfaces.
-m	Displays information about memory management usage.
-n	Lists network addresses in number form rather than symbolic form.
-r	Lists routing tables.
-s	Provides statistics per protocol.
-t	Displays the time until the interface watchdog routine starts (for use with the -i option).

The -I option provides statistics for a specific interface. See Appendix A for an example of using the -I option to monitor Ethernet, Fiber Distributed Data Interface (FDDI), and token ring interfaces, and a description of the counters, status, and characteristics.

The `-i` option provides statistics on each configured network interface. Outgoing packet errors (`Oerrs`) indicate a potential problem with the local host. Incoming errors (`Ierrs`) indicate a potential problem with the network connected to the interface.

The `-f inet` and `-f inet6` options limit the data displayed to either IPv4 or IPv6, respectively. For example, the `netstat -f inet6 -rn` command displays only IPv6 routing table entries, as opposed to the default, which displays both IPv4 and IPv6 entries.

The `netstat -s` command displays statistics for all protocols, including IPv6 and ICMPv6.

The following example shows normal output from the `netstat` command with the `-i` option:

```
% netstat -i
Name Mtu Network Address Ipkts Ierrs Opkts Oerrs Coll
ln0 1500 <Link> 8324125 0 8347463 0 237706
ln0 1500 16.31.16 host1 8324125 0 8347463 0 237706
fza0* 4352 <Link> 0 0 0 0 0
sl0* 296 <Link> 0 0 0 0 0
sl1* 296 <Link> 0 0 0 0 0
tra0 4092 <Link> 34 0 20 0 0
tra0 4092 16.40.15 host21 34 0 20 0 0
lo0 1536 <Link> 909234 0 909234 0 0
lo0 1536 loop localhost 909234 0 909234 0 0
```

There are no `Ierrs` or `Oerrs`, which indicates that there are currently no network connectivity problems.

See `netstat(1)` for more information about this command and its options.

9.4 Displaying and Modifying the Internet (IPv4) to MAC Address Translation Tables

You can display and modify the Internet to Media Access Control (MAC) address translation tables used by the Address Resolution Protocol (ARP) to help diagnose direct IPv4 routing problems resulting from the following circumstances:

- A source host has incorrect Ethernet address information for a destination host.
- Two hosts have the same IPv4 address.

Although you can work around this problem by modifying the translation tables, it is best to change one host's IPv4 address to permanently resolve the conflict.

Use the `arp -a` command to display the entries in the Internet-to-MAC address translation tables. To modify the tables, log in as root and use the `arp` command as follows:

```
/usr/sbin/arp [options] hostname
```

The following example shows the Ethernet address for an IPv4 host named `host1`. The system response tells you that the Ethernet address for `host1` is `aa-00-04-00-8f-11`.

```
# /usr/sbin/arp host1
host1 (16.20.32.2) at aa:0:4:0:8f:11 permanent
```

The following example shows how to temporarily add `host9` to the system translation tables:

```
# /usr/sbin/arp -s host9 0:dd:0:a:85:0 temp
```

The following example shows how to remove `host8` from the system translation tables:

```
# /usr/sbin/arp -d host8
```

See `arp(8)` for more information on this command.

9.5 Displaying a Datagram's Route to a Network Host

You can display a datagram's route to a network host to manually test, measure, and manage the network.

To display a datagram's route, use the `traceroute` command with the following syntax:

```
traceroute [options...] hostname [packetsize]
```

Table 9-3 describes some of the `traceroute` command options.

Table 9-3: Options to the `traceroute` Command

Option	Function
<code>-m max_ttl</code>	Sets the maximum time-to-live (<code>ttl</code>) used in outgoing probe packets. The <code>ttl</code> parameter specifies the maximum number of hops a packet can take to reach its destination. The default is 30 hops.
<code>-n</code>	Displays hop addresses numerically only, rather than both numerically and symbolically.
<code>-p port</code>	Sets the base User Datagram Protocol (UDP) port number to be used in outgoing probe packets. The default is 33434. The port information is used to select an unused port range if a port in the default range is already used.

Table 9–3: Options to the traceroute Command (cont.)

Option	Function
-r	Bypasses the normal routing tables and sends the probe packet directly to a host on an attached network. If the host is not on a directly attached network, the <code>traceroute</code> command returns an error.
-s <i>IP_address_number</i>	Uses the specified IP address number as the source address in outgoing probe packets. On hosts with more than one IP address, this option forces the <code>traceroute</code> command to use the specified source address rather than any others the host might have. If the IP address is not one of the receiving host's interface addresses, the command returns an error and does not send a probe packet.
-t <i>type-of-service value</i>	Sets the type-of-service in probe packets to the specified value. The default is zero. The value must be a decimal integer in the range 0–255. This option tells you if different types of service result in different paths. This option is available only in Berkeley UNIX (4.4BSD) environments. Not all types of service are legal or meaningful. Useful values for this option are 16 (low delay) and 8 (high delay). See RFC 791, <i>Internet Protocol</i> for more information on types of service.
-v	Displays verbose output, which includes received ICMP messages other than <code>time exceeded</code> and <code>port unreachable</code> .
-V <i>version</i>	Specifies the IP version number (4 or 6) of the address returned by the resolver when a host name has both IPv4 and IPv6 addresses. By default, the <code>traceroute</code> command tries to resolve host names as an IPv6 address then IPv4 address.
-w <i>wait_time</i>	Sets the time (in seconds) to wait for a response to a probe. The default is 3 seconds.
<i>packetsize</i>	Sets the packet size (in bytes) for the probe packet. The default size is 38 bytes.

The `traceroute` command sends UDP packets (known as probe packets) to an unused port on the remote host, and listens for ICMP replies from IP routers. It sends the probe packets with a small `tTL` parameter, which specifies the maximum number of hops a packet can take to reach its destination. The `traceroute` command starts by specifying a `tTL` of one hop and it increases the `tTL` by one for each probe packet it sends. It continues sending probe packets until a packet reaches the destination or until the `tTL` reaches the maximum number of hops.

In response to each probe packet, the `traceroute` command can receive one of the following ICMP messages:

- `time exceeded`

The IP router that received the probe packet cannot forward it any further due to the `tll` value. This message tells you which IP routers are processing the packets.

- `port unreachable`

The probe packet reached its intended destination, but could not access the intended port.

When the `traceroute` command sends three probe packets (datagrams) for each `tll` setting, it displays a line showing the following:

- `tll`
- IP address of the host or router that responded
- Round-trip time of each probe datagram/ICMP response

If multiple IP routers respond to the probe, the `traceroute` command displays the address of each IP router. If the `traceroute` command does not elicit a response in 3 seconds (the default wait time), an asterisk (*) is displayed for the probe.

The following example shows a successful `traceroute` command to `host2`:

```
% traceroute host2
traceroute to host2 (555.55.5.5), 30 hops max, 40 byte packets
 1 host3 (555.55.5.1) 2 ms 2 ms 2 ms
 2 host5 (555.55.5.2) 5 ms 6 ms 4 ms
 3 host7 (555.55.5.3) 7 ms 7 ms 6 ms
 4 host2 (555.55.5.5) 12 ms 8 ms 8 ms
```

The `traceroute` command with the `host` argument prints the route that packets take to both IPv4 and IPv6 hosts.

See `traceroute(8)` for more information about this command and its options.

9.6 Displaying Headers of Packets on the Network

You display packet headers on the network when you want to monitor the network traffic associated with a particular network service. This is usually done to determine whether requests are being received or acknowledged, or to determine the source of network requests, in the case of slow network performance.

Use the `tcpdump` command to display packet headers for a network interface. This command enables you to specify the interface on which to

listen, the direction of the packet transfer, and the type of protocol traffic to display. In addition, it enables you to identify the source of the packet. See `tcpdump(8)` for more information.

Note

In order to use the `tcpdump` command, the `packetfilter` option must be configured into the kernel and the system rebooted. See `packetfilter(7)` for more information.

9.7 Viewing the Error Log File

To diagnose kernel and hardware errors, you can look at the system events that occurred prior to the errors. Messages from system events, such as error messages relating to the software kernel and system hardware, and informational messages about system status, startup, and diagnostics, are recorded in the binary error log file, `/var/adm/binary.errlog`.

Because this log file is in binary format, the operating system offers special utilities, `DECEvent` and `Compaq Analyze`, that read the binary log file and run the data through a formatter to display the information. See `dia(8)` and `ca(8)` for more information about `DECEvent` and `Compaq Analyze`, respectively.

Note that these utilities are not available in the operating system by default; you must install the Web-Based Enterprise Services (WEBES) kit, a suite of diagnostic utilities, to obtain them. WEBES is available for installation from the Associated Product CD-ROMs or for download from the following URL:

<http://www.support.compaq.com/svctools/webes>

See the *System Administration* manual for information about using the Event Viewer to present errors as interpreted by `DECEvent` and `Compaq Analyze`. Also, see `uerf(8)` for an alternative to these utilities.

