Tru64 UNIX

Network Administration

Part Number: AA-PS2SD-TE

April 1999

Product Version: Tru64 UNIX Version 4.0F 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.

Compaq Computer Corporation
Houston, Texas

Contents

About This Manual

Part 1 Management Information

1 Overview of Network Administration

2 Basic Network Connections

2.1 Network Environment 2–1
2.2 Preparing for the Configuration 2–2
2.2.1 Information for Interfaces and Daemons 2–2
2.2.1.1 All Network Interfaces 2–3
2.2.1.2 Token Ring Interface 2–6
2.2.1.3 NetRAIN Interface 2–6
2.2.1.4 rwhod Daemon 2–6
2.2.1.5 routed daemon 2–6
2.2.1.6 Gateways File 2–7
2.2.1.7 gated Daemon 2–8
2.2.1.8 IP Router 2–8
2.2.2 Information for Network Files 2–8
2.2.2.1 Static Routes File (/etc/routes) 2–9
2.2.2.2 Host File (/etc/hosts) 2–9
2.2.2.3 hosts.equiv File 2–10
2.2.2.4 Networks File (/etc/networks) 2–10
2.3 Configuring the Network Components 2–11
2.3.1 Configuring Network Interfaces 2–12
2.3.2 Configuring the rwhod Daemon 2–13
2.3.3 Configuring the routed Daemon 2–14
2.3.4 Configuring the gated Daemon 2–15
2.3.5 Configuring the System as an IP Router 2–15
2.3.6 Configuring the Static Routes File 2–16
2.3.7 Configuring the hosts File 2–16
2.3.8 Configuring the hosts.equiv File 2–17
2.3.9 Configuring the networks File 2–17

2.4 NetRAIN Interfaces 2–18
2.4.1 Configuring NetRAIN 2–18
2.4.1.1 Hardware Restrictions and Configuration 2–18
2.4.1.2 Configuring the NetRAIN Interface 2–19
2.4.2 Monitoring NetRAIN Activity 2–20
2.5 Enabling Access Filtering on an Interface 2–20
2.6 Monitoring the Local Host's Status 2–21
2.7 Displaying and Modifying the FDDI Parameters 2–22
2.8 Managing Token Ring Source Routing 2–24
2.9 Displaying and Modifying the Token Ring IP MTU Size 2–26

3 Dynamic Host Configuration Protocol

3.1 DHCP Environment 3–1
3.1.1 DHCP Parameter Assignment 3–2
3.1.2 DHCP and Security 3–3
3.2 Planning DHCP 3–3
3.2.1 Verifying Installation of DHCP Software 3–4
3.2.2 Preparing for the Configuration 3–4
3.2.2.1 Server/Security Parameters 3–4
3.2.2.1.1 IP Ranges 3–6
3.2.2.1.2 Host name list 3–7
3.2.2.2 Information for Basic DHCP Parameters 3–8
3.3 Configuring a DHCP Server 3–12
3.3.1 Configuring Server/Security Parameters 3–13
3.3.2 Configuring IP Ranges 3–13
3.3.3 Configuring Host Name Lists 3–14
3.3.4 Configuring a Subnetwork 3–14
3.3.5 Configuring a DHCP Client Node 3–15
3.3.6 Setting Group Parameters 3–16
3.4 Starting the DHCP Server (joind) 3–16
3.5 Starting the DHCP Client 3–17
3.6 Monitoring DHCP Client Configuration 3–17
3.7 Mapping Client IP Addresses Permanently 3–18
3.8 Restricting Access to the DHCP Server 3–18
3.9 Configuring a BOOTP Client 3–19
3.10 Disabling DHCP Address Assignment 3–19
3.11 Solving DHCP Problems 3–20

4 Point-to-Point Connections

4.1 Serial Line Internet Protocol (SLIP) 4–1

4.1.1 SLIP Environment 4–1
4.1.2 Planning SLIP 4–2
4.1.2.1 Verifying the Hardware 4–3
4.1.2.2 Preparing for the Configuration 4–4
4.1.3 Configuring SLIP 4–8
4.1.3.1 Configuring a Dial-In System 4–8
4.1.3.2 Configuring a Dial-Out System 4–9
4.1.4 Terminating a SLIP Dial-Out Connection 4–10
4.2 Point-to-Point Protocol (PPP) 4–11
4.2.1 PPP Environment 4–11
4.2.1.1 PPP Options 4–13
4.2.1.2 Authentication 4–14
4.2.2 Planning PPP 4–15
4.2.2.1 Verifying the Hardware 4–15
4.2.2.2 Verifying PPP Support in the Kernel 4–15
4.2.2.3 Preparing for Configuration 4–16
4.2.3 Configuring PPP 4–18
4.2.3.1 Establishing a PPP Dial-Out Connection 4–18
4.2.3.2 Establishing a PPP Dial-In Connection 4–22
4.2.4 Terminating PPP Connections 4–23
4.3 Guidelines for Using Modems 4–23
4.3.1 Using the Correct Modem Cables 4–24
4.3.2 Configuring a System for Dial-In Access 4–24
4.3.3 Configuring Your System for Dial-Out Access 4–26

5 Local Area Transport Connections

5.1 LAT Environment 5–1
5.1.1 Types of LAT Connections 5–2
5.1.2 Controlling Access in a LAT Network 5–3
5.1.3 Specifying Passwords for Remote Services 5–4
5.1.4 Load Balancing 5–4
5.2 Planning LAT 5–4
5.2.1 Verifying That the LAT Subset Is Installed 5–4
5.2.2 Verifying DLB Support in the Kernel 5–4
5.2.3 Preparing for the Configuration 5–5
5.3 Configuring LAT 5–6
5.4 Starting and Stopping LAT 5–7
5.5 Creating a LAT Startup File 5–7
5.6 Customizing the inittab File 5–9
5.7 Running LAT Over Specific Network Adapters 5–9
5.8 Setting Up Printers 5–10

5.8.1 Setting Up the Printer on a Terminal Server 5–10
5.8.2 Testing the Port Configuration 5–11
5.8.3 Setting Up a Service Node for the Printer 5–12
5.8.4 Setting Up the Print Spooler on the Service Node 5–12
5.8.5 Testing the Printer 5–13
5.9 Setting Up Host-Initiated Connections 5–13
5.9.1 Setting Up the System for Host-Initiated Connections ... 5–13
5.9.2 Program Interface 5–14
5.10 Setting Up Outgoing Connections 5–14
5.10.1 Setting Up the System for Outgoing Connections 5–15
5.10.2 Program Interface 5–15
5.11 Setting Up the LAT/Telnet Gateway 5–15
5.12 Creating Your Own Dedicated or Optional Service 5–17
5.13 Providing a Dedicated tty Device on a Terminal 5–18
5.13.1 Setting Up a Dedicated tty Device 5–18
5.13.2 Removing a Dedicated tty Device 5–19

6 Domain Name Service

6.1 The DNS Environment 6–1
6.2 Planning DNS 6–4
6.2.1 Server 6–5
6.2.2 Client 6–6
6.3 Configuring DNS 6–7
6.3.1 Configuring a DNS Server 6–8
6.3.2 Configuring a DNS Client 6–10
6.4 Modifying the svc.conf File with svcsetup 6–11
6.5 Updating DNS Data Files on the Primary Server 6–12
6.6 Obtaining Host Name and IP Address Information 6–13
6.6.1 The nslookup Command 6–13
6.6.2 NIC whois Service 6–14

7 Network Information Service

7.1 NIS Environment 7–1
7.2 Planning NIS 7–2
7.2.1 Verifying That the Additional Networking Services Subset is Installed 7–2
7.2.2 Preparing for the Configuration 7–3
7.2.2.1 Master Server 7–4
7.2.2.2 Slave Server 7–6
7.2.2.3 Client 7–7
7.3 Configuring NIS 7–8

7.3.1	Configuring a NIS Master Server	7–9
7.3.2	Configuring a Slave Server	7–13
7.3.3	Configuring an NIS Client	7–16
7.3.4	Modifying the svc.conf File with svcsetup	7–18
7.3.5	Modifying or Removing an NIS Configuration	7–18
7.4	Managing an NIS Server	7–19
7.4.1	Adding an NIS Slave Server to a Domain	7–19
7.4.2	Removing an NIS Slave Server from the Domain	7–21
7.4.3	Adding a New User to an NIS Domain	7–22
7.4.4	Updating an NIS Map	7–23
7.4.5	Adding an NIS Map to a Domain	7–25
7.4.6	Removing an NIS Map from a Domain	7–26
7.4.7	Modifying the /var/yp/Makefile File	7–26
7.4.7.1	Adding an Entry	7–26
7.4.7.2	Deleting an Entry	7–27
7.4.7.3	Makefile Editing Guidelines	7–27
7.4.8	Restricting Access to NIS Data	7–28
7.5	Managing an NIS Client	7–29
7.5.1	Changing an NIS Password	7–29
7.5.2	Obtaining NIS Map Information	7–29

8 Network File System

8.1	NFS Environment	8–1
8.1.1	Distributing the hosts Database	8–1
8.1.2	Automount and NFS	8–1
8.1.2.1	NIS and automount Maps	8–2
8.1.2.2	Local automount Maps	8–2
8.2	Planning NFS	8–3
8.2.1	Server	8–4
8.2.2	Client	8–5
8.3	Configuring NFS	8–6
8.3.1	Configuring an NFS Server	8–7
8.3.2	Configuring an NFS Client	8–8
8.4	Managing an NFS Server	8–9
8.4.1	Exporting a File System or Directory	8–9
8.4.1.1	Exporting a File System with the NFS Configuration Application	8–9
8.4.1.2	Exporting a File System Manually	8–10
8.4.1.3	Export Guidelines	8–11
8.4.2	Halting Export of a Directory or File System	8–12
8.4.3	Enabling Client Superuser Access to Files	8–12

8.4.4 Sending Mail to Superuser (root) Across NFS 8–13
8.4.5 Enabling Port Monitoring 8–14
8.4.6 Monitoring the NFS Load 8–15
8.5 Managing an NFS Client 8–16
8.5.1 Mounting a Remote File System or Directory 8–16
8.5.1.1 Mounting a Remote File System Using NFS Configuration Application 8–16
8.5.1.2 Mounting a Remote File System Manually 8–17
8.5.1.3 Mounting a Remote File System Automatically 8–18
8.5.1.4 Mount Command Options 8–19
8.5.1.5 Using automount to Mount a Remote File System 8–19
8.5.1.6 Specifying automount Arguments 8–21
8.5.2 Unmounting a Remote File System or Directory 8–22

9 UNIX-to-UNIX Copy Program

9.1 UUCP Environment 9–1
9.2 Planning UUCP 9–2
9.2.1 Verifying the Correct Hardware 9–2
9.2.2 Preparing for the Configuration 9–3
9.2.2.1 Information for Connections 9–3
9.2.2.2 Information for Outgoing Systems 9–6
9.2.2.3 Information for Incoming Systems 9–9
9.3 Configuring UUCP 9–11
9.3.1 Configuring Connections 9–12
9.3.2 Configuring Outgoing Systems 9–13
9.3.3 Configuring Incoming Systems 9–13
9.3.4 Configuring the Poll File 9–15
9.4 Monitoring the File Transfer Queue 9–15
9.4.1 Getting Queue Status Manually 9–15
9.4.2 Getting Queue Status Automatically 9–16
9.4.3 Guidelines for Checking Queue Status 9–16
9.5 Cleaning Up the Spooling Directories 9–17
9.5.1 Cleaning Up Directories Manually 9–17
9.5.2 Cleaning Up Directories Automatically 9–18
9.5.3 Guidelines for Removing Files 9–19
9.6 Viewing Log Files 9–20
9.7 Cleaning Up sulog and cron/log Files 9–21
9.8 Limiting the Number of Remote Executions 9–21
9.9 Scheduling Work in the Spooling Directory 9–22
9.9.1 Starting uusched Manually 9–22
9.9.2 Starting uusched Automatically 9–22

9.10 Calling File Transfer Programs (uudemon.hour) 9–23
9.11 Polling Remote Systems (uudemon.poll) 9–24

10 Network Time Protocol

10.1 NTP Environment 10–2
10.2 NTP Planning 10–3
10.2.1 Server Information 10–4
10.2.2 Client Information 10–5
10.3 Configuring NTP 10–5
10.4 Enabling the High-Resolution Clock 10–8
10.5 Monitoring Hosts Running the xntpd Daemon 10–8
10.6 Monitoring Hosts Running the ntpd Daemon 10–10
10.7 Querying Servers Running NTP 10–10