9.8 Viewing the syslogd Daemon Message Files

You can use the `syslogd` daemon to help diagnose session layer problems such as access control problems for the Internet Protocol Version 4 (IPv4) and Internet Protocol Version 6 (IPv6).

The `syslogd` daemon starts running when you boot the system and whenever it receives a hangup signal. By default, it records the system messages for these events in a set of files in the `/var/adm/syslog.dated` directory (as specified in the `/etc/syslog.conf` file). The system messages can indicate error conditions or warnings, depending on the priority codes they contain.

Although it is possible to review the contents of the system message files from the command line, it is best to use the Event Viewer that is part of the SysMan Menu utility, because it simplifies access to the files and makes it easier for you to find particular problems. To start the Event Viewer, invoke the SysMan Menu as described in Section 1.2.1, then select Monitoring and Tuning→View events. Alternatively, you can invoke the Event Viewer from a command line by entering the following command:

```
# /usr/bin/sysman event_viewer
```

Once the Event Viewer is displayed, you can use it to sort the log entries, filter the entries (for a certain event name, priority level, posting host, or date), and obtain more detailed information about individual entries.

For more information about event management and accessing the system log files, see `evm(5)`, `syslogd(8)`, the *System Administration* manual, and the online help.

Reporting Network Problems

If you are unable to solve a critical problem with the network or network service, do the following:

1. Read the release notes for the product to see if the problem is known. If it is, follow the solution offered to solve the problem.
2. Determine whether the product is still under warranty or whether your company purchased support services for the product. Your operations manager can supply you with the necessary information.
3. If either condition in step 2 was met, take one of the following actions:
 - a. Access the online service database, if you have purchased this service, and determine if the problem you are experiencing has already been reported. If it has not, log your problem.
 - b. Call your service representative and describe the problem.
4. If you are requested to supply any information pertaining to the problem, gather the necessary information and submit it.

You might be asked to submit some information that can help isolate problems to a particular area of the system and speed the resolution of the problem. It is a good idea to keep all basic information in a `system.information` file. Then you can easily include it with your problem report.

The following sections describe some of the information that you might be asked to submit.

10.1 Gathering General Information

Gather the following information about your system:

- The operating system version and revision number (from the `/etc/motd` file). Add this to the `system.information` file.
- A description of your system's activity before the error.
- A listing of the exact command line or lines executed and the output.

- A copy of the application source code, if running a user-created application. If possible, include a sample test program that demonstrates the problem.

10.2 Gathering Hardware Architecture Information

Gather the following information about the hardware architecture:

- A description of the model of the workstation or server (from the `/usr/sys/conf/HOSTNAME` file), including the type of graphics controller (if a workstation), the amount of memory, and third-party hardware

- A description of the X server

To determine which type you are running, enter the following command:

```
# ps ax | grep /usr/bin/X >> system.information
```

- A description of the disks used and the size of your swap partition

For example, if your system disk is unit 0, enter the following commands as root to add this information to the `system.information` file:

```
# disklabel -r /dev/rrz0a >> system.information
# echo df: >> /system.information
# df >> /system.information
# echo mount: >> /system.information
# mount >> /system.information
# echo xdpinfo: >> /system.information
# xdpinfo >> /system.information
```

- Any networking information

To add this to the `system.information` file, enter the following commands:

```
# echo netstat: >> /system.information
# netstat -i -n >> system.information
# netstat -r -n >> /system.information
# echo nslookup: >> /system.information
# nslookup localhost >> /system.information
```

- Any event logging information

To add this to the `system.information` file, enter the following commands:

```
# uerf -R -o full | head -200 >> /system.information
```

10.3 Gathering Software Architecture Information

Gather the following information about the software architecture:

- A description of the software subsets installed

To add this to the `system.information` file, enter the following commands:

```
# echo setld: >> /system.information
# setld -i >> /system.information
```

- The output of the `setld` log file

To add this to the `system.information` file, enter the following command:

```
# pr /usr/adm/smlogs/setld.log >> /system.information
```

- The automatic reboot file

To add this to the `system.information` file, enter the following commands:

```
# pr /etc/rc.config* >> /system.information
# pr /sbin/rc[023] >> /system.information
# pr /sbin/init.* >> /system.information
```

- A description of the layered products installed

A

Monitoring the Network Interfaces

The `netstat` command can help you monitor the Ethernet, Fiber Distributed Data Interface (FDDI), and token ring network interfaces. The following sections contain sample system output and a description of the information for each network interface.

A.1 Monitoring the Ethernet Interface

You can use the `netstat -I ln0 -s` command to obtain a listing of the Ethernet counters. The following is sample system output from this command:

```
ln0 Ethernet counters at Thu Nov 6 07:33:00 1992
    1289 seconds since last zeroed
16812469 bytes received
4657308 bytes sent
    42555 data blocks received
    28418 data blocks sent
860360 multicast bytes received
    7710 multicast blocks received
    546 multicast bytes sent
    13 multicast blocks sent
    0 blocks sent, initially deferred
    1864 blocks sent, single collision
    5542 blocks sent, multiple collisions
    6 send failures, reasons include:
        Excessive collisions
    0 collision detect check failure
    3 receive failures, reasons include:
        Block check error
        Framing Error
    0 unrecognized frame destination
    0 data overruns
    0 system buffer unavailable
    0 user buffer unavailable
```

The following section lists each field in the previous example alphabetically, and describes each field.

blocks sent, initially deferred

The number of times a frame transmission was deferred on its first transmission attempt. Used in measuring Ethernet contention with no collisions.

blocks sent, multiple collisions

The number of times a frame was successfully transmitted on the third or later attempt after normal collisions on previous attempts.

blocks sent, single collision

The number of times a frame was successfully transmitted on the second attempt after a normal collision on the first attempt.

bytes received

The number of bytes successfully received.

bytes sent

The number of bytes successfully transmitted.

collision detect check failure

The number of times a collision detection was not sensed after a transmission.

data blocks received

The number of frames successfully received.

data blocks sent

The number of frames successfully transmitted.

data overruns

The number of times a frame was discarded because no receive buffer was available.

multicast blocks received

The number of frames successfully received in multicast frames.

multicast blocks sent

The number of frames successfully transmitted in multicast frames.

multicast bytes received

The number of bytes successfully received in multicast frames.

multicast bytes sent

The number of bytes successfully transmitted in multicast frames.

receive failures, reasons include:

The number of times a receive error occurred. Each receive error is classified as one of the following:

- Block check error
- Framing error
- Frame too long

seconds since last zeroed

The number of seconds since the associated counter attributes were set to zero.

send failures, reasons include:

The number of times a transmit error occurred. Each transmit error is classified as one of the following:

- Excessive collisions
- Carriers check failed
- Short circuit
- Open circuit
- Frame too long
- Remote failure to defer

system buffer unavailable

The number of times a frame was discarded because no link buffer was available.

unrecognized frame destination

The number of times a frame was discarded because there was no data link port. The count includes frames received for the physical address only. It does not include frames received for the multicast or broadcast address.

user buffer unavailable

The number of times a frame was discarded because no user buffer was available.

A.2 Monitoring the FDDI Interface

You can use the `netstat -I interface -s` command to obtain a listing of the Fiber Distributed Data Interface (FDDI) counters, status, and characteristics for the FDDI interface. The following is sample system output from this command for the `fza0` interface. See `faa(7)`, `fta(7)`, `fza(7)`, and `mfa(7)` for adapter error messages.

```
fza0 FDDI counters at Wed Jun 12 14:02:44 1992
      89 seconds since last zeroed
6440875 ANSI MAC frame count
      0 ANSI MAC frame error count
      0 ANSI MAC frames lost count
37488 bytes received
39005 bytes sent
  447 data blocks received
  479 data blocks sent
30170 multicast bytes received
  321 multicast blocks received
29163 multicast bytes sent
  360 multicast blocks sent
      0 transmit underrun errors
      0 send failures
      0 FCS check failures
      0 frame status errors
      0 frame alignment errors
      0 frame length errors
      0 unrecognized frames
      0 unrecognized multicast frames
      0 receive data overruns
      0 system buffers unavailable
      0 user buffers unavailable
      0 ring reinitialization received
      0 ring reinitialization initiated
      0 ring beacon process initiated
      0 ring beacon process received
      0 duplicate tokens detected
      0 duplicate address test failures
      0 ring purger errors
      0 bridge strip errors
      0 traces initiated
      0 traces received
      0 LEM reject count
      0 LEM events count
      0 LCT reject count
```

0 TNE expired reject count
1 completed connection count
0 elasticity buffer errors

fza0 FDDI status

Station State: On
Last Station ID: Not Implemented
Station UID: 00-00-08-00-2B-A2
Link State: On ring running
Link UID: 08-00-2B-A2-B5-84
Negotiated TRT: 7.987 ms
Duplicate Address Test: Absent
Upstream Neighbor Address: 08-00-2B-18-B3-D7
Old Upstream Neighbor Address: 08-00-2B-1E-C0-3E
Upstream Neighbor Dup Addr Flag: Unknown
Downstream Neighbor Address: 08-00-2B-1E-C0-3E
Old Downstream Neighbor Address: 08-00-2B-1E-C0-3E
Ring Purger State: Purger off
Frame Strip Mode: Source Address Match
Ring Error Reason: No reason
Loopback Mode: False
Ring Latency: 0.000 ms
Ring Purge Address: Not Implemented
Physical Port State: In use
Physical Port UID: 08-00-2B-A2-B5-84
Neighbor Physical Port Type: Master
Physical Link Error Estimate: 15
Broken Reason: None
Reject Reason: No reason

fza0 FDDI characteristics

Station ID: 00-00-08-00-2B-A2
Station Type: SAS
SMT Version ID: 2
SMT Max Version ID: 2
SMT Min Version ID: 2
Link Address: 08-00-2B-A2-B5-84
Requested TRT: 8.000 ms
Valid Transmission Time: 2.621 ms
Restricted Token Timeout: 1000.000 ms
Ring Purger Enable: FALSE
Physical Port Type: Slave
PMD Type: ANSI multimode
LEM Threshold: 8

The Downstream Neighbor Address and Restricted Token Timeout are reported only for the DEFZA firmware revision 1.2 and higher.

The following sections list each field in the previous example alphabetically, and describe each field.

A.2.1 FDDI Counters

This section lists the FDDI counters alphabetically.

ANSI MAC frame count

The total number of frames (other than the token frame) seen by this link.

ANSI MAC frame error count

The total number of times the media access control (MAC) changed the E indicator in a frame from R to S.

ANSI MAC frames lost count

The total number of times a frame (other than the token frame) was improperly terminated.

bridge strip errors

The number of times a frame content independent strip operation was terminated by receipt of a token.

bytes received

The number of bytes successfully received.

bytes sent

The number of bytes successfully transmitted.

completed connection count

The number of times the physical (PHY) port entered the In Use state, having completed the initialization process.

data blocks received

The number of frames successfully received.

data blocks sent

The number of frames successfully transmitted.

duplicate address test failures

The number of times the duplicate address test failed.

duplicate tokens detected

The number of times the MAC detected a duplicate token, either via the duplicate token detection algorithm or by receiving a token while already holding one.

elasticity buffer errors

The number of times the Elasticity Buffer function in the PHY port had an overflow or underflow.

FCS check failures

The number of times a received frame failed the Frame Control Status (FCS) check.

frame alignment errors

The number of times a received frame had an alignment error.

frame length errors

The number of times a received frame had an invalid length, either too long or too short.

frame status errors

The number of times a received frame had the E indicator in error but the cyclic redundancy check (CRC) was correct.

LCT reject count

The number of times a connection on this physical port was rejected due to failure of the link confidence test (LCT) at either end of the physical connection.

LEM events count

The number of errors detected by the link error monitor (LEM) on the physical layer.

LEM reject count

The number of times an active connection on this physical port was disconnected due to rejection by the LEM at this end of the physical connection.

multicast blocks received

The number of frames successfully received in multicast frames.

multicast blocks sent

The number of frames successfully transmitted in multicast frames.

multicast bytes received

The number of bytes successfully received in multicast frames.

multicast bytes sent

The number of bytes successfully transmitted in multicast frames.

receive data overruns

The number of times a frame was discarded because no receive buffer was available.

ring beacon process initiated

The number of times the ring beacon process was initiated by this link.

ring beacon process received

The number of times the ring beacon process reinitialization was initiated by some other link.

ring purger errors

The number of times the ring purger received a token while still in the ring purge state.

ring reinitialization initiated

The number of times a ring reinitialization was initiated by this link.

ring reinitialization received

The number of times a ring reinitialization was initiated by some other link.

seconds since last zeroed

The time at which the link entity was created. This value indicates when the associated counter attributes were set to zero.

send failures

The number of times a transmit error (other than transmit underrun) occurred.

system buffers unavailable

The number of times a frame was discarded because no link buffer was available.

TNE expired reject count

The number of times an active connection on this physical port was disconnected due to rejection by expiration of the noise timer (TNE).

traces initiated

The number of times the PC-trace process was initiated by this link.

traces received

The number of times the PC-trace process was initiated by some other link.

transmit underrun errors

The number of times a transmit underrun error occurred. This indicates the transmit first-in/first-out (FIFO) buffer became empty during frame transmission.

unrecognized frames

The number of times a received, individually addressed logical link control (LLC) frame was discarded because there was no data link port.

unrecognized multicast frames

The number of times a received LLC frame addressed to a multicast address was discarded because there was no data link port.

user buffers unavailable

The number of times a frame was discarded because no user buffer was available.

A.2.2 FDDI Status

This section lists the FDDI status alphabetically.

Broken Reason

The reason that the physical port is in the Broken state (for non-SAS stations). This field can have one of the following values:

Broken	The physical port is broken.
None	The physical port is not in the Broken state.

Downstream Neighbor Address

The 48-bit hardware address of the station that is on the downstream side of the ring from this station.

Duplicate Address Test

The result of the duplicate address test performed by the FDDI MAC entity of the station. This field can have one of the following conditions:

Absent	The FDDI MAC entity determined that there is no duplicate of its own line address on the ring.
Present	The FDDI MAC entity determined that a duplicate of its own line address exists on the ring. No data can be transmitted or received on the line until this logical ring fault is resolved.
Unknown	The FDDI MAC entity is performing the duplicate address test to determine if any other stations on the ring have the same address as the line.

Frame Strip Mode

The frame strip mode used by the station. This field can have one of the following values:

Source Address Match	The station strips frames from the ring that contain its own address in the source address field.
Bridge Strip	The station maintains a count of frames sent since obtaining the token, sends a void frame when the transmission is complete (two void frames if it is serving as ring purger), and strips the returning frames from the ring until the count of frames sent is decremented to zero. Bridge stripping is used by bridges because they are sensitive to no-owner frames and frequently send frames that do not contain their own address in the source address field.
Unknown	The station is not operating on the ring.

Last Station ID

If implemented, this is the 48-bit address of the station that last performed a successful Parameter Management Frame (PMF) change,

add, or remove operation. If not implemented, the phrase “Not implemented” is displayed.

Link State

The operational state of the FDDI MAC entity of the station. This field can have one of the following values:

Broken	A hardware problem exists.
Off Fault Recovery	The FDDI MAC entity is recovering from a logical ring fault such as a failure of the duplicate address test, a local or remote stuck beaconing condition, or ring operational oscillation.
Off Maintenance	The FDDI MAC entity is performing loopback testing and online diagnostics.
Off Ready	The FDDI MAC entity is ready for operation but is not yet connected to the logical ring.
On Ring Initializing	The FDDI MAC entity is connecting to the logical ring.
On Ring Running	The FDDI MAC entity is connected to the logical ring and is fully operational.
Unknown	The FDDI MAC entity is not connected to the ring.

Link UID

The 48-bit address of the physical port for the data link.

Loopback Mode

The operational state of loopback mode for the link entity. This field can have one of the following values:

False	Loopback mode is off. The link entity is not set up to receive frames that it transmits in order to perform loopback testing on the ring or of the physical port.
True	Loopback mode is on. The link entity is set up to receive frames that it transmits in order to perform loopback testing on the ring or of the physical port.

Negotiated TRT

The negotiated target token rotation time (TTRT) value is referred to as T_Neg in the ANSI FDDI specifications. It is negotiated during the claim token process.

Neighbor Physical Port Type

The type of the neighbor physical port. This field can have one of the following values:

- A The physical port on a dual attachment wiring concentrator (DAC) or dual attachment station (DAS) that connects to the incoming primary ring and the outgoing secondary ring of the FDDI dual ring.
- B The physical port on a dual attachment wiring concentrator (DAC) or dual attachment station (DAS) that connects to the outgoing primary ring and the incoming secondary ring of the FDDI dual ring.
- Master One of the physical ports on a wiring concentrator that connects to a single attachment station (SAS) such as a DECbridge 500 device.
- Slave The physical port on a single attachment station (SAS) that connects to a wiring concentrator or another SAS.
- Unknown Physical port type is undefined.

Old Downstream Neighbor Address

The 48-bit hardware address of the station that was previously on the downstream side of the ring from this station.

Old Upstream Neighbor Address

The 48-bit hardware address of the station that was previously on the upstream side of the ring from this station.

Physical Link Error Estimate

The current link error rate as estimated by the link error monitor (LEM). For a value of n , the actual rate is 1×10^{-n} .

Physical Port State

The operational state of the physical port. This field can have one of the following values:

- Broken The physical port failed its diagnostic tests and is nonoperational.
- Failed Same as Waiting, except that the physical port failed at least once; by failing the link confidence test (LCT) during initialization, by exceeding the link error monitor (LEM) threshold during operation, or because it is part of an illegal topology.