11 Mail System

11.1 Mail Environment 11–1
11.1.1 Directing Outgoing Mail to Servers 11–4
11.1.2 Handling Incoming Mail to the Domain 11–4
11.1.3 Delivering Mail to Clients 11–5
11.1.4 Distributing the aliases File 11–5
11.1.5 Distributing the passwd File 11–5
11.1.6 Handling DECnet Mail 11–5
11.2 Planning Mail 11–7
11.2.1 Verifying that Required Protocols are Installed 11–7
11.2.2 Verifying that Required Services are Configured 11–7
11.2.3 Preparing for the Configuration 11–8
11.2.3.1 General System Information 11–8
11.2.3.2 Protocol Information 11–9
11.3 Configuring Mail 11–12
11.3.1 Configuring a Standalone Mail System 11–13
11.3.2 Configuring a Mail Client 11–13
11.3.3 Configuring a Mail Server 11–14
11.4 Mail Utilities 11–16
11.5 Monitoring the Mail Queue 11–17
11.6 Archiving the Mail Queue 11–17
11.7 Administering and Distributing Alias Information 11–18
11.8 Display Mail Statistics 11–19

11.9 Adding a New Mail Host 11–19

12 Simple Network Management Protocol Agent

12.1 SNMP Environment 12–1

Part 2 Problem Solving Information

13 Solving Network and Network Services Problems

13.1 Using the Diagnostic Map 13–1
13.2 Getting Started 13–2
13.3 Solving Network Problems 13–3
13.4 Solving DNS/BIND Server Problems 13–7
13.5 Solving DNS/BIND Client Problems 13–9
13.6 Solving NIS Server Problems 13–10
13.7 Solving NIS Client Problems 13–14
13.8 Solving NFS Server Problems 13–18
13.9 Solving NFS Client Problems 13–21
13.10 Solving UUCP Problems 13–24
13.11 Solving NTP Problems 13–26
13.12 Solving SLIP Problems 13–29
13.13 Solving PPP Problems 13–31
13.14 Solving LAT Problems 13–33
13.15 Solving sendmail Problems 13–40

14 Using the Problem Solving Tools

14.1 Testing Reachability of Network Hosts on the Internet Network 14–1
14.2 Displaying and Modifying the Internet to Ethernet Translation Tables 14–2
14.3 Displaying a Datagrams's Route to a Network Host 14–3
14.4 Displaying Headers of Packets on the Network 14–6
14.5 Testing a UUCP Remote Connection 14–6
14.6 Monitoring a File Transfer 14–8
14.7 Viewing the Error Log File 14–8
14.8 Viewing the syslogd Daemon Message Files 14–9

15 Testing DNS Servers

15.1 Glossary 15–1
15.2 Starting the DNS Server Test 15–2

15.3 Determining the Server Type 15–3
15.4 Finding the Target Domain Information 15–6
15.5 Testing the Forwarders 15–8
15.6 Testing Secondary Authoritative Servers 15–9
15.7 Testing Primary Authoritative Servers 15–13
15.8 Tracing Information from the Root Name Server 15–16
15.9 Resolving Target Data 15–18
15.10 Finding the First Nonexistent Domain 15–20

16 Reporting Network Problems

16.1 Gathering Information 16–1
16.1.1 General Information 16–1
16.1.2 Hardware Architecture 16–2
16.1.3 Software Architecture 16–2

A Configuration Worksheet

B Monitoring the Network Interfaces

B.1 Monitoring the Ethernet Interface B–1
B.2 Monitoring the FDDI Interface B–4
B.2.1 FDDI Counters B–6
B.2.2 FDDI Status B–9
B.2.3 FDDI Characteristics B–18
B.3 Monitoring the Token Ring Interface B–20
B.3.1 Token Ring Counters B–21
B.3.2 Token Ring and Host Information B–24

C Writing automount Maps

C.1 Substitution and Pattern Matching C–5
C.2 Environment Variables C–6
C.3 Mounting File Systems C–7
C.3.1 Multiple Mounts C–7
C.3.2 Shared Mounts C–8
C.3.3 Replicated File Systems C–8

D NIS ypservers Update Scripts

D.1 Add Slave Server Script D–1

D.2 Remove Slave Server Script D–2

E NFS Error Messages

E.1 Server Error Messages E–1
E.2 Client Error Messages E–2
E.2.1 Remote Mount Error Messages E–3
E.2.2 automount Error Messages E–6
E.2.3 Console Error Messages E–10

F uucp Messages

F.1 Status and Log File Messages F–1
F.2 tip Error Messages F–8

G sendmail Error Messages

H Host Resources MIB Implementation

H.1 Tru64 UNIX Implementation Summary H–1
H.2 System Group H–1
H.3 Storage Group H–2
H.4 Device Tables H–3
H.5 File System Table H–8
H.6 The Running Software Tables H–9

I DNS Server Worksheet

J Format of DNS Data File Entries

J.1 Format of DNS Resource Records J–1
J.2 Description of Data File Entries J–3
J.2.1 Include Entry J–3
J.2.2 Origin Entry J–3
J.2.3 Address Entry J–4
J.2.4 Canonical Name Entry J–4
J.2.5 Host Information Entry J–5
J.2.6 Mailbox Entry J–6
J.2.7 Mail Group Entry J–7
J.2.8 Mailbox Information Entry J–7
J.2.9 Mail Rename Entry J–8
J.2.10 Mail Exchanger Entry J–9
J.2.11 Name Server Entry J–9

J.2.12 Domain Name Pointer Entry J–10
J.2.13 Start of Authority Entry J–10
J.2.14 Well Known Services Entry J–12

Index

Examples

3–1 Sample DHCP Log File Message 3–20
5–1 Sample /etc/latstartup.conf File 5–8
6–1 Sample /etc/hosts File 6–10
C–1 Multiple Mounts in a Direct Map C–2
C–2 Multiple Mounts and Shared Mounts in a Direct Map C–3
C–3 Multiple Mounts, Shared Mounts, and Replicated File Systems in a Direct Map C–3
C–4 Simple Indirect Map C–3
C–5 Multiple Mounts in an Indirect Map C–4
C–6 Multiple Mounts and Shared Mounts in an Indirect Map C–4
C–7 Multiple Mounts, Shared Mounts, and Replicated File Systems in an Indirect Map C–4

Figures

2–1 Sample Network Configuration 2–2
2–2 Configuration Worksheet, Part 1A 2–3
2–3 Configuration Worksheet, Part 1B 2–9
3–1 DHCP Configuration (acme-net) 3–2
3–2 Configuration Worksheet, Part 2A 3–5
3–3 Configuration Worksheet, Part 2B 3–9
4–1 Sample Simple SLIP Configuration 4–2
4–2 SLIP Configuration with Gateway System 4–2
4–3 Configuration Worksheet, Part 3A 4–4
4–4 Simple PPP Configurations 4–12
4–5 Network PPP Configuration 4–13
4–6 Configuration Worksheet, Part 3B 4–16
5–1 Sample LAT Network Configuration 5–2
5–2 Configuration Worksheet, Part 4 5–5
6–1 Sample Small DNS Configuration 6–3
6–2 Sample Large DNS Configuration 6–4
6–3 Configuration Worksheet, Part 5 6–5
7–1 NIS Configuration 7–2
7–2 Configuration Worksheet, Part 6 7–3

8–1	Configuration Worksheet, Part 7	8–3
9–1	Sample Simple UUCP Configuration	9–2
9–2	Sample UUCP Over TCP/IP Configuration	9–2
9–3	Configuration Worksheet, Part 8A	9–4
9–4	Configuration Worksheet, Part 8B	9–6
9–5	Configuration Worksheet, Part 8C	9–9
10–1	Sample NTP Configuration (Local Clock)	10–2
10–2	Sample NTP Configuration (Internet Source)	10–3
10–3	Configuration Worksheet, Part 9	10–4
11–1	Sample Mail Standalone Configuration	11–2
11–2	Sample Mail Client/Server Configuration	11–3
11–3	Configuration Worksheet, Part 10A	11–8
11–4	Configuration Worksheet, Part 10B	11–10
A–1	Configuration Worksheet, Part 1A	A–2
A–2	Configuration Worksheet, Parts 1B and 2A	A–3
A–3	Configuration Worksheet, Part 2B	A–4
A–4	Configuration Worksheet, Parts 3A and 3B	A–5
A–5	Configuration Worksheet, Parts 4 and 5	A–6
A–6	Configuration Worksheet, Part 6	A–7
A–7	Configuration Worksheet, Part 7	A–8
A–8	Configuration Worksheet, Parts 8A and 8B	A–9
A–9	Configuration Worksheet, Parts 8C and 9	A–10
A–10	Configuration Worksheet, Parts 10A and 10B	A–11
C–1	Sample automount Maps	C–2

Tables

2–1	Options to the netstat Command	2–21
2–2	Options to the fddi_config Command	2–22
2–3	Options to the srconfig Command	2–24
4–1	Types of Null Modem Cable	4–3
4–2	Basic startslip Subcommands	4–6
4–3	Optional startslip Subcommands	4–6
4–4	slhosts File Options	4–7
4–5	Modem Commands for Dial-Out Access	4–9
4–6	Types of Modem Cable	4–24
4–7	Modem Commands for Dial-In Access	4–25
5–1	LAT Parameters	5–8
7–1	NIS Map Information Commands	7–30
9–1	Additional uucpsetup Commands	9–12
10–1	Options to the ntpq Command	10–9
10–2	Options to the xntpdc Command	10–10

13–1	Problem Solving Starting Points	13–2
14–1	Options to the ping Command	14–1
14–2	Options to the traceroute Command	14–4
F–1	ASSERT Error Messages	F–2

About This Manual

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

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

Audience

This manual is intended for system and network adminstrators responsible for configuring and managing network services. Administrators should have knowledge of operating system concepts, commands, and configuration. They should also 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

This manual has been revised to include the following:

- A section on configuring network adapters for automatic failover by using a Redundant Array of Independent Network Adapters (NetRAIN).
- An updated chapter that describes how to configure the Dynamic Host Configuration Protocol (DHCP).
- An appendix describing the format of DNS data file entries.
- Several sections that were previously included in the *Release Notes*.

Organization

This manual is divided into 16 chapters, 10 appendixes, and an index. The following list describes the contents in more detail:

Chapter 1	Describes the meaning of network administration and the components covered in this manual.
Chapter 2	Describes the tasks to administer the basic network connections.
Chapter 3	Describes the tasks to administer the Dynamic Host Configuration Protocol (DHCP).
Chapter 4	Describes the tasks to administer point-to-point connections.
Chapter 5	Describes the tasks to administer Local Area Transport (LAT).
Chapter 6	Describes the tasks to administer the Domain Name Service (DNS).
Chapter 7	Describes the tasks to administer the Network Information Service (NIS).
Chapter 8	Describes the tasks to administer the Network File System (NFS).
Chapter 9	Describes the tasks to administer the UNIX-to-UNIX Copy Program (UUCP).
Chapter 10	Describes the tasks to administer the Network Time Protocol (NTP).
Chapter 11	Describes the tasks to administer the mail environment.
Chapter 12	Describes the Simple Network Management (SNMP) environment.
Chapter 13	Describes how to solve network and network service problems.
Chapter 14	Describes the various diagnostic tests available to help solve problems.
Chapter 15	Describes how to test DNS servers and resolve DNS server problems.
Chapter 16	Describes how to report your problem to Compaq and the information you should provide.
Appendix A	Contains the Configuration Worksheet. Copy this appendix and fill in the information before completing the tasks described in this manual.
Appendix B	Describes how to monitor the Ethernet, Fiber Distributed Data Interface (FDDI), and token ring network interfaces by using the `netstat` command.
Appendix C	Describes how to write `automount` maps.
Appendix D	Contains two scripts you can copy for adding NIS slave servers to and removing NIS slave servers from an NIS domain.
Appendix E	Contains NFS client error messages and describes possible solutions.
Appendix F	Contains `uucp` error messages and describes possible solutions.
Appendix G	Contains `sendmail` error messages and describes possible solutions.
Appendix H	Describes the Tru64 UNIX host MIB implementation, including sample data.

Appendix I Contains a worksheet for recording information as you solve DNS server problems.

Appendix J Describes the format of DNS file entries.