In use	The physical port established a connection and is fully operational.
Off maintenance	The physical port is reserved for diagnostic testing and loopbacks.
Off ready	The physical port is disabled.
Starting	The physical port received a response from its neighbor physical port and is exchanging information and performing the link confidence test (LCT) before completing the connection.
Unknown	The condition of the physical port is not known.
Waiting	The physical port is establishing a connection and is waiting for a response from its neighbor physical port.
Watching	Same as Starting, except that the physical port failed at least once; by failing the link confidence test (LCT) during initialization, by exceeding the link error monitor (LEM) threshold during operation, or because it is part of an illegal topology.

Physical Port UID

The 48-bit address of the physical port.

Reject Reason

The reason that the last connection on the physical port was lost. This field is updated every time the physical port loops through the Failed and Watching states. This field can have one of the following values:

LCT Both	The link confidence test (LCT) failed on both this physical port and the neighbor physical port.
LCT Local	The link confidence test (LCT) failed on this physical port.
LCT Remote	The link confidence test (LCT) failed on the neighbor physical port.
LEM Failure	The bit error rate on the physical port exceeded the link error monitor (LEM) threshold. The LEM monitors the quality of the link during operation.
No Reason	The physical port is initializing. This value is cleared when the physical port enters the In Use state.
Remote Reject	The neighbor physical port broke the connection for an unknown reason.
Standby	The physical port is not ready, it is initializing.

TNE Expired	The noise timer expired because a single noise event lasted for more than 1.31072 milliseconds. The noise timer is operational only when the physical port is In Use.
Topology Rules	The neighbor physical port is an illegal match for this physical port; for example, an A and an A or a Master and a Master.
Trace in Progress	A PC Trace occurred while the physical port was initializing. When a PC trace occurs, any physical ports that have not established a connection are shut down to prevent the topology from changing.
Trace Received-Trace Off	The physical port was momentarily disabled because it received a PC trace when its own PC trace function was disabled. The Trace Disable switch is designed to protect the physical port from faulty implementations of the PC trace algorithm. The Trace Disable switch is not remotely manageable.

Ring Error Reason

The reason there is an error condition on the ring. This field can have one of the following values:

Bridge Strip Error	A station using bridge frame stripping received a token before decrementing its Sent count to zero. In bridge strip mode, the station maintains a count of frames sent since obtaining the token, and decrements the count each time one of its frames returns.
Directed Beacon Received	A station that is stuck beaoning sent a frame to the directed beacon multicast address, indicating the suspected cause of the ring break. (A station is stuck beaoning when its FDDI MAC entity has been beaoning longer than the time defined by the ANSI FDDI parameter T_Stuck.) This is the last recovery procedure before initiating the PC trace.
Duplicate Address Detected	A station detected a duplicate of its own address.
Duplicate Token Detected	A station received a token while it was holding the token.
No Reason	The ring is operating correctly.

PC Trace Initiated	A station that is stuck beaconing has forced its upstream neighbors to perform their self-tests. (A station is stuck beaconing when its FDDI MAC entity has been beaconing longer than the time defined by the ANSI FDDI parameter T_Stuck.) PC trace is the most drastic fault recovery procedure.
PC Trace Received	The station received a PC trace frame, instructing the station to initiate a self-test.
Ring Beaconing Initiated	A station initiated the ring beacon process because its TRT timer expired before the claim token process recovered the ring. The beacon process locates the ring break. The station downstream from the break will be stuck beaconing. (A station is stuck beaconing when its FDDI MAC entity has been beaconing longer than the time defined by the ANSI FDDI parameter T_Stuck.)
Ring Init Initiated	The FDDI MAC entity of this station initiated the claim token process because it detected a configuration change or a missing token.
Ring Init Received	Another station initiated the claim token process because it detected a configuration change or a missing token.
Ring OP Oscillation	The ring is suffering from ring OP (operational) oscillation. That is, it repeatedly comes up briefly and then goes back into initialization. This problem is frequently caused by a duplicate address condition.
Ring Purge Error	The station serving as the ring purger received a token when it was not expecting one. The station expects two void frames and then the token when it is serving as the ring purger.

Ring Latency

The amount of time (in milliseconds) for a signal element to proceed completely around the entire ring.

Ring Purge Address

The 48-bit data link address of the station currently elected as Ring Purger.

Ring Purger State

The state of the ring purger algorithm of the station's FDDI MAC entity. This field can have one of the following values:

Candidate	The ring is operational and the FDDI MAC entity is bidding to become the ring purger by sending Candidate Hello frames to the ring purger multicast address. The station with the highest station ID becomes the ring purger.
Non Purger	The ring is operational and the FDDI MAC entity is not the ring purger, either because another station won the candidate bidding or because this line has a duplicate address.
Purger	The ring is operational and the FDDI MAC entity is serving as ring purger, constantly purging the ring of fragments and no-owner frames. The station periodically sends Ring Purger Hello frames to the ring purger multicast address.
Purger Off	The ring purger algorithm is not active because the ring is not operational.

Station State

The state of the station. This field can have one of the following values:

Loopback	The station is enabled to operate in loopback mode; it will not connect to the ring.
Off	The station is disabled.
On	The station is enabled to operate in normal operating mode.

Station UID

The 48-bit ID of the FDDI port of the station. The first two bytes are zero (0). The remaining bytes are the link address value of the first MAC of the station.

Upstream Neighbor Address

The 48-bit hardware address of the station that is on the upstream side of the ring from this station.

Upstream Neighbor Dup Addr Flag

The upstream neighbor's duplicate address status. This field can have one of the following values:

Absent	The duplicate address test passed.
Present	The duplicate address test failed.

A.2.3 FDDI Characteristics

This section lists FDDI characteristics alphabetically.

LEM Threshold

The link error monitor (LEM) threshold set for the physical port. The LEM monitors the bit error rate (BER) on the physical port during normal operation. When the bit error rate rises above the LEM threshold, the station disables the physical port, preventing it from disrupting the ring.

The LEM threshold is expressed as the absolute value of the exponent of the bit error rate. The legal range for the threshold is 5 through 8, corresponding to the range of bit error rates, which is 1×10^{-5} (0.00001) bit errors per second through 1×10^{-8} (0.00000001) bit errors per second.

Link Address

The 48-bit hardware address of this FDDI network interface.

Physical Port Type

The type of the neighbor physical port. This field can have one of the following values:

- A The physical port on a dual attachment wiring concentrator (DAC) or dual attachment station (DAS) that connects to the incoming primary ring and the outgoing secondary ring of the FDDI dual ring.
- B The physical port on a dual attachment wiring concentrator (DAC) or dual attachment station (DAS) that connects to the outgoing primary ring and the incoming secondary ring of the FDDI dual ring.
- Master One of the physical ports on a wiring concentrator that connects to a single attachment station (SAS) such as a DECbridge 500 device.
- Slave The physical port on a single attachment station (SAS) that connects to a wiring concentrator or another SAS.
- Unknown No connection has been established.

PMD Type

The type of physical medium to which this physical port is attached. This field can have one of the following values:

ANSI Multimode	Inexpensive thick core fiber combined with light-emitting diode (LED) sources and p-type intrinsic n-type (PIN) detectors.
ANSI Singlemode Type 1	Expensive thin core fiber combined with laser diode sources and avalanche photodiode (APD) detectors.
ANSI Singlemode Type 2	Expensive thin core fiber combined with laser diode sources and avalanche photodiode (APD) detectors.
ANSI SONET	Synchronous Optical Network

Requested TRT

The ANSI MAC parameter T_req, which is the requested value for the Token Rotation Timer. The default value is 8.0 milliseconds.

Restricted Token Timeout

This value limits how long a single restricted mode dialog can last before being terminated.

Ring Purger Enable

If True, this link participates in the Ring Purger election. If elected, the link performs the Ring Purger function.

SMT Max Version ID

The highest value supported for SMT Version ID. A value of 1 corresponds to SMT Revision 6.2.

SMT Min Version ID

The lowest value supported for SMT Version ID. A value of 1 corresponds to SMT Revision 6.2.

SMT Version ID

The version number of the FDDI Station Management (SMT) protocol.

Station ID

The 48-bit ID of this FDDI network interface for station management (SMT). The first two bytes are zero (0). The remaining bytes are the link address value of the first MAC of the station.

Station Type

The type of station. This field can have one of the following values:

DAS	A dual attachment station (DAS). A station that has one or two links and two physical ports, one of type A and one of type B.
SAS	A single attachment station (SAS).

Valid Transmission Time

The valid transmission time (TVX) used by the FDDI MAC entity. If the FDDI MAC entity does not receive a valid frame or unrestricted token within the valid transmission time, it initializes the ring. The default value is 2.621 milliseconds.

A.3 Monitoring the Token Ring Interface

You can use the `netstat -I tra0 -s` command to obtain a listing of the token ring counters and other attributes. The following is sample system output from this command:

```
tra0 Token ring counters at Thu Mar 24 07:33:00 1993
 82502 seconds since last zeroed
 2230 bytes received
 1704 bytes sent
   34 data blocks received
   20 data blocks sent
 288 multicast bytes received
   8 multicast blocks received
 306 multicast bytes sent
  13 multicast blocks sent
   0 unrecognized frames
   0 unrecognized multicast frames
   0 transmit failures
   0 transmit underrun errors
   1 line errors
   9 internal errors
   4 burst errors
   0 ARI/FCI errors
   0 abort delimiters transmitted
   3 lost frame errors
   0 receive data overruns
   0 frame copied errors
   0 token errors
   9 hard errors
   3 soft errors
   1 adapter resets
   1 signal loss
   5 beacon transmits
   2 ring recoveries
   0 lobe wire faults
   0 removes received
```

```

    0 single stations
    0 self test failures
tra0 Token ring and host information:
MAC address:                00-00-C9-19-4A-F3
Group address:              00-C0-00-80-00-00
Functional address:         00-C0-00-00-00-00
Physical drop number:      0
Upstream neighbor address: 00-00-10-C9-F5-3B
Upstream physical drop number: 0
Transmit access priority:  0
Last major vector:         Standby monitor present
Ring status:                No problems detected
Monitor contender:         Yes
Soft error timer value:    2000 ms
Local ring number:         0
Reason for transmitting beacon: No beacon
Reason for receiving beacon:  No beacon
Last beacon upstream neighbor address: 00-00-10-C9-F3-4A
Beacon station physical drop number: 0
Ring speed:                 4Mbps
Early token release:        False
Open status:                Open
Token ring chip:            TMS380C26

```

A.3.1 Token Ring Counters

This section lists the token ring counters alphabetically.

abort delimiters transmitted

The number of times an abort delimiter was transmitted while transmitting data.

adapter resets

The number of times the adapter was reset.

ARI/FCI errors

The number of times a standby monitor present (SMP) MAC frame or active monitor present (AMP) MAC frame was received with the address recognized indicator (ARI) or frame copied indicator (FCI) bits set to zero, followed by another SMP MAC frame with the ARI and FCI bits set to zero.

beacon transmits

The number of beacon MAC frames transmitted.

burst errors

The number of times a burst error was detected.

bytes received

The number of bytes successfully received.

bytes sent

The number of bytes successfully transmitted.

data blocks received

The number of frames successfully received.

data blocks sent

The number of frames successfully transmitted.

frame copied errors

The number of times a frame with a station's recognized address had the frame copied indicator (FCI) set.

hard errors

The number of times a streaming error, frequency error, signal loss error, or internal error was detected.

internal errors

The number of times a recoverable internal error was detected.

line errors

The number of times a frame was repeated or copied, the error detected indicator (EDI) was zero in the incoming frame, or one of the following occurred:

- A code violation occurred between the starting delimiter and ending delimiter of the frame
- A code violation existed in the token
- A frame check sequence (FCS) error occurred

lobe wire faults

The number of times a wire fault condition was detected.

lost frame errors

The number of times an adapter was transmitting data and failed to receive the end of the frame it transmitted.

multicast blocks received

The number of frames successfully received in multicast frames.

multicast blocks sent

The number of frames successfully transmitted in multicast frames.

multicast bytes received

The number of bytes successfully received in multicast frames.

multicast bytes sent

The number of bytes successfully transmitted in multicast frames.

receive data overruns

The number of times a frame was received and the station had no available buffer space.

removes received

The number of times a remove ring station MAC frame was received.

ring recoveries

The number of times a ring recovery has occurred.

seconds since last zeroed

The number of seconds since the associated counter attributes were set to zero.

self test failures

The number of times the self test has failed.

signal loss

The number of times a broken ring, faulty wiring concentrator, transmitter malfunction, or receiver malfunction was detected.

single stations

The number of times there was only one station on the ring.

soft errors

The number of times an error MAC frame was transmitted.

token errors

The number of times an active monitor recognized an error condition that required a token be transmitted.

transmit failures

The number of times a transmit error (other than transmit underrun) occurred.

transmit underrun errors

The number of times a transmit underrun error occurred. This indicates the transmit first-in/first-out (FIFO) buffer became empty during frame transmission.

unrecognized frames

The number of times a received, individually addressed logical link control (LLC) frame was discarded because there was no data link port.

unrecognized multicast frames

The number of times a received LLC frame addressed to a multicast address was discarded because there was no data link port.

A.3.2 Token Ring and Host Information

This section lists the token ring and host information alphabetically.

Beacon station physical drop number

The physical location of the upstream station that transmitted a beacon.

Early token release

This field can have one of the following values:

- | | |
|-------|---|
| True | The station will release the token when it completes frame transmission. The default for 16 Mb/s rings. |
| False | The station will release the token when it receives the transmitted frame header. The default for 4 Mb/s rings. |

Functional address

The functional address of the station. Functional addresses identify predefined devices through bit-significant locally-administered group addresses. Some devices include:

Active monitor	C0 00 00 00 00 01
Ring Parameter Server (RPS)	C0 00 00 00 00 02
Ring Error Monitor (REM)	C0 00 00 00 00 08
Configuration Report Server (CRS)	C0 00 00 00 00 10
Source Route Bridge (SRB)	C0 00 00 00 01 00

Group address

The group address of the station.

Last beacon upstream neighbor address

The address of the upstream station that transmitted a beacon.

Last major vector

The function the adapter is to perform. This field can have one of the following values:

Active monitor present	The active monitor requested a standby monitor present MAC frame from its nearest downstream neighbor.
Beacon	Used by the adapter in the beacon process.
Change parameters	The network manager is changing adapter parameters.
Claim token	Used by the adapter in the monitor contention process.
Duplicate address test	The adapter is verifying that its address is unique on the ring.
Initialize ring station	The ring parameter server is setting adapter parameters.
Lobe media test	The adapter is testing the continuity of the wire in a loopback path.
Remove ring station	The network manager is requesting the adapter to remove itself from the ring.
Report error	The adapter is reporting soft error events to the ring error monitor.

Report monitor error	The adapter is reporting a problem with the active monitor or a possible duplicate station address to the ring error monitor.
Report new monitor	The active monitor adapter, after winning contention, is reporting this status to the network manager.
Report ring poll failure	The active monitor is reporting a failure in the ring poll process to the ring error monitor.
Report station address	The adapter is reporting its station address to the network manager.
Report station attachment	The adapter is reporting its attachment status to the network manager.
Report station state	The adapter is reporting its state to the network manager.
Report SUA change	The adapter is reporting a change in the stored upstream address (SUA) to the network manager.
Report transmit forward	The adapter is reporting a frame that has been forwarded and stripped to the network manager.
Request initialization	The adapter is requesting operational parameters from the ring parameter server.
Request station address	The network manager is requesting a report station address MAC frame from the adapter.
Request station attachment	The network manager is requesting a report station attachment MAC frame from the adapter.
Request station state	The network manager is requesting a report station state MAC frame from the adapter.
Response	The adapter is sending a positive acknowledgement to frames that require acknowledgement or is reporting syntax errors in the MAC frame.
Ring purge	Used by the active monitor during the ring purge process.
Standby monitor present	The adapter is responding to an active monitor present or standby monitor present MAC frame.
Transmit forward	Used in the transmit forward process.

Local ring number

The local ring number of the station.

MAC address

The MAC address of the station.

Monitor contender

Indicates whether the station will participate in the monitor contention process. This field can have the following values:

- | | |
|-----|---|
| No | The station will not participate in the monitor contention process. |
| Yes | The station will participate in the monitor contention process. |

Open status

The status of the adapter on the ring. This field can have one of the following values:

- | | |
|-------|---|
| Close | The adapter is not operational on the ring. |
| Open | The adapter is operational on the ring. |

Physical drop number

The physical location of the station.

Reason for receiving beacon

The reason why the adapter is receiving a beacon MAC frame. This field can have one of the following values:

- | | |
|----------------------|--|
| Bit streaming | A monitor contention timeout occurred while an adapter was in monitor contention transmit mode and before a claim token MAC frame was received. |
| Contention streaming | A monitor contention timeout occurred while an adapter was in monitor contention mode (transmit or receive) and received one or more claim token MAC frames. |
| No beacon | The adapter is not receiving a beacon MAC frame. |
| Signal loss | An adapter detected a signal loss. |

Reason for transmitting beacon

The reason why the adapter is transmitting a beacon MAC frame. This field can have one of the following values:

- | | |
|---------------|--|
| Bit streaming | A monitor contention timeout occurred while the adapter was in monitor contention transmit mode and before a claim token MAC frame was received. |
|---------------|--|

Contention streaming	A monitor contention timeout occurred while the adapter was in monitor contention mode (transmit or receive) and received one or more claim token MAC frames.
No beacon	The adapter is not transmitting a beacon MAC frame.
Signal loss	The adapter detected a signal loss on the ring.

Ring speed

The ring speed: 4 Mb/s or 16 Mb/s.