Related Documents

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

- *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 document can be accessed by opening the following file with a web browser:

  `/usr/doc/join/TOC.html`

- *Sendmail Installation and Operation Guide*

  Provides additional information about using the `sendmail` command. This HTML document is available at the following URL:

  `http://uwsg.ucs.indiana.edu/usail/mail/op/op.html`

- The *sendmail* guide by O'Reilly & Associates

  Provides additional information about using the `sendmail` command.

The printed version of the Tru64 UNIX documentation set is color coded to help specific audiences quickly find the books that meet their needs. (You can order the printed documentation from Compaq.) This color coding is reinforced with the use of an icon on the spines of books. The following list describes this convention:

Audience	Icon	Color Code
General users	G	Blue
System and network administrators	S	Red
Programmers	P	Purple
Device driver writers	D	Orange
Reference page users	R	Green

Some books in the documentation set help meet the needs of several audiences. For example, the information in some system books 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 books 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`

- Mail:

  Compaq Computer Corporation
  UBPG Publications Manager
  ZKO3-3/Y32
  110 Spit Brook Road
  Nashua, NH 03062-9987

  A Reader's Comment form is located in the back of each printed manual. The form is postage paid if you mail it in the United States.

Please include the following information along with your comments:

- The full title of the book and the order number. (The order number is printed on the title page of this book and on its back cover.)
- 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).

Part 1

Management Information

1

Overview of Network Administration

Network administration comprises those tasks that deal with the setting up and configuration of 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 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
- Dynamic Host Configuration Protocol (DHCP)
- Point-to-point connections, including Serial Line Internet Protocol (SLIP) and Point-to-Point Protocol (PPP)
- Local Area Transport (LAT)
- Domain Name Service (DNS)
- Network Information Service (NIS), formerly named Yellow Pages
- Network File System (NFS)
- UNIX-to-UNIX Copy Program (UUCP)
- Network Time Protocol (NTP)
- Mail environment
- Simple Network Management Protocol (SNMP)

Day-to-day management varies with each network service, as each one provides different capabilities. Typically, management involves making small changes and adjustments, such as adding users, mounting remote file systems or directories, obtaining status information, and setting up automatic maintenance scripts. Each chapter in Part 1 of this book describes a specific task, presenting the generic steps required to perform the task followed by examples and additional information.

In addition to the day-to-day management of the network and network services, this manual contains information to help you solve problems that might occur. Problem solving is handled as a separate part of 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.

2

Basic Network Connections

This chapter describes the basic Tru64 UNIX network environment, including how to configure:

- Ethernet
- Token Ring
- Fiber Distributed Data Interfaces (FDDI)
- Automatic network adapter failover (NetRAIN)
- Various network daemons in order to operate in a TCP/IP network environment

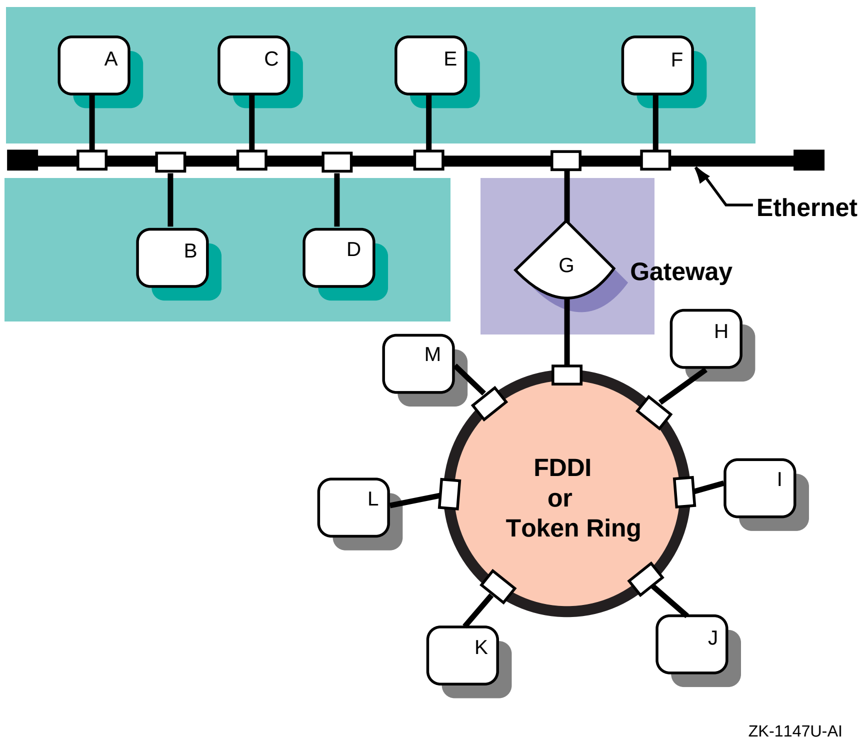
In addition, this chapter describes some of the commands you use to monitor the network environment.

For information about point-to-point connections, see Chapter 4.

2.1 Network Environment

Figure 2–1 shows a sample corporate network in which there is an Ethernet backbone and an FDDI or Token Ring network connected to it through a gateway.

Figure 2–1: Sample Network Configuration



2.2 Preparing for the Configuration

You configure the network components by using the Network Configuration application. Appendix A contains a worksheet that you can use to record the information that you need to configure the network components.

2.2.1 Information for Interfaces and Daemons

Figure 2–2 shows Part 1A of the Configuration Worksheet. The following sections explain the information you need to record in Part 1A of the worksheet.

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

Figure 2–2: Configuration Worksheet, Part 1A

Part 1A: Interface and Daemon Information

All Network Interfaces

Adapter name: ______________ ______________

Host name: ______________ ______________

IP address source: DHCP server ☐ User supplied ☐

Internet address: ______________ ______________

Network mask: ______________ ______________

Token Ring interface

Adapter speed: ______________

NetRAIN interface

Set members: ______________ ______________

rwhod daemon

rwhod: Yes ☐ No ☐

Flags: Broadcast only ☐ Listen only ☐ Both ☐

routed daemon

routed: Yes ☐ No ☐

Flags: ☐ Run routed on gateway host
☐ Write all packets to standard output
☐ Log additional information

RIP data: Supply ☐ Run quietly ☐

Gateways file

Destination type: Net ☐ Host ☐

Destination: ______________

Gateway: ______________

Hop count: ______________

Route type: External ☐ Passive ☐ Active ☐

gated daemon

gated: Yes ☐ No ☐

Configuration file: ______________

IP router

IP router: Yes ☐ No ☐

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 are supported on Tru64 UNIX:

Interface	Device Name

Ethernet	`le`
	`ln`
	`tu`
	`xna`

Fiber Distributed Data Interface (FDDI)	`faa`
	`fta`
	`fza`

Token Ring	`tra`

Note that if you configuring a NetRAIN interface, as documented in Section 2.4, the adapter name is the virtual device name of your NetRAIN set (`nr`).

Host name

The fully qualified host name assigned to your system or NetRAIN interface. 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 Internet Protocol (IP) address of your system or NetRAIN interface. You should have obtained an IP address for your network from InterNIC Registration Services. After you receive your network's address, you must 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
Email: 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: hostmaster@apnic.net (for general information and IP address registrations)
WWW: http://www.apnic.net

Note

You should 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 local area network (LAN) to 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. *n.n.n*
B	128.0.0.0 to 191.0.0.0	255.255. *n.n*
C	192.0.0.0 to 223.0.0.0	255.255.255. *n*

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

Set members

The device names of the network interfaces that are part of the NetRAIN set, as discussed in Section 2.4. When one interface ceases to function, NetRAIN will fail over to another interface on this list.

2.2.1.4 rwhod Daemon

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.

2.2.1.5 routed daemon

Running 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.

Note

You can choose the `routed` daemon or `gated` daemon, but not both.

Flags

Specifies how you want `routed` to run. You can run `routed` 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 `routed` is to supply RIP information, check Supply; otherwise, check Run Quietly.

2.2.1.6 Gateways 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 address of the gateway host to which messages should 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.

2.2.1.7 gated Daemon

Running the `gated` daemon allows your system's internal routing tables for different routing protocols to be updated automatically.

gated

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

Note

You can choose the `routed` daemon or `gated` daemon, but not both.

Configuration file

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

2.2.1.8 IP Router

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 `routed` or `gated`.

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 2–3 shows Part 1B of the Configuration Worksheet. The following sections explain the information you need to record in Part 1B of the worksheet. If you are viewing this manual online, you can use the print feature to print a copy of this part of the worksheet.

Figure 2–3: Configuration Worksheet, Part 1B

Part 1B: Network Files Information

Static routes file

Destination type: Default gateway ☐ Host ☐ Network ☐

Destination: ____________

Route via: Gateway ☐ Interface ☐

Gateway: ____________

Hosts file

Host name: ____________ ____________ ____________

Internet address: ____________ ____________ ____________

Alias: ____________ ____________ ____________

hosts.equiv file

Host name: ________ ________ ________ ________

User name: ________ ________ ________ ________

Networks file

Network name: ____________ ____________ ____________

Network address: ____________ ____________ ____________

Alias: ____________ ____________ ____________

2.2.2.1 Static Routes File (/etc/routes)

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.

2.2.2.2 Host File (/etc/hosts)

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.

2.2.2.3 hosts.equiv 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.

2.2.2.4 Networks File (/etc/networks)

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.

2.3 Configuring the Network Components

Use the Network Configuration application of the Common Desktop Environment (CDE) Application Manager to configure network components on systems with graphics capabilities. You can configure the following components:

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

See `netconfig`(8X) for more information on the Network Configuration application.

To invoke the Network Configuration application, log in as root and do the following:

1. Click on the Application Manager icon on the CDE front panel.
2. Double-click on the System_Admin application group icon.
3. Double-click on the Configuration application group icon.
4. Double-click on the Network Configuration application icon. The Network Configuration main window is displayed, showing available network components and configured network components.

To exit the Network Configuration application, choose File then Exit.

Note

For systems without graphics capabilities, you can use the `netsetup` utility. See `netsetup`(8) for more information.

The Network Configuration application has an extensive online help system that you can use, instead of the instructions in the following section, to configure network components on your system.

2.3.1 Configuring Network Interfaces

Use the following procedure to configure the Ethernet, FDDI, or Token Ring network interface. For information about how to configure NetRAIN, see Section 2.4.

1. Select an interface from the Available Network Components list box in the Network Configuration main window. All network adapters that are installed on the system are listed in the list box.
2. Click on the Configure button to display the Configuring Interface dialog box for the selected interface.
3. Set the Interface Configuration Enable check button to the On position to enable the interface.
4. Enter the name of the interface in the Host Name field. If this is the first or only network interface you are configuring and there is a default host name, the utility displays the default.
5. For the Ethernet interface, do the following:

    a. To obtain the IP address data from the DHCP server, set the Use DHCP Server radio button to the On position. Otherwise, set the User Supplied Value radio button to the On position and enter the IP address and network mask data in the appropriate fields.

    b. Click on the down arrow button to display the advanced configuration parameters for the selected interface.

    c. Click on the Multicast check button to enable the reception of multicast packets. See `map-mbone`(7) for more information on multicast routing.

    d. Click on the check boxes for the appropriate interface options.

    e. Go to step 8.

6. For the FDDI interface, do the following:

    a. To obtain the IP address data from the DHCP server, set the Use DHCP Server radio button to the On position. Otherwise, set the User Supplied Value radio button to the On position and enter the IP address and network mask data in the appropriate fields.

    b. Click on the down arrow button to display the advanced configuration parameters for the selected interface.

    c. Click on the check boxes for the appropriate interface options.

    d. Enter the broadcast address for the interface in the Broadcast Address field.

e. Go to step 8.

7. For the 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. Click on the button in option box to display a menu of token ring adapter speeds. Select the appropriate adapter speed: 4 or 16.
   d. Click on the down arrow button to display the advanced configuration parameters for the selected interface.
   e. Enter the broadcast address for the interface in the Broadcast Address field.
   f. Go to step 8.

8. Click on Commit.
9. Click on Yes to save the changes and start the interface.
10. Click on Close to close the Configuring Interface dialog box.

You can also use the Network Configuration application to modify and disable network interfaces. See the online help for more information.

2.3.2 Configuring the rwhod Daemon

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

1. Select the Rwho Daemon from the Available Network Components list box in the Network Configuration main window.
2. Click on the Configure button to display the Configuring Rwho Daemon dialog box.
3. Set the rwhod Daemon Enable check box to the On position.
4. Click on the appropriate `rwhod` flag.
5. Click on Commit.
6. Click on Yes to save the changes and start the daemon.
7. Click on Close to close the Configuring rwho Daemon dialog box.

You can also use the Network Configuration application to modify and 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. Select the Route Daemon from the Available Network Components list box in the Network Configuration main window.
2. Click on the Configure button to display the Configuring Route Daemon dialog box.
3. Set the routed Daemon Enable check box to the On position.
4. Set the `routed` flags to the On position as needed.
5. Set the Supply RIP Data radio button to the On position if the `routed` daemon will run on a gateway host and supply Routing Information Protocol (RIP) data. Set the Run Quietly radio button to the On position if the `routed` daemon will not supply RIP information.
6. Click on Gateways File. The Gateways dialog box is displayed. Do the following:

   a. In the Destination Type field, set the net radio button to the On position if if the destination is a network. Set the host radio button to the On position if the destination is a host.

   b. Enter the destination name, IP address, or **`default`** in the Destination field.

   c. Enter the name or IP address of the gateway host in the Gateway field.

   d. Enter the hop count in the Hop Count field.

   e. Set the appropriate Route Type radio button to the On position.

   f. Click on Add. Repeat steps a through f for additional gateways.

   g. Click on Commit and Close to save the changes and close the Gateways dialog box.

7. Click on Commit.
8. Click on Yes to save the changes and start the routed daemon.
9. Click on Close to close the Configuring Route Daemon dialog box.

You can also the Network Configuration application to modify and disable the `routed` daemon and entries in the `gateways` file. 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. Select the Gate Daemon from the Available Network Components list box in the Network Configuration main window.
2. Click on the Configure button to display the Configuring Gate Daemon dialog box.
3. Set the gated Daemons Enable check box to the On position.
4. Enter the 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. Click on Commit.
6. Click on Yes to save the changes and start the daemon.
7. Click on Close to close the Configuring Gate Daemon dialog box.

You can also use the Network Configuration application to modify and 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 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. Select IP Router from the Available Network Components list box in the Network Configuration main window.
2. Click on the Configure button to display the Configuring IP Router dialog box.
3. Set the IP Router Enable check box to the On position.
4. Click on Commit.

5. Click on Yes to start the IP Router.
6. Click on Close to close the Configuring IP Router dialog box.

You can also use the Network Configuration application 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. Select the Static Routes File from the Available Network Components list box in the Network Configuration main window.
2. Click on the Configure button to display the Static Routes dialog box.
3. Set the appropriate Destination Type radio button to the On position.
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. Set one of the Route Via radio buttons to the On position. Click on the Gateway button if the route is through a gateway; click on the Interface button 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. Click on Add to accept the entry. Repeat steps 3 through 6 for additional static routes.
7. Click on Commit and click on Close to save the current changes and close the dialog box.

You can also use the Network Configuration application 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. Select the Host File from the Available Network Components list box in the Network Configuration main window.
2. Click on the Configure button to display the Hosts dialog box.
3. Enter the official host name in the Host Name field.

4. Enter the IP address of the new host in the Host Address field.
5. If an unofficial name or names (aliases) are assigned to the new host, enter the names in the Aliases field.
6. Click on Add to accept the entry. Repeat steps 3 through 6 for additional hosts.
7. Click on Commit and Close to update the `hosts` file and close the Hosts dialog box.

You can also use the Network Configuration application 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. Select the Host.equiv File from the Available Network Components list box in the Network Configuration main window.
2. Click on the Configure button to display the Hosts.equiv dialog box.
3. Enter the host name in the Host name field.

> **Note**
>
> You cannot add a host that is not on the network.

4. Enter the name of a user on the remote host.
5. Click on Add to accept the entry. Repeat steps 3 through 5 for additional hosts.
6. Click on Commit and Close to update the `/etc/hosts.equiv` file and close the Hosts.equiv dialog box.

You can use the Network Configuration application to modify and delete entries in the `hosts.equiv` file. See the online help for more 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. Select the Network File from the Available Network Components or Configured Network Components list box in the Network Configuration main window.

2. Click on the Configure button to display the Networks dialog box.
3. Enter the official network name in the Net Name field.
4. Enter the IP address of the network in the Net Address field.
5. If an unofficial name or names (aliases) are assigned to the new network, enter the aliases in the Aliases field.
6. Click on Add to accept the entry. Repeat steps 3 through 6 for additional networks.
7. Click on Commit and Close to update the `/etc/networks` file and close the Networks dialog box.

You can also use the Network Configuration application 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.

2.4 NetRAIN Interfaces

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

NetRAIN integrates multiple network interfaces on the same LAN segment into a single virtual interface called a NetRAIN set. One network interfaces 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.

2.4.1 Configuring NetRAIN

This section describes how to configure the hardware and the network interfaces for a NetRAIN set.

2.4.1.1 Hardware Restrictions and Configuration

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, that is, the interfaces cannot be marked as "up" in the Network Configuration application and cannot be configured with any upper layer protocol (for example, IP, LAT, or DECnet/OSI).
- 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 should run separate cables from each network adapter 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.
- You might need to adjust the timeout values to ensure that NetRAIN will detect and respond to network failure successfully. You can tune these parameters with the `sysconfig` command, the `ifconfig` command, or the `ioctl` system call. See the `nr`(7), `ifconfig`(8), `sysconfig`(8), `dxkerneltuner`(8), and `sys_attrs_netrain`(5) reference pages for details.

  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 might 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.

In addition, 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
This is the permanent address that is taken from the small PROM that each LAN adapter contains.

Current physical address
This is the address that the network responds to on the wire.

Licensing schemes based on MAC addresses should 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.

2.4.1.2 Configuring the NetRAIN Interface

To configure a NetRAIN interface, log in as root and enter the following `ifconfig` command:

```
# ifconfig netrain-interface-id IP-address netmask network-mask add interface-
id,interface-id
```

For example, to create a NetRAIN set called `nr0` with two FDDI interfaces called `fta0` and `fta1`, you might enter the following command:

```
# ifconfig nr0 18.240.32.40 netmask 255.255.255.0 add fta0,fta1
```

Or, to create a NetRAIN set called `nr1` with two ATM LANE interfaces, you might enter this command:

```
# ifconfig nr1 18.240.32.41 netmask 255.255.255.0 add elan0,elan1
```

After you create the NetRAIN set, use the `ifconfig` command as documented in Section 2.4.2 to determine if the NetRAIN interface is running.

To have the NetRAIN interface start each time you boot your system, add the `ifconfig` command and parameters as shown in the previous examples to the `inet.local` file. See `inet.local`(8) for more information.

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 ffffff00 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 currently using `fta0` for communication. If NetRAIN determines that `fta0` is not active, it will switch to the next interface in the set, `fta1`.

NetRAIN activity is also logged in the `kern.log` file.

2.5 Enabling Access Filtering on an Interface

Interface access filtering provides a mechanism for detecting and preventing 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 Monitoring the Local Host's Status

You use the netstat command to monitor the status of the local host by viewing the contents of network-related data structures. You can select several forms of display; each allows you to specify the type of information you want to emphasize.

To monitor the local host's network status, use the netstat command with the following syntax:

netstat [*options...*]

Table 2–1 shows the netstat command options.

Table 2–1: Options to the netstat Command

Option	Function
–A	Displays the address of any associated protocol control blocks.
–a	Includes information for all sockets.
–f *address_family*	Includes statistics or address control block reports for the specified address family.
–I *interface*	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 B 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.

See netstat(1) for more information on this command.

The following example shows normal output (no `Ierrs` or `Oerrs`) 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
```

2.7 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, do the following:

1. Log in as root.
2. Use the `fddi_config` command and options as follows:

   fddi_config [*options*...]

Table 2–2 shows the `fddi_config` command options.

Table 2–2: Options to the fddi_config Command

Option	Function
–i *interface_name*	Changes or displays the FDDI characteristics for *interface_name*. You must provide the interface name.
–c *counter_update_interval*	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.)
–d	Displays the FDDI interface parameters you can set.

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

Option	Function
–l *lem_threshold*	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}.
–p [1|0]	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.
–r *restricted_token_timeout*	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.
–t *token_request_time*	Sets the Request Token Rotation Time (T_req) for *interface_name*. 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.
–v *valid_transmit_time*	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.
–x [1|0]	Enables (1) or disables (0) full-duplex operation for the interface. If the full-duplex operation is enabled, the interface can be in one of the following states: Idle, Request, Confirm, or Operational. (For the DEFTA (fta) FDDI interface only.)

See `fddi_config`(1) for more information on this command.

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
```

2.8 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 and options as follows:

srconfig *options...*

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` *mac_address*[a]	Deletes a source routing table entry.
`–DISEntry` *mac_address*[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` *mac_address*	Displays a specific source routing table entry.
`–RTable`	Displays the source routing table.

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

Option	Function
`–SETAgetimer` *timer*[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.
`–SETDsctimer` *timer*[a]	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.
`–SETMaxentry` *value*[a]	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.
`–u`	Specifies that the MAC addresses are in uncanonical form. This option can be used with the `–DElEntry` *mac_address*, `–DISEntry` *mac_address*, and `–RTable` options only.
`–Zcounter`	Sets the source routing counters to zero.

[a]Requires superuser privileges.

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

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 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 is included for formatting purposes 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 displays 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 is included for formatting purposes 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)
```

2.9 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 ffffff00 broadcast 16.141.208.255 ipmtu 4092
```

The following command changes the IP MTU size of DETRA interface to 2044 bytes:

```
% ifconfig tra0 ipmtu 2044
```

3

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, and provides basic information for setting up and maintaining the DHCP database. In addition, this chapter provides information on the `xjoin` application, and the different DHCP configurations.

The implementation of DHCP in Tru64 UNIX is based on the 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`

Note

Starting with Tru64 UNIX Version 4.0F, DHCP database files are 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.

3.1 DHCP Environment

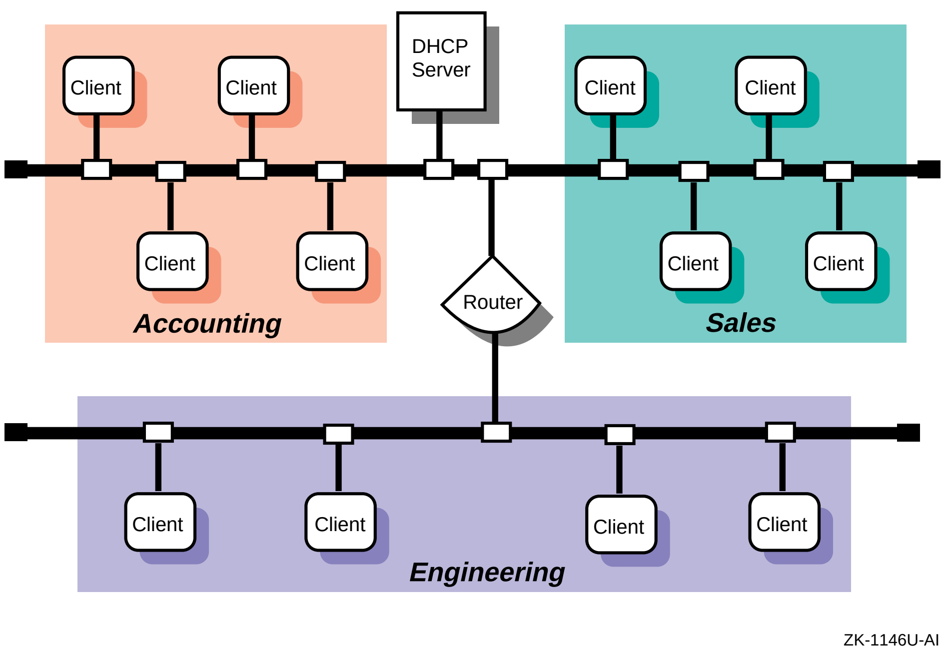
In the DHCP environment, systems can have the following roles:

- Server — A system that offers DHCP and BOOTP services to other systems on the network. There can be one DHCP server on a subnetwork. Multiple servers can exist on a subnetwork, but each server's IP address range cannot overlap.

- Client — A system that requests configuration information from a DHCP server.

Figure 3–1 shows a sample corporate LAN 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 the `bprelay`(8) reference page for more information.

Figure 3–1: DHCP Configuration (acme-net)



3.1.1 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 3–1 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 3–1 showed a single LAN network 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.

3.1.2 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 3.8 for more information.

3.2 Planning DHCP

This section describes the tasks you need to do before configuring DHCP.

3.2.1 Verifying Installation of DHCP Software

For a DHCP server system, verify that the DHCP server is installed by entering the following command:

```
# setld -i | grep OSFINET440
```

If the subset is not installed, install it by using the `setld` command. For more information on installing subsets, see the `setld`(8) reference page, the *Installation Guide*, or the *System Administration* manual.