Ring status

Status reported by the adapter to the driver. This field can have one of the following values:

Auto removal error	The adapter failed the lobe wrap test and removed itself from the ring.
Counter overflow	One of the adapter's error counters has exceeded its maximum value.
Hard error	The adapter is transmitting beacon frames to or receiving beacon frames from the ring.
Lobe wire fault	The adapter detected an open or short circuit in the cable between the adapter and the wiring concentrator.
No problems detected	The ring is operating normally.
Remove received	The adapter received a remove ring station MAC frame request and removed itself from the ring.
Ring recovery	The adapter is observing claim token MAC frames on the ring.
Signal loss	The adapter detected a loss of signal on the ring.
Single station	The adapter sensed that it is the only station on the ring.
Soft error	The adapter transmitted a report error MAC frame.
Transmit beacon	The adapter is transmitting beacon frames on the ring.

Soft error timer value

The number of milliseconds that elapse from the time the adapter detects a soft error until it sends a report error MAC frame to the ring error monitor.

Token ring chip

The type of chip used by the sending station.

Transmit access priority

The priority level at which this station can access the ring. This field can have a value from 0 (lowest priority) to 7 (highest priority).

Upstream neighbor address

The address of the upstream station.

Upstream physical drop number

The location of the upstream station.

Index

Numbers and Special Characters

6bone

- address assignment, 3–10
- connecting to, 3–34

A

acucap file, 6–32

adapters

(*See* network interfaces)

address

- and CIDR, 3–3
- assignment, 3–10
- for 6bone testing, 3–10
- IPv6, 3–2
- mapping to name, 3–10
- multicast, 3–7
- not autoconfigured, 8–19
- size, 3–2
- text representation, 3–2
- unicast, 3–4

address autoconfiguration

- DHCPv6, 3–9
- stateful, 3–9
- stateless, 3–9

address prefix, 3–9

- advertising on a link, 3–22
- advertising on a tunnel, 3–21
- remote network, 3–22

Address Resolution Protocol

(*See* ARP)

ARP

- specifying server ATM address, 4–10

- specifying server IP address, 4–11
- specifying system role, 4–10

arp command, 9–7

ARP server

- and /etc/atmhosts file, 4–17

ARP table

- and CLIP, 4–26
- creating entries, 4–26
- deleting entries, 4–26
- displaying for LANE, 4–27

Asynchronous Transfer Mode

(*See* ATM)

ATM

- changing message level, 4–28
- configuring, 4–15
- configuring with CLIP, 4–17
- configuring with IP switching, 4–22
- configuring with LANE, 4–20
- disabling device driver, 4–26
- displaying network information, 4–26
- enabling device driver, 4–26
- information required for configuration, 4–7
- managing signaling, 4–26
- managing the environment, 4–25
- messages, 4–28
- preparing for configuration, 4–7
- specifying ESIs, 4–8
- specifying flow control, 4–8
- specifying ILMI, 4–8
- specifying network layer, 4–8
- specifying signaling, 4–9
- specifying UNI version, 4–9
- specifying VC accounting, 4–9
- troubleshooting, 8–21

- verifying kernel options, 4–6
- verifying subsets are installed, 4–6
- ATM Configuration application**, 4–15
 - configuring CLIP, 4–19
 - configuring IP switching, 4–23
 - configuring LANE, 4–21
- atmarp command**, 4–26
- atmconfig command**, 4–26
- atmelan command**, 4–27
- atmhosts file**
 - and ARP server, 4–17
 - editing for CLIP, 4–17
 - editing for LANE, 4–21
- atmifmp command**, 4–27
- atmsig command**, 4–26
- authentication**
 - and PPP, 6–15

B

- BOOTP**
 - and DHCP, 5–20
 - configuring a client, 5–20
- bound interactive service**, 7–17
- broadcast address**
 - (*See* multicast address)
- BSD devices**
 - LAT, 7–6

C

- cable**
 - guidelines for use with modems, 6–29
 - null modem, 6–3
- CDSL**, 1–9
- Challenge Authentication Protocol**
 - (*See* CHAP)
- CHAP**
 - chap-secrets file, 6–15, 6–23
 - use with PPP, 6–15
- chat script**, 6–13

- CIDR**, 3–3
- Classical IP**
 - (*See* CLIP)
- Classless Inter-Domain Routing**
 - (*See* CIDR)
- client**
 - DHCP, 5–2
 - monitoring DHCP, 5–18
 - obtaining IP address for, 2–10
- CLIP**
 - adding static routes, 4–20
 - configuring LIS interfaces, 4–20
 - configuring with ATM, 4–17
 - description, 4–2
 - editing */etc/atmhosts* file, 4–17
 - editing */etc/hosts* file, 4–18
 - planning configuration with ATM, 4–9
 - running ATM Configuration application, 4–19
 - setting up PVCs, 4–19
 - specifying remote host support, 4–11
 - troubleshooting, 8–22
- cloning**
 - installation and configuration, 1–9
- Compaq Analyze**, 9–11
- Compaq Insight Manager**, 1–7
- configuration cloning**, 1–9
- configured tunnel**
 - running RIPng, 3–21
- connection**
 - terminates abnormally (IPv6 host), 8–14
 - terminates abnormally (IPv6 router), 8–20
- connection refused message**
 - IPv6 host, 8–14
 - IPv6 router, 8–20
- connections**
 - configuring system for dial-in access, 6–29
 - configuring system for dial-out access, 6–32

- LAT host-initiated, 7–13
- outgoing (LAT), 7–15
- terminating SLIP, 6–10

Context-Dependent Symbolic Link

(*See* CDSL)

counters

- Ethernet, A–1
- FDDI, A–6
- token ring, A–20

D

datagram

- displaying route through a network, 9–8

DECEvent, 9–11

dedicated service, 7–17

dedicated tty device on a terminal, 7–18

destination prefix, 3–22

DHCP, 5–1

- and security, 5–3, 5–19
- configuration worksheet, 5–4, 5–8
- configuring a client, 5–15
- configuring a server, 5–12
- disabling address assignment, 5–20
- identifying server during configuration, 2–10
- information required for configuration, 5–4
- joind daemon, 5–17
- mapping hardware addresses, 5–19
- monitoring clients, 5–18
- planning for configuration, 5–2
- specifying during network configuration, 2–10
- starting clients, 5–18
- starting servers, 5–17
- troubleshooting, 8–29
- xjoin, 5–12

dial-in connections, 6–29

- PPP, 6–26
- SLIP, 6–8

dial-out connections, 6–32

- PPP, 6–19
- SLIP, 6–9

DNS

- domain for reverse lookup, 3–10
- IPv6 data format, 3–10
- IPv6 record type, 3–10

Domain Name System

(*See* DNS)

dropped packets, 2–39

Dynamic Host Configuration

Protocol

(*See* DHCP)

E

ELAN

- and /etc/hosts file, 4–21
- configuring elan interfaces, 4–22
- specifying name, 4–12
- specifying number, 4–12

emulated LAN

(*See* ELAN)

end system

- definition, 4–1

end system identifier

(*See* ESI)

error log file

- viewing, 9–11

error messages, 8–1

(*See also* problem; troubleshooting)

ESI

- creating, 4–26
- destroying, 4–26
- specifying for ATM adapter, 4–8

/etc files

(*See* files)

Ethernet

- configuration worksheet, 2–8
- configuring, 2–19
- counters, A–1
- information required for
 - configuration, 2–8
 - monitoring, A–1
- Event Viewer**
 - viewing syslogd message files, 9–11

F

FDDI

- configuration worksheet, 2–8
- configuring, 2–19
- displaying information about, 2–35
- displaying parameters, 2–35
- information required for
 - configuration, 2–8
- interface characteristics, A–17
- interface counters, A–6
- interface status, A–9
- modifying parameters, 2–35
- monitoring, A–4
- using netstat to monitor, 9–6

fdi_config command, 2–35

Fiber Distributed Data Interface

(*See* FDDI)

files

- acucap, 6–32
- chap-secrets, 6–15, 6–23
- configuration worksheet, 2–16
- editing manually, 1–9
- gated.conf, 2–15
- gateways, 2–14
- hosts, 2–26
- hosts.equiv, 2–26
- ifaccess.conf, 2–35
- inittab, 7–9
- ip6rtrd.conf, 3–40
- latstartup.conf, 7–8
- networks, 2–27
- options, 6–21, 6–27
- pap-secrets, 6–15, 6–22

- rc.config, 1–9, 2–28, 3–38
- remote, 6–33
- routes, 2–25
- secrets, 6–22
- slhosts, 6–7

flow, 4–4

- displaying information about, 4–27

format prefix, 3–9

G

gated daemon, 2–14

- configuring, 2–23

gated.conf file, 2–15

gateway

- and PPP, 6–27
- and SLIP, 6–8

gateways file, 2–14

H

host, 8–10

(*See also* node)