For DHCP client systems, the DHCP client software is installed with the mandatory subsets.

3.2.2 Preparing for the Configuration

After you verify that the DHCP software is installed, you 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. Appendix A contains a worksheet that you can use to record the information required to configure DHCP.

3.2.2.1 Server/Security Parameters

Figure 3–2 shows Part 2A of the Configuration Worksheet.

Figure 3–2: Configuration Worksheet, Part 2A

Part 2A: DHCP Server/Security Parameters

BOOTP address from pool: True ☐ False ☐
BOOTP compatibility: True ☐ False ☐
Default lease time: ________
Name service: `/etc/hosts` ☐ DNS ☐ NIS ☐
Ping timeout: ________
Provisional time to live: ________
Restrict to MAC address: True ☐ False ☐

IP ranges

Subnetwork address: ________
DHCP server: ________
IP ranges: ________ ________
________ ________

Host name Lists

Domain name: ________
DHCP server: ________
Host name prefix: ________
Host names: ________ ________
________ ________

If you are viewing this manual online, you can use the print feature to print part of the worksheet. The following sections explain the information you need to record in Part 2A of the worksheet.

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 3.10.

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:

- The Domain Name Service (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.
- A Local Name Service updates the `/etc/hosts` file with information about dynamically assigned names and addresses.

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 addresses

If you want to assign an IP address to a client's matching MAC address, check True; otherwise, check False. See Section 3.8 for additional information on restricting client access to the server.

3.2.2.1.1 IP Ranges

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 the preceding 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 that will be assigned to clients on the selected subnetwork. Using the preceding 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 may 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 3.2.2.2 and the `bprelay`(8) reference page for information about boot file and BOOTP parameters.

3.2.2.1.2 Host name list

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`.

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.

3.2.2.2 Information for Basic DHCP Parameters

Figure 3–3 shows Part 2B of the Configuration Worksheet.

Figure 3–3: Configuration Worksheet, Part 2B

Part 2B: Basic DHCP Parameters

Configuration type: Node ☐ Subnet ☐ Group ☐
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 ☐ False ☐
Subnetwork mask: ____________
TFTP root directory: ____________
Broadcast address: ____________
Subnetworks are local: True ☐ False ☐
Supply masks: True ☐ False ☐
DHCP rebinding time: ____________
DHCP renewal time: ____________
Lease time: ____________

If you are viewing this manual online, you can use the print feature to print part of the worksheet. The following sections explain the information you need to record in Part 2B of the worksheet.

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 should use when resolving host names using the Domain Name Service.

DNS server IP addresses

A list of IP addresses of DNS (STD 13, RFC 1035) 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 (BOOTP)

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 as per RFC 950. 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. The client should assume that some subnetworks of the directly connected network may have smaller MTUs.

Supply masks

If the client should respond 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. 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.

3.3 Configuring a DHCP Server

Use the `xjoin` application to configure a DHCP server. To start the application, 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

To update the server so that the new configuration takes effect, click on the Add/Update button in the lower right-hand side of the window. To exit the application, select File and Exit from the menu bar. See the `xjoin`(8) reference page and the *JOIN Server Administrator's Guide* (`/usr/doc/join/TOC.html`) for more information.

3.3.1 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.

3.3.2 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 Tab 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.

3.3.3 Configuring Host Name Lists

You configure host name lists only if the Accept Client Name server parameter is set to False. (See Section 3.2.2.1.) 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.

3.3.4 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.
4. Enter the name of the subnetwork configuration, for example, Subnet3.
5. Select the Member of Group parameter. Enter the name of the group of which the subnetwork will be a member.
6. Select the Net or Subnet IP Address parameter. Enter the Net or Subnet IP address that identifies the subnetwork portion of the network.
7. Select the Broadcast Address parameter. Enter the broadcast address for this subnetwork.

8. Enter information for basic DHCP parameters in the appropriate fields. See Section 3.2.2 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.

9. Click on the Add/Update button to update the server with new subnetwork configuration information.

10. 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

3.3.5 Configuring a DHCP Client Node

To configure a node, do the following:

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.
4. Enter the name of the node configuration; for example, Client5.
5. 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.
6. 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.

7. Select the Member of Group parameter. Enter the name of the group of which the node will be a member.
8. Enter information for basic DHCP parameters. See Section 3.2.2 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.

9. Click on the Add/Update button to update the server with new node configuration information.

3.3.6 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.
4. Enter the name of the group configuration; for example, Global.
5. Select the Member of Group parameter. If appropriate, enter the name of the group of which that the new group will be a member.
6. Select the Group Members parameter. Enter the names of subnetworks, nodes, or other groups that will be members of the group. Press Tab between entries.
7. Enter information for basic DHCP parameters. See Section 3.2.2 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.
8. Click on the Add/Update button to update the server with new group configuration information.

3.4 Starting the DHCP Server (joind)

After you install the OSFINET440 optional subset, run the installation script, and configure the server, you must start the server to implement the new configuration. Use the Network Configuration application of the Common Desktop Environment (CDE) Application Manager to start the DHCP server on systems with graphics capabilities.

To start the Network Configuration application, log in as root, and double click on Network Configuration icon in the Configuration group. The Network Configuration main window is displayed, showing available network components and configured network components.

To exit the Network Configuration application, choose File then Exit. See the `netconfig`(8X) reference page for more information.

To start the DHCP server, do the following:

1. Select the DHCP Server Daemon from the Available Network Components list box in the Network Configuration main window.
2. Click on Configure. The Configuring DHCP Server Daemon Dialog Box is displayed.
3. Click on the Enable button in the DHCP Server Daemon field to start the DHCP server daemon each time the system boots.
4. Click on Commit to save the configuration and display a pop-up dialog box.
5. Click on Yes to start the DHCP daemon and close the pop-up window.
6. Click on Close to close the Configuring DHCP Server Daemon dialog box. See the online help for additional information.

Note

> You should never 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.

For more information about `joind`, see the `joind`(8) reference page.

3.5 Starting the DHCP Client

When you configure the basic network connections on the client system you must specify an Internet address source. If you specify DHCP server and restart the network, the DHCP client daemon starts and uses DHCP to obtain IP configuration information. From then on, the DHCP client automatically starts each time the client computer boots.

3.6 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` application 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` application 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 the `xjoin`(8) reference page and the *JOIN Server Administrator's Guide* (`/usr/doc/join/TOC.html`) for more information.

3.7 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. 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` application 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 Return or Tab after each entry. Specify the integer –1 for Lease Expiration to ensure that the IP address assignment will be 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.

3.8 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 3.2.2.1.) 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 Return 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.

Alternatively, you can import a file into the MAC address database by clicking on Import and entering a file name. See the `jdbmod`(8) reference page 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 Delete.

3.9 Configuring a BOOTP Client

To register a client to use BOOTP only, do the following:

1. Log in as root.
2. Invoke the `xjoin` application 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 then select DHCP parameters.
5. Click on the File/Update button to update the server with these changes.

3.10 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.

3.11 Solving DHCP Problems

If DHCP clients have problems obtaining DHCP information from the server, do the following:

1. Log in as root.
2. Stop the `joind` daemon with the `kill –HUP` command. (Never use `kill –9` to stop the DHCP daemon; it can corrupt your database files).
3. 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:

   a. Edit the `/etc/inetd.conf` file and add the `–d4` flag.
   b. Stop the `joind` daemon with the `kill –HUP` command.
   c. Stop the `inetd` daemon with the `kill –HUP` command. This forces `inetd` to reread the `/etc/inetd.conf` file.

4. Review the `/var/join/log` file for information about the cause of any DHCP client problems.

Example 3–1 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.

Example 3–1: Sample DHCP Log File Message

```
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`.

4

Point-to-Point Connections

The Tru64 UNIX system supports point-to-point connections using the following protocols:

- Serial Line Internet Protocol (SLIP)
- Point-to-Point Protocol (PPP)

This chapter describes both environments, how to plan for both environments, how to configure your system for both environments, and how to manage both environments.

4.1 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.

4.1.1 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 4–1 shows both of these simple SLIP configurations. Figure 4–2 shows a SLIP connection between two systems with HOSTB acting as a gateway system.

Figure 4–1: Sample Simple SLIP Configuration

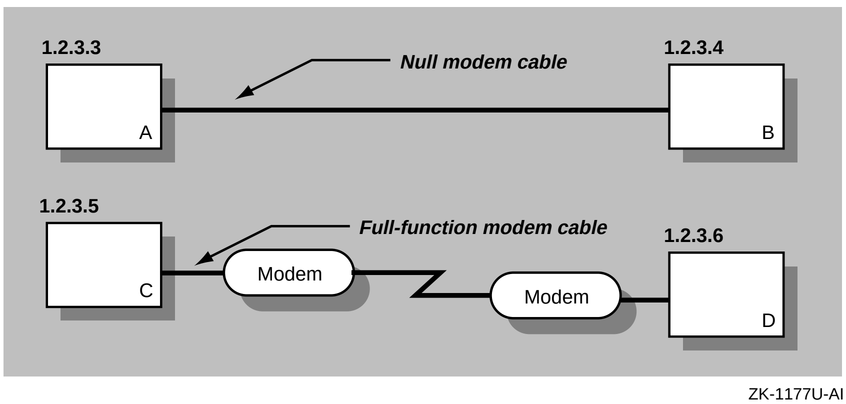
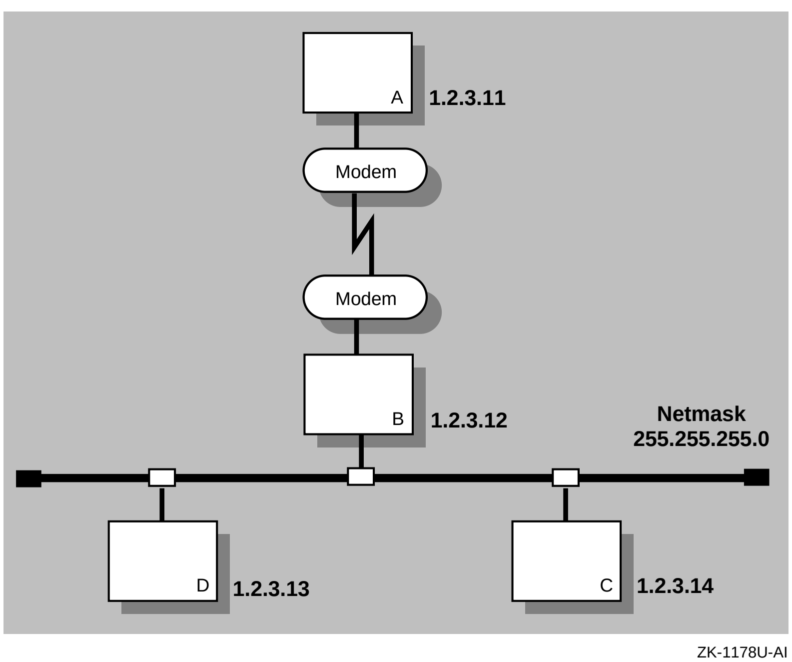


Figure 4–2: SLIP Configuration with Gateway System



4.1.2 Planning SLIP

This section describes the tasks you must complete before configuring SLIP.

4.1.2.1 Verifying the Hardware

When you verify the hardware, you 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 4–1.

Table 4–1: 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 4–6 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, you should 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 control flow, 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.

4.1.2.2 Preparing for the Configuration

After you verify the communication hardware, you set up the system to run SLIP. Appendix A contains a worksheet that you can use to record the information that you need to provide to configure SLIP. If you are viewing this manual online, you can use the print feature to print part of the worksheet.

Figure 4–3 shows Part 3A of the Configuration Worksheet. The following sections explain the information you need to record in Part 3A of the worksheet.

Figure 4–3: Configuration Worksheet, Part 3A

Part 3A: SLIP Setup

Type of connection:	☐ Hardwired	☐ Modem
Type of system:	☐ Dialin	☐ Dialout
Local IP address:	________	
Network mask:	________	
Destination IP address:	________	
Terminal name:	________	
Speed:	________	
SLIP login information:	________	
Dialout systems		
startslip subcommands:	________	
	________	
	________	
	________	
Dialin systems		
slhosts file options:	________	
Gateway:	☐ Yes	☐ 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 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 4–2 shows the minimum `startslip` subcommands to specify in a setup script file that you create. Table 4–3 shows the optional `startslip` subcommands.

Table 4–2: Basic startslip Subcommands

Subcommand	Information Required
`myip`	Your system's IP address.
`dstip`	The destination system's IP address.
`netmask`	The network mask for the subnetwork.
`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`	Configured the network interface and attaches the serial line to the network interface.

Table 4–3: Optional startslip Subcommands

Subcommand	Description
`debug`	This 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) is not permitted to be sent over the SLIP connection. This frees line bandwidth for more critical traffic.
`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.

Table 4–3: Optional startslip Subcommands (cont.)

Subcommand	Description
tcpauto	The local system compresses 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.

See startslip(8) for a complete list of the startslip subcommands.

slhosts file options

For dial-in systems, Table 4–4 shows a list of options for each SLIP link specified in the /etc/slhosts file.

Table 4–4: slhosts File Options

Option	Description
debug	This generates debugging messages to the daemon.log file.
icmpsup	Suppresses Internet Control Message Protocol (ICMP) traffic. ICMP traffic (such as that generated by the ping command) is not permitted to be sent over the SLIP connection. This frees line bandwidth for more critical traffic.
tcpauto	The local system compresses 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.
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. Do not specify the tcpcomp and tcpauto options together.

See slhosts(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.

4.1.3 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 some system files and use the `startslip` program to configure both dial-in connections and dial-out connections.

4.1.3.1 Configuring a Dial-In System

If the system is to answer calls from remote systems, you want to establish a dial-in connnection. 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 4.3.2 for more information.

   Note

   It is best to 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. The `/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 4–1, 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. 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 `gated`. See Chapter 2 for basic network setup information.

If any problems occur while using SLIP, see Chapter 13.

4.1.3.2 Configuring a Dial-Out System

If the system is to place calls to a remote system, you want to establish a dial-out connection. 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 4–5 in the synchronization string (`ss`) of the entry. The other modem settings can remain as they are.

Table 4–5: Modem Commands for Dial-Out Access

Command	Description
`at&c1`	Normal Carrier Detect (CD) operation. Tells the modem to not raise Carrier Detect until it sees Carrier Detect from the other modem.
`at&d2`	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.
`ate1`	Turns on echoing.
`atq0`	Displays the result codes.
`ats0=0`	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 getty 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. Create a file that contains startslip subcommands for SLIP dial-out connections by doing the following:

   a. Copy the sample script file from the startslip(8) reference page 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 a complete list of subcommands and a sample script file.

4. 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, startslip runs in the background. The telephone number (if any) and the process ID are logged in the /var/run/tty*xx*.tel-pid file.

If any problems occur while using SLIP, see Chapter 13.

4.1.4 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, `tty`*xx* 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 returned in step 1:

```
# kill 821
```

Alternatively, you can turn off your modem to terminate the dial-out connection.

4.2 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 Internet Protocol (IP) datagrams, allowing TCP/IP connections between the peers.

The Tru64 UNIX PPP subsystem is derived from public domain ppp-2.2, and supports IP datagrams. See RFC 1661, RFC 1662, RFC 1332, and RFC 1334 for more information about PPP.

4.2.1 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 4–4 shows two simple PPP configurations with PPP connections between two systems.

Figure 4–4: Simple PPP Configurations

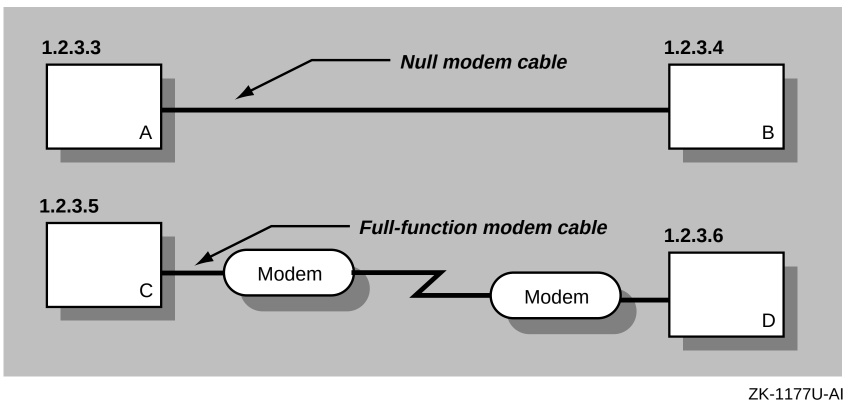
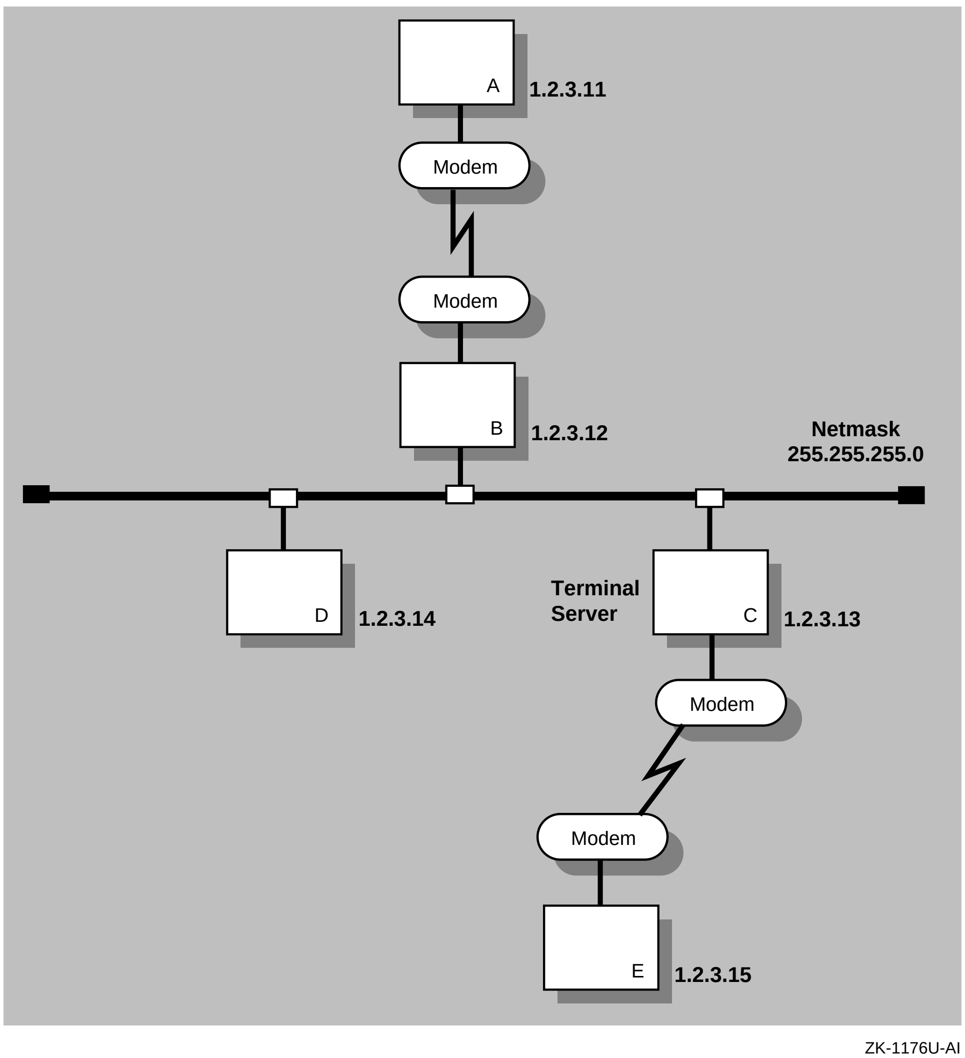


Figure 4–5 shows two PPP connections. The first is between host A and host B, with host B acting as a gateway system. The second is between personal computer E and host D through terminal server C. The latter configuration might be common for employees working at home and dialing in to a system at work.

Figure 4–5: Network PPP Configuration



4.2.1.1 PPP Options

When you invoke the `pppd` daemon you can specify PPP options on the command line. In addition, the following files on a system can contain PPP options:

- `/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 `pppd` to use whenever it is run. 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. Only root should be able to write to this file.

- `/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.

See `pppd`(8) for a description of `pppd` options.

4.2.1.2 Authentication

PPP provides two protocols for authenticating hosts and for authenticating your host system to others:

- Password Authentication Protocol (PAP)
- Challenge Handshake Authentication Protocol (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. Only root should be able to read these files. 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 may 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 will 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 should 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.

For more information about CHAP and PAP authentication, see the pppd(8) reference page.

4.2.2 Planning PPP

This section describes the tasks you must complete before configuring PPP.

4.2.2.1 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 4–1.

If the two systems are connected through modems and telephone lines, see Table 4–6 for a list of modem cables to use. The modems are set to 8 bit, no parity, and connected to the telephone network.

4.2.2.2 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. Add the following entry to the /etc/sysconfigtab file by using the sysconfigdb utility:

```
ppp:
nppp=2
```

This provides for two PPP connections. If your system requires more PPP connections, increase the number.

6. Reboot the system.

4.2.2.3 Preparing for Configuration

After you verify PPP support in the kernel, you configure PPP by performing a sequence of steps. Appendix A contains a worksheet that you can use to record the information that you need to provide to configure PPP. If you are viewing this manual online, you can use the print feature to print part of the worksheet.

Figure 4–6 shows Part 3B of the Configuration Worksheet. The following sections explain the information you need to record in Part 3B of the worksheet.

Figure 4–6: Configuration Worksheet, Part 3B

Part 3B: PPP Setup	
Local IP address:	____________
Remote IP address:	____________
Network mask:	____________
Terminal name:	____________
Speed:	____________
Level of authentication:	____________
Type of authentication:	☐ PAP ☐ CHAP
Options:	________________________

Local IP address

The local system's IP address. For systems connected to a local area network (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).

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, you should require that the remote host authenticate itself and 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

Options to supply the `pppd` daemon. The following options might be useful:

- `defaultroute` — If your system is standalone and you are connecting to the Internet through the remote system, add a default route via the remote host by specifying this option.

- `asyncmap` — If the serial line is not completely 8-bit transparent, specify this option; `asyncmap 200a0000` is appropriate if the serial link includes a `telnet` link.
- `mru` — To improve performance for multiple IP connections, reduce the MRU (maximum receive unit) on the local and remote systems. It is best to specify the `mru 296` option.

See `pppd`(8) for additional options.

4.2.3 Configuring PPP

After you complete the PPP planning tasks, you can establish a PPP connection between your local system and a remote system. Establishing a PPP connection between two systems means setting up a serial link and running the `pppd` daemon on both ends of the link. In a PPP environment, a system can be a dial-in system or a dial-out system.

Guidelines for running the `pppd` daemon are as follows:

- 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, followed by a colon. For example:

  `local_addr`:

- 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 `pppd` 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.

4.2.3.1 Establishing a PPP Dial-Out Connection

If the system will place calls to a remote system, you should to establish a dial-out connection. After you connect your modem to a serial port on your system, do the following:

1. Verify that you can communicate with the modem. Do the following:

   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. Use the `tip` command to access the modem as follows:

```
% tip system_name
```

The *system_name* is stored in the `/etc/remote` file.

d. 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 used by the `chat` program to direct the modem what number to dial and what 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 to automate the dial-out process. Each entry in a `chat` script has the following format:

   [*string_chat_expects string_chat_sends*]

   For example, the following file named `/etc/ppp/chat-script` contains the following information:

```
"" atdt2135476   1
CONNECT   2
login: myname   3
Password: "\qmypassword"   4
"$ " "\qpppd"   5
```

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 $ (the shell prompt) and sends pppd to start the pppd daemon on the remote machine. The \q cancels the effect of the previous \q.

See the chat(8) reference page for more information on chat and chat scripts.

4. Edit the /etc/ppp/options file and include the following pppd options as required by the remote system or ISP:

```
defaultroute   1
asyncmap 0   2
mru 296   3
netmask dd.dd.dd.dd   4
lcp-echo-interval 60   5
lcp-echo-failure 5   6
noipdefault   7
crtscts   8
debug   9
```

1 If your system is standalone and you are connecting to the Internet through the remote system, specify this option to add a default route via the remote host.

2 If the serial line is not completely 8-bit transparent, specify this option; asyncmap 200a0000 is appropriate if the serial link includes a telnet link.

3 Reduces the maximum receive unit (MRU) on the local and remote systems to improve performance for multiple IP connections.

4 Sets the interface netmask to the specified value. Your ISP should provide this information.

5 Sends a Link Control Protocol (LCP) echo request frame to the remote system every 60 seconds to determine whether the link to the remote system is still active.

6 If the local system does not receive a response from the remote system after 5 LCP echo request frames, pppd considers the link broken and disconnects.

7 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.

8 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.

9 Enables debugging. All log messages are sent to the file specified in the `/etc/syslog.conf` file. After your connection is working correctly, remove this entry from the PPP options file.

See `pppd`(8) for a complete list of `pppd` options.

5. If necessary, create a PAP or CHAP secrets file, the format of which is discussed in Section 4.2.1.2.

6. Edit the `/etc/syslog.conf` file to enable message logging, as follows:

Note

Whitespace in the `/etc/syslog.conf` file, as in the following procedure, 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* guide.

   b. Add the following entry to the file to create a `ppp-log` file. Note that any whitespace in this file must consist of tab characters:

```
local2.debug                        /etc/ppp/ppp-log
```

   c. Save the edits and close the file.

7. Stop and restart the `syslogd` daemon by entering the following commands:

```
# /sbin/init.d/syslog stop
# /sbin/init.d/syslog start
```

8. Invoke `pppd` 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'
```

9. 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
   ```

If problems occur while using PPP, see Chapter 13.

4.2.3.2 Establishing a PPP Dial-In Connection

If the system will answer calls from remote systems, you should establish a dial-in connnection. 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 4.3.2 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`; 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. Edit the `/etc/ppp/options` file and include the following `pppd` options required to support dial-in access for all remote users:

   ```
   netmask dd.dd.dd.dd   1
   proxyarp   2
   crtscts   3
   asyncmap 0   4
   :remote_ip_address   5
   debug   6
   ```

1 Sets the interface netmask to the specified value.

2 Adds an entry to the local system's Address Resolution Protocol (ARP) table containing the IP address of the remote system and the Ethernet address of the local system.

3 Enables hardware flow control for the serial port.

4 If the serial line is not completely 8-bit transparent, specify this option; `asyncmap 200a0000` if the serial link includes a telnet link.

5 Specifies an IP address for the remote system.

If you want to specify options for each serial port, create a `/etc/ppp/options.tty`*xx* file and include the remote IP address and any other options that apply to that specific serial port. See `pppd`(8) for a complete list of `pppd` options.

6 Enables debugging. All log messages are sent to the file specified in the `/etc/syslog.conf` file. After your connection is working correctly, remove this entry from the PPP options file.

After an incoming call is received and a connection established, `startppp` runs in the background. The process ID is logged in the `/etc/ppp/ppp`*xx*`.pid` file.

If problems occur while using PPP, see Chapter 13.

4.2.4 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, `ppp`*xx* specifies the `pppd` used for the PPP connection. The `pppd` specified in the command notifies related `pppd` daemons to terminate (clean up and exit).

If `pppd` is attached to a hardware serial port connected to a modem, it should 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.

4.3 Guidelines for Using Modems

The Tru64 UNIX system 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/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 4.1.2.1 for specific information on SLIP and PPP connections and see Chapter 9 for information about UUCP connections.

4.3.1 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 4–6 lists the cables you should 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 4–6: Types of Modem Cable

Cable Number	Description
`BC22E-`*xx*[a]	16-wire modem cable (male DB25 pin to female DB25 pin cable)
`BC22F-`*xx*[a]	25-wire modem cable (male DB25 pin to female DB25 pin cable)
`BC29P-`*xx*[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.

4.3.2 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 a mismatch.

2. 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.

3. 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.

4. Set up the modem for dial-in access. Table 4–7 lists the AT commands required. Most of these command settings are the default settings.

Table 4–7: Modem Commands for Dial-In Access

Command	Description
`at&c1`	Normal Carrier Detect (CD) operation. Tells the modem not to raise Carrier Detect until it sees Carrier Detect from the other modem.
`at&d2`	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.
`atq1`	Sets the modem to quiet mode. Result codes are not sent to the system.
`ate0`	Echo off. This prevents modem from echoing the login prompt issued by the `getty` process.
`ats0=`*n*	Specifies the number of rings to wait before answering. If *n* = 0 (zero), the modem will not answer.
`at&w0`	Saves the current modem settings in NVRAM.

Tru64 UNIX supports both hardware and software flow control. If the system 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, you should use software flow control.

5. 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 uugetty utility, you must install the UNIX-to-UNIX Copy Facility subset.

6. As root, start the getty or uugetty process by entering the following command:

```
init q
```

   The getty or uugetty process starts, then goes to sleep, waiting for someone to dial in to the system.

4.3.3 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 tip:

```
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. Among the information you can supply is the terminal device name, speed, and the /etc/acucap file that defines your modem. For example, the following two entries are for the modem specified in step 1a:

```
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.

See remote(4) for more information.

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. 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 `tip`, 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.

5

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, and provides information for setting up and maintaining LAT. In addition, this chapter provides information on the LAT startup file, `latstartup.conf`, the system `inittab` file, and the different LAT configurations.

For additional introductory information on LAT, see `lat_intro`(7). For information on solving LAT problems, see Chapter 13.

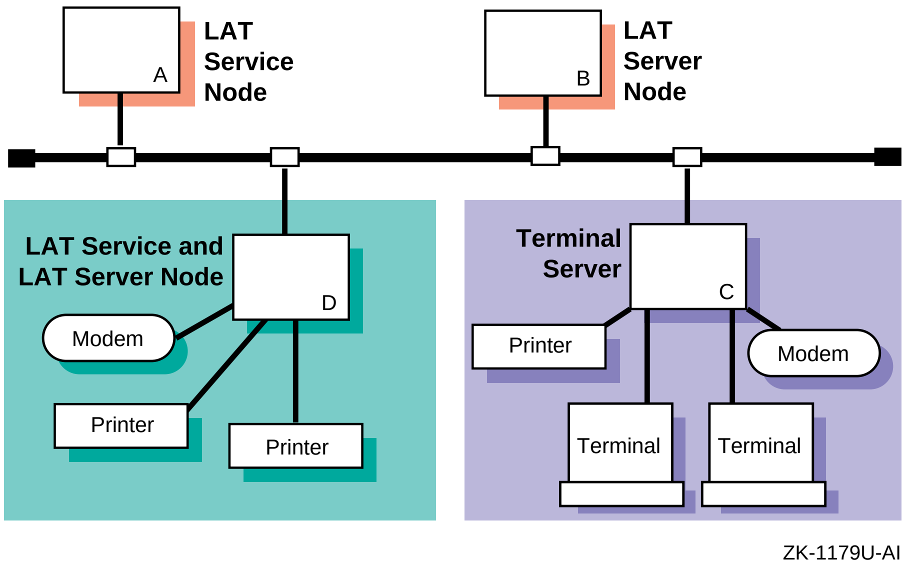
5.1 LAT Environment

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 5–1 shows a sample LAN with LAT server nodes and LAT service nodes.

Figure 5–1: Sample LAT Network Configuration



The LAT software also permits host applications to initiate connections to server ports, designated as application ports, to access remote devices. This section describes:

- Types of LAT connections
- Access control in a LAT network
- Password specification for remote servers
- Load balancing

5.1.1 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 5–1 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 set up host A to use a printer on host D in Figure 5–1 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 5–1 is using an outgoing connection.
- Lattelnet 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 connecting to the lattelnet service on host D in Figure 5–1 is using a lattelnet connection.

5.1.2 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 users to enter a password to gain access to terminal servers. (Refer to your terminal server documentation.)
- LAT groups: You can establish LAT groups and restrict host communication to particular groups by designating those groups on a LAT service node (by issuing a `latcp –g –a` command), on a LAT server node (by issuing a `latcp –u` command), and on a terminal server (refer to your terminal server documentation).

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).

5.1.3 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.

5.1.4 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.

5.2 Planning LAT

This section describes the tasks you must complete before configuring LAT.

5.2.1 Verifying That the LAT Subset Is Installed

Verify that the LAT subset is installed by entering the following command:

```
# setld -i | grep LAT
```

If the LAT subset is not installed, install it by using the `setld` command. For more information on installing subsets, see `setld`(8), the *Installation Guide*, or the *System Administration* manual.

After the LAT subset is installed, your system is configured to dynamically load the LAT module into the running kernel when the system boots.

5.2.2 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.

5.2.3 Preparing for the Configuration

After you verify DLB support in the kernel, you configure LAT by using the `latsetup` utility. Appendix A contains a worksheet that you can use to record the information that you need to provide to configure LAT. If you are viewing this manual online, you can use the print feature to print part of the worksheet.

Figure 5–2 shows Part 4 of the Configuration Worksheet. The following sections explain the information you need to record in Part 4 of the worksheet.

Figure 5–2: Configuration Worksheet, Part 4

Part 4: LAT Setup

Start LAT automatically at boot time: Yes ☐ No ☐
Type of tty devices: ________________
Number of LAT tty devices: ________________
Number of LAT entries (getty) in /etc/inittab: ________________

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 an unlimited number of devices.

SVR4 device special files have the following format:

`/dev/lat/`*n*

The value *n* is a number between 620 and approximately 5000. For example, `/dev/lat/620`, `/dev/lat/777`, and `/dev/lat/4000` specify SVR4 devices. The SVR4 format allows you to create an unlimited number of devices.

BSD device special files have the following format:

`/dev/tty`*WX*

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/tty02`, `/dev/tty0e`, and `/dev/tty9f` specify BSD LAT terminal devices. However, all BSD terminal device names are not case sensitive. The device special files `/dev/tty9f` and `/dev/tty9F` 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 may 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.

5.3 Configuring LAT

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.

To invoke the latsetup utility choose the Local Area Transport (LAT) option from the Setup Menu or enter the following command:

```
# /usr/sbin/latsetup
```

If your terminal does not support curses, you must specify the –nocurses flag. This flag allows you to run latsetup 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.

5.4 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
```

If you stop LAT from within a LAT session, the session will hang. You will have to reboot your system to clear up the problem.

5.5 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 5–1 lists the LAT parameters and their default values.

Table 5–1: LAT Parameters

Parameter	Default Value	
Node name	Host name	
Multicast timer	60 seconds	
Network adapter	All network adapters connected to broadcast media.	
Service name	From the LAT node name parameter. Each service has the following parameters:	
	Parameter	**Default Value**
	Service description	“DIGITAL UNIX Version *X.X* 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.

Example 5–1 shows a sample `/etc/latstartup.conf` file.

Example 5–1: 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 5.11 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 two LAT devices to it: 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.

Note

A latcp command that adds a service must occur in the latstartup.conf file before a latcp command requiring the service name. Lines 3 and 4 in Example 5–1 illustrate this point.

5.6 Customizing the inittab File

You can modify the /etc/inittab file to use a program other than getty. 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 5.12. 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.

5.7 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. For adapters connected to different logical networks, this is probably desirable. However, for adapters connected to a single logical network, you should specify that the LAT protocol run 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 names of adapters defined on your system.

5.8 Setting Up Printers

Before you set up a printer to print through LAT you should be familiar with setting up printers. See the *System Administration* manual, the Printer Configuration System Administration utility (if using CDE), and `lprsetup.dat`(4), and `lprsetup`(8) (if not using CDE) for information on setting up printers. In addition, you need the following information:

- The name of the terminal server to which the printer is to be attached
- Either or both of the following:
  - The name of the port to which the printer is to be attached
  - The name of the service assigned for the remote printer
- Terminal server documentation
- Printer documentation

This section provides information on how to set up a printer to print through LAT, using host-initiated connections.

Note

The examples in this section use the DECserver 700 server. Please refer to the documentation supplied for your terminal server. In addition, the examples use information from preceding procedures.

5.8.1 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
   ```

5.8.2 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.

5.8.3 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 5.8.1.

For example, the following command maps the local application port 621 for the server LOCSER to the remote printer port port07.