- configuring for IPv6, 3–30
- creating lists for DHCP, 5–14
- IPv6 configuration variables, 3–38
- static routes, 3–20
- testing Internet access, 9–4
- unknown (IPv6 host), 8–10

host is unreachable message

- off-link node (IPv6 host), 8–13
- off-link node (IPv6 router), 8–18
- on-link node (IPv6 host), 8–11
- on-link node (IPv6 router), 8–16

host name

- configuring network interface, 2–10

hosts file

- and ELAN hosts, 4–21
- configuring, 2–26
- defined, 2–17
- editing for CLIP, 4–18
- editing for IP switching, 4–23
- editing for LANE, 4–21

hosts.equiv file
configuring, 2–26
defined, 2–18

ICMP messages, 9–10
ifaccess.conf file, 2–35
ifconfig command

adding an IPv6 address, 3–36
assigning an IPv6 interface ID,
3–35
deleting an IPv6 address, 3–37
enabling access filtering with, 2–35
initializing an IPv6 interface, 3–35
removing an IPv6 interface, 3–35

ILMI
specifying for ATM adapter, 4–8

inittab file
customizing for LAT, 7–9

Insight Manager, 1–7

installation cloning, 1–9

Integrated Layer Management

Interface
(See ILMI)

interface ID
and MAC address, 3–4
and unicast address, 3–4
assigning to IPv6 interface, 3–35

interfaces
(See network interfaces)

Internet Protocol address
(See IP address)

Internet Protocol Version 6
(See IPv6)

**Internet to MAC address
translation tables**, 9–7

InterNIC Registration Services
and IP address, 2–10

IP address
and InterNIC Registration Services,
2–10

and network numbers, 2–12
configuring for DHCP, 5–13
obtaining, 2–10

IP aliases, 2–27

IP MTU size, 2–39

IP router
configuring system as, 2–24
defined, 2–15
requirements, 2–24

IP switching
adding routes, 4–25
characteristics, 4–4
configuring ips interfaces, 4–25
configuring with ATM, 4–22
disabling, 4–27
displaying statistics, 4–27
editing /etc/hosts file, 4–23
enabling, 4–27
managing, 4–27
planning configuration with ATM,
4–13

running ATM Configuration
application, 4–23
troubleshooting, 8–26

ip6_setup utility
configuring a host, 3–30
configuring a router, 3–31

ip6rtrd daemon
log file, 3–41
logging debug information, 3–42
not running, 8–15

ip6rtrd.conf file, 3–40

ips
configuring interfaces, 4–25

iptunnel command, 3–36

IPv4 address
and configured tunnel, 3–21

IPv4-compatible IPv6 address,
3–5

IPv4-mapped IPv6 address, 3–6

IPv6
6bone network, 3–34

- address autoconfiguration, 3–9
- addressing, 3–2
- configuration worksheet, 3–18
- configuring a host, 3–30
- configuring a router, 3–31
- configuring support in kernel, 3–38
- creating a configured tunnel, 3–36
- editing rc.config file for, 3–38
- information required for
 - configuration, 3–18
- initializing on an interface, 3–35
- messages, 3–41
- planning, 3–17
- removing from an interface, 3–35
- sample configurations, 3–23
- support required in kernel, 3–17
- supported commands, 3–13
- supported interfaces, 3–13
- terms, 3–2
- troubleshooting, 8–9
- types of addresses, 3–3

IPv6 subsystem
tuning, 3–41

ipv6forwarding attribute, 3–38

ipv6router attribute, 3–38

J

joind daemon, 5–17

K

kernel
configuring ATM in, 4–6
configuring IPv6 routing in, 3–38
configuring PPP in, 6–16

L

lag, 2–6

LAN
running RIPng, 3–22

LAN emulation

(*See* LANE)

LAN Emulation Configuration Server

(*See* LECS)

LAN Emulation Server

(*See* LES)

LANE

- changing message level, 4–28
- characteristics, 4–3
- configuring elan interfaces, 4–22
- configuring with ATM, 4–20
- displaying ARP table, 4–27
- editing /etc/atmhosts file, 4–21
- editing /etc/hosts file, 4–21
- managing the environment, 4–27
- planning configuration with ATM, 4–11
- running ATM Configuration application, 4–21
- troubleshooting, 8–24

LAT

- adding devices, 7–7
- and NetRAIN, 7–7
- automatic startup and shutdown, 7–5, 7–7
- configuration worksheet, 7–5
- configuring kernel for, 7–4
- configuring multiple network adapters, 7–10
- configuring with latsetup, 7–7
- connections, 7–2
- controlling access, 7–3
- creating a startup file, 7–8
- creating your own service, 7–17
- customizing the inittab file, 7–9
- dedicated tty device on a terminal, 7–18
- defining LAT/Telnet service, 7–16
- defining port names, 7–12
- defining server names, 7–12
- devices, 7–6
- gateway service, 7–16
- host-initiated connection, 7–13

- information required for
 - configuration, 7-5
- LAN service, 7-17
- latsetup command, 7-7
- line disciplines, 7-14
- load balancing, 7-4
- outgoing connections, 7-15
- password protection, 7-4
- planning for configuration, 7-4
- printer hardware characteristics, 7-11
- printer setup, 7-10
- program interface, 7-14
- sample configurations, 7-1
- server node, 7-1
- service node, 7-1
- service node groups, 7-3
- services, 7-17
- setup, 7-7
- starting and stopping, 7-7
- terminal server port settings, 7-11
- testing printer settings, 7-12
- testing printer setup, 7-13
- troubleshooting, 8-36
- user-created LAN service, 7-17
- verifying DLB support, 7-4
- LAT/Telnet gateway**, 7-16
 - setup, 7-16
 - startup, 7-16
- latsetup command**, 7-7
- latstartup.conf file**, 7-8
- learp command**, 4-27
- LEC**
 - configuring, 4-27
 - creating, 4-27
 - displaying status, 4-27
- LECS**
 - specifying ATM address, 4-13
- LES**
 - and /etc/atmhosts file, 4-21
 - specifying ATM address, 4-13

- line disciplines**, 7-14
- link aggregation**, 2-6
 - and LAT, 2-33
 - ports, 2-12
- link-local address**, 3-6
- LIS**
 - configuring lis interfaces, 4-20
 - creating, 4-26
 - definition, 4-2
 - specifying numbers, 4-10
- load balancing**
 - LAT, 7-4
- Local Area Transport**
 - (*See* LAT)
- local node**
 - not reachable (IPv6 host), 8-14
 - not reachable (IPv6 router), 8-19
- log files**, 9-11
 - (*See also* messages)
- loopback address**, 3-5

M

- MAC address**
 - and interface ID, 3-4
- Microsoft Challenge Authentication Protocol**
 - (*See* MS-CHAP)
- Microsoft NT RAS server**, 6-25
 - configuring, 6-25
 - solving CHAP authentication problems, 6-26
- modem**
 - and acucap file, 6-32
 - cable guidelines, 6-29
 - commands for dial-in access, 6-31
 - commands for dial-out access, 6-9
 - guidelines, 6-16, 6-29
 - using with SLIP, 6-3
- monitoring network interfaces**, 9-1
- MS-CHAP**

use with PPP, 6–15

MTU

specifying size, 4–13

multicast address, 3–7

group, 3–8

transient, 3–8

well-known, 3–8

N

name

mapping to address, 3–10

nd6hostd daemon

log file, 3–41

logging debug information, 3–41

netconfig application, 2–19

NetRAIN

and LAT, 2–28

and MAC address licensing, 2–5

configuration worksheet, 2–8

configuring an interface, 2–28

hardware restrictions, 2–28

monitoring the interface, 2–32

set members, 2–12

netstat command, 9–6, A–1

network adapters

(*See* network interfaces)

network data structures

displaying with the netstat

command, 9–6

Network Information Center

(*See* InterNIC Registration Services)

network interfaces, 2–1

adding an IPv6 address to, 3–36

and LAT configuration, 7–10

assigning IPv6 IDs, 3–35

configuring, 2–19

configuring multiple for failover,
2–28

controlling access, 2–35

deconfiguring, 2–21

deleting an IPv6 address from,
3–37

displaying packet headers on, 9–10

information required for

configuration, 2–8, 4–7

initializing for IPv6, 3–35

link aggregation, 2–12

managing, 2–27

monitoring, 2–32, 9–1, 9–6, A–1

multiple in same subnetwork, 2–3,
2–27

NetRAIN, 2–4, 2–12

removing IPv6 from, 3–35

token ring, 2–12

types of, 2–9

network is unreachable message

off-link node (IPv6 host), 8–13

off-link node (IPv6 router), 8–18

on-link node (IPv6 host), 8–11

on-link node (IPv6 router), 8–16

network mask

(*See* subnet mask)

network problems, 8–1

(*See also* problem;
troubleshooting)

gathering information, 10–1

reporting, 10–1

tools for solving, 9–1

Network Setup Wizard, 1–5

networks file

configuring, 2–27

defined, 2–18

niffd daemon, 9–1

node, 8–11, 8–14, 8–19

(*See also* local node; off-link
node; on-link node)

defined, 3–2

nr interface, 2–4

null modem cable, 6–3

O

off-link node

unreachable (IPv6 host), 8–13

unreachable (IPv6 router), 8–18

on-link node

- cannot autoconfigure addresses, 8–19
- not reachable (IPv6 host), 8–11
- not reachable (IPv6 router), 8–16
- on-link prefix**
 - adding a route for, 3–37
- optional service**, 7–17
- options file**, 6–21, 6–27
- outgoing connections**, 7–15

P

- packets**
 - dropped by bridges, 2–39
 - rejected by remote hosts, 2–39
- PAP**
 - use with PPP, 6–15
- pap-secrets file**, 6–15, 6–22
- Password Authentication Protocol**
 - (*See* PAP)
- password protection**
 - LAT, 7–4
- performance**
 - tuning, 3–41
- permanent virtual circuit**
 - (*See* PVC)
- ping command**, 9–4
- Point-to-Point Protocol**
 - (*See* PPP)
- PPP**, 6–11
 - (*See also* SLIP)
 - and gateways, 6–27
 - chap-secrets file, 6–15, 6–23
 - chat script, 6–13
 - configuration worksheet, 6–17
 - configuring dial-in system, 6–26
 - configuring dial-out system, 6–19
 - connection guidelines, 6–29
 - guidelines for running pppd, 6–24
 - information required for
 - configuration, 6–16
 - Microsoft NT RAS server, 6–25

- options file, 6–21, 6–27
- pap-secrets file, 6–15, 6–22
- sample configurations, 6–11
- security, 6–15
- terminating a connection, 6–28
- troubleshooting, 8–34
- PPP link**
 - running RIPng, 3–22
- pppd daemon**
 - guidelines for running, 6–24
 - options, 6–14, 6–19
- printer**
 - using with LAT, 7–10
- printer setup**
 - LAT, 7–11
 - testing, 7–13
- problem**
 - connection refused, 8–14, 8–20
 - connection terminates abnormally, 8–14, 8–20
 - connection timed out, 8–8
 - host is unreachable, 8–7
 - network daemon not running, 8–6
 - network is unreachable, 8–6
 - network software not configured, 8–5
 - off-link node is not reachable, 8–13
 - off-link node is unreachable, 8–18
 - on-link node cannot autoconfigure addresses, 8–19
 - on-link node not reachable, 8–11, 8–16
 - remote host is unknown, 8–6, 8–10, 8–15
 - router daemon not running, 8–15
 - router not forwarding messages, 8–19
 - your node not reachable, 8–14, 8–19
- problem solving**, 8–1
 - (*See also* troubleshooting)
 - diagnostic map, 8–1

tools, 9–1

PVC

configuring for CLIP, 4–19

creating, 4–26

destroying, 4–26

verifying creation, 4–20

Q

QoS

defined, 2–40

quality of service

(*See* QoS)

Quick Setup, 1–4

R

rc.config file

editing with rcmgr utility, 1–9

IPv6 variables, 3–38

NetRAIN, 2–28

rejected packets, 2–39

remote file, 6–29, 6–33

remote host, 8–10

(*See also* host)

unknown (IPv6 router), 8–15

remote node

connection refused (IPv6 host),
8–14

connection refused (IPv6 router),
8–20

resource record, 3–10

Resource ReSerVation Protocol

(*See* RSVP)

RIPng

running on a PPP link, 3–22

running on a tunnel, 3–21

route

adding for an on-link prefix, 3–37

static, 2–16, 2–25, 3–20, 3–22,
3–36

route command

adding a router (IPv6), 3–37

deleting a router (IPv6), 3–37

routed daemon, 2–13

configuring, 2–22

defined, 2–13

router

adding default for IPv6, 3–37

configuring for IPv6, 3–31

configuring kernel support for IPv6,
3–38

daemon not running, 8–15

deleting default for IPv6, 3–37

editing the IPv6 configuration file,
3–40

IPv6 configuration variables, 3–38

not forwarding messages, 8–19

static routes, 3–20

router advertisements

purpose, 3–40

routes

exchanging through a tunnel, 3–21

exchanging with other routers,
3–22

routes file

configuring, 2–25

defined, 2–16

routing, 2–16

dynamic, 2–13

source, 2–37

static, 2–16, 2–25

RSVP

defined, 2–40

managing, 2–41

starting, 2–41

stopping, 2–41

rwhod daemon, 2–13

configuring, 2–21

defined, 2–13

S

SDH

specifying for ATM adapter, 4–8

secrets files, 6–22

security

and DHCP, 5–3, 5–19

- and PPP, 6–15
- controlling access to interfaces, 2–35
- Serial Line Internet Protocol**
 - (*See* SLIP)
- server**
 - configuring for DHCP, 5–12
 - DHCP, 5–2
 - LAT, 7–1
 - obtaining IP address from, 2–10
- service**
 - quality of, 2–40
- site-local address**, 3–7
- slhosts file**, 6–7
- SLIP**, 6–1
 - (*See also* PPP)
 - and gateways, 6–8
 - configuration worksheet, 6–4
 - configuring dial-in system, 6–8
 - configuring dial-out system, 6–9
 - information required for
 - configuration, 6–4
 - IP address, 6–5
 - modem guidelines, 6–3
 - planning for configuration, 6–2
 - sample configurations, 6–1
 - startslip command, 6–5
 - terminating a connection, 6–10
 - troubleshooting, 8–32
- SNAP**
 - specifying default VCI, 4–15
- SONET**
 - specifying for ATM adapter, 4–8
- source routing**, 2–37
 - (*See also* token ring)
- srconfig command**, 2–37
- startppp command**
 - use for dial-in connections, 6–27
- startslip command**, 6–5
 - invoking subcommands from script file, 6–10
 - subcommands, 6–5
 - use for dial-in connections, 6–8
- static routes**, 2–16, 2–25, 3–20, 3–36
- subnet mask**, 2–11
 - and network class, 2–12
 - defined, 2–11
 - for IP switching, 4–5
- subnetwork**
 - and DHCP, 5–14
 - multiple interfaces in, 2–3, 2–27
- Subnetwork Attachment Point**
 - (*See* SNAP)
- SVC**
 - destroying, 4–26
- SVR4 devices**
 - LAT, 7–6
- switch**
 - definition, 4–1
- Synchronous Data Hierarchy**
 - (*See* SDH)
- Synchronous Optical Network**
 - (*See* SONET)
- syslogd daemon**, 9–11
- SysMan Menu**, 1–2
 - configuring ATM, 4–15
 - configuring IP router, 2–24
 - configuring LAT, 7–7
 - creating PPP options file, 6–21, 6–27
 - deconfiguring network interfaces, 2–21
 - invoking, 1–2
 - modifying PPP chap-secrets file, 6–23
 - modifying PPP pap-secrets file, 6–22
 - Network Setup Wizard, 1–5
 - Quick Setup, 1–2, 1–4
 - setting up gate daemon, 2–23
 - setting up hosts file, 2–26
 - setting up hosts.equiv file, 2–26
 - setting up network interfaces, 2–19

setting up networks file, 2-27
setting up route daemon, 2-22
setting up rwho daemon, 2-21
setting up static routes file, 2-25
viewing syslogd message files, 9-11

T

tcpdump command, 9-10

timeout message

off-link node (IPv6 host), 8-13
off-link node (IPv6 router), 8-18
on-link node (IPv6 host), 8-11
on-link node (IPv6 router), 8-16

tip command, 6-20, 6-30

token ring

adapter speed, 2-12
configuration worksheet, 2-8
configuring, 2-19
counters, A-20
displaying IP MTU size, 2-39
host information, A-23
information required for
 configuration, 2-8
IP MTU size, 2-39
modifying IP MTU size, 2-39
monitoring, A-19
source routing, 2-37
using netstat to monitor, 9-6

traceroute command, 9-8

traffic control, 2-41

troubleshooting, 8-1

ATM, 8-21
CLIP, 8-22
DHCP, 8-29
IP switching, 8-26
IPv6, 8-9
LANE, 8-24
LAT, 8-36
PPP, 8-34
SLIP, 8-32

trunk group, 2-6

trunking

(*See link aggregation*)

tuning

IPv6 subsystem, 3-41

tunnel

automatic, 3-20
configured, 3-20
creating, 3-36
specifying during configuration,
 3-22
to 6bone, 3-34

U

uerf command, 9-11

unbound interactive service, 7-17

UNI

version number for ATM adapter,
 4-9

unicast address

defined, 3-4
IPv4-compatible IPv6 address, 3-5
IPv4-mapped IPv6 address, 3-6
link-local address, 3-6
loopback address, 3-5
site-local address, 3-7
unspecified address, 3-5

unknown host message

IPv6 host, 8-10
IPv6 router, 8-15

unspecified address, 3-5

User-Network Interface

(*See UNI*)

V

VC

accounting for ATM adapter, 4-9
monitoring, 4-26

VCI

specifying for PVC, 4-11

virtual channel identifier

(*See VCI*)

virtual circuit

(*See VC*)

virtual path identifier
(*See* VPI)
VPI
specifying for PVC, 4–11

W

well-known multicast addresses,
3–8

X

xjoin utility, 5–12