```
# latcp -A -p 621 -H LOCSER -R port07
```

The following command does the same thing, but specifies the remote printer service name instead of the remote print port:

```
# latcp -A -p 621 -H LOCSER -V REMprinter07
```

For more information, see `latcp`(8).

5.8.4 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/lat/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. For example, the `/dev/lat/621` defined previously. You must specify the full path name for the lp symbol.

5.8.5 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 information on errors that could have occurred.

5.9 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. The printer connections are discussed in Section 5.8.

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.

5.9.1 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 the preceding 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, you should 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.

5.9.2 Program Interface

Applications developed to 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 variable `errno`.
- 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 `OSFEXAMPLES440` 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 `OSFEXAMPLES440` 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).

5.10 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.

5.10.1 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 in the preceding example.

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 `latcp8` and `lat_intro`(7) for more information on learned services.

5.10.2 Program Interface

Applications developed to employ outgoing connections adhere to the same guidelines as applications developed for host-initiated connections. See Section 5.9.2 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 `OSFEXAMPLES440` optional subset.

5.11 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 lattelnet. 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 devices 624, 625, and 626. The last field in each line lattelnet is the name of the optional service defined in step 1.

   ```
   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 syslog function to log messages to the /var/adm/syslog.dated/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 the preceding 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:
```

5.12 Creating Your Own Dedicated or Optional Service

A Tru64 UNIX host can offer the following types of services:

- Bound interactive
- Unbound interactive
- Dedicated (or optional)

These services are described in `lat_intro`(7). For more information on the commands used to create these services, see `latcp`(8).

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 5.11.) 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 `/etc/inittab` entry for every simultaneous service you want to run. The previous example only allows for 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
```

5.13 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. 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.

5.13.1 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 show you 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.

5.13.2 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.

6
Domain Name Service

The Domain Name Service (DNS) is a mechanism for resolving unknown hostnames and Internet Protocol (IP) addresses that originate from sites on your company's intranet or the Internet. A database lookup service that is part of the DNS daemon searches for the unknown hosts in local and remote `hosts` databases, which are distributed networkwide by the DNS.

The implementation of DNS in Tru64 UNIX is based on Version 4.9.3 of the Berkeley Internet Name Domain (BIND) service, which is maintained by the Internet Software Consortium.

This chapter describes the DNS environment, how to plan for DNS, how to configure your system for DNS, and how to manage DNS servers and clients. For introductory information on DNS, see `bind_intro`(7). For additional information about BIND service, visit the Internet Software Consortium website at `www.isc.org`.

6.1 The DNS Environment

In the DNS environment, systems can have the following roles:

- Primary server — A system that is an authoritative source for information about a zone or zones and that maintains the master copy of the DNS database for the zone or zones.

  This system runs the `named` daemon, answers requests from clients and other servers, caches information, and distributes the databases to secondary servers.

- Secondary server — A system that is an authoritative source for information about a zone or zones, but does not maintain the master copy of the DNS database for the zone or zones. Instead, a secondary server loads its database files from the primary server and periodically polls the primary server to ensure that its databases are up to date.

  This system runs the `named` daemon, provides backup for the primary server, answers requests from clients and other servers, and caches information.

- Slave server — A server that might be an authoritative source for information about a zone or zones, but is restricted as to how it obtains information about zones for which it is not authoritative.

  This system runs the `named` daemon and answers queries from other servers and clients from its authoritative data and cache data. If the information is not present, it forwards queries to a list of systems called forwarders specified in its `named.boot` file. The queries are forwarded to each forwarder system until the list is exhausted or the query is satisfied. Slave servers store the information they receive until the data expires.

- Caching server — A system that is not authoritative for any zones. This system runs the `named` daemon and services queries from other servers and clients by asking other servers for the information and caching the information it receives. Information is stored until the data expires.

- Client — A system that queries a server for host name and address information, interprets responses, and passes information to requesting applications. The client is also called a resolver. A client does not run the `named` daemon.

DNS runs on each system in your network. You must decide what role each system will play within the DNS environment that you are creating. For each domain, select one host to be the primary server; there can be only one primary server for each domain. Select one or more hosts to be secondary, slave, and caching servers. The rest of the hosts should run as DNS clients.

Figure 6–1 shows a domain in which there are two servers, one on each subnet, and multiple clients. Server A has primary authority for the zone and maintains the database files for the zone. Server B has secondary authority for the zone, obtaining a copy of the zone database from Server A and answering queries from clients.

Figure 6–1: Sample Small DNS Configuration

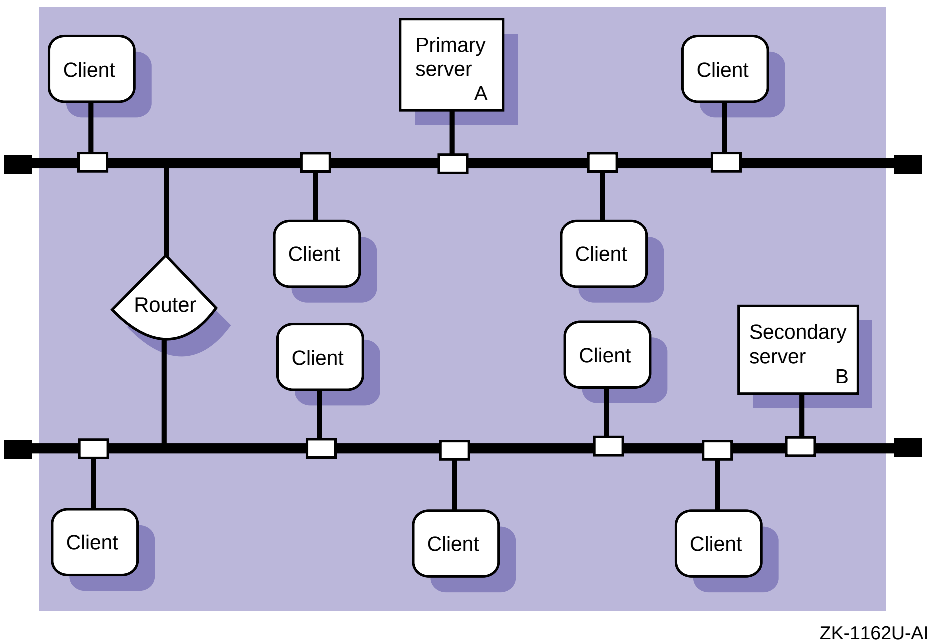
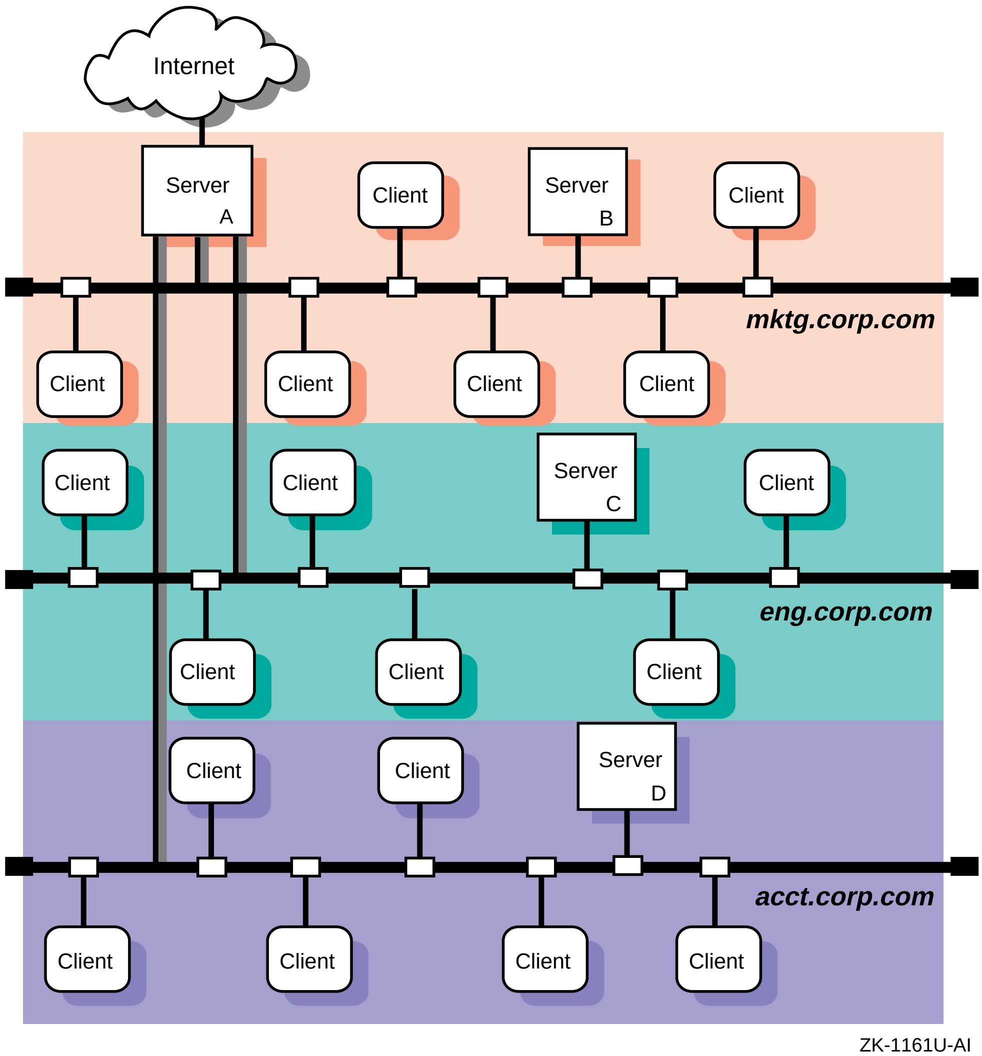


Figure 6–2 shows a domain in which there are three zones: `mktg.corp.com`, `eng.corp.com`, and `acct.corp.com`. Server B has primary authority for zone `mktg.corp.com` and secondary authority for each of the other two zones. Server C has primary authority for zone `eng.corp.com` and secondary authority for each of the other two zones. Server D has primary authority for zone `acct.corp.com` and secondary authority for each of the other two zones. Server A is both a router and a caching server. As a caching server, it caches information it receives from queries out of the parent domain.

Figure 6–2: Sample Large DNS Configuration



6.2 Planning DNS

Appendix A contains a worksheet that you can use to record the information that you need to provide to configure DNS. If you are viewing this manual online, you can use the print feature to print a copy of this part of the worksheet.

Figure 6–3 shows Part 5 of the Configuration Worksheet. The following sections explain the information you need to record in Part 5 of the worksheet.

Figure 6–3: Configuration Worksheet, Part 5

Part 5: DNS Setup

Local domain name: ______________________

Server

Scope: ☐ Master ☐ Slave

Host name resolution: ☐ `/etc/hosts` ☐ DNS

Zones

Zone domain name:	Authority:	Data file or server address:
__________	☐ Primary ☐ Secondary	__________
__________	☐ Primary ☐ Secondary	__________
__________	☐ Primary ☐ Secondary	__________
__________	☐ Primary ☐ Secondary	__________

Forwarders

Forwarder name: __________ __________

Client

Server name:	Internet address:
__________	__________
__________	__________
__________	__________
__________	__________

Host name resolution: ☐ `/etc/hosts` ☐ DNS

Local domain name

The parent domain name of which your local system is a part. For example, if your system's domain name is `cxcxcx.abc.xyz.com`, your local domain name is `abc.xyz.com`.

6.2.1 Server

Scope

If you want to restrict your system to query a specific list of systems (forwarders) only, check Slave; otherwise, check Master.

Host name resolution order

The first source in resolving host name queries on your system. If you want to query DNS first, check DNS. If you want to query the `/etc/hosts` file first, check `/etc/hosts`.

Zone domain name

The name of the top-level domain in the zone.

Authority

If the server is a primary authority for information about the zone (maintains the zone database file), check Primary. If the server is a secondary authority for information about the zone, check Secondary.

Data file

For a server that is a primary authority for information about a zone, the pathname of the file that is the master copy of zone information.

For a server that is a secondary authority for information about a zone, the pathname of the file that is to contain zone information obtained from the primary server. This is optional, but useful when the server restarts. Instead of waiting to obtain information from a primary server, which might not be available, the secondary server can restart using the information in the data file.

Server address

For a server that is a secondary authority for information about a zone, the address of the server that has primary authority for the zone domain.

Forwarder name

The host name of a system or systems to which your server will forward queries that it cannot resolve locally. When the server receives a query that it cannot answer from its cache, it sends the query to a forwarder for resolution. If the forwarder cannot answer the query, the server might contact other servers directly. If you checked Slave in the Scope field, you must include forwarder names; otherwise, forwarders are optional.

6.2.2 Client

Server name

The name of a server to contact for host name resolution. Specify up to three servers.

Internet address

A corresponding IP address for the server or servers.

Host name resolution order

The first source in resolving host name queries on your system. If you want to query DNS first, check DNS. If you want to query the `/etc/hosts` file first, check `/etc/hosts`.

6.3 Configuring DNS

Use the BIND Configuration application of the Common Desktop Environment (CDE) Application Manager to configure DNS on systems with graphics capabilities. You can configure the following systems:

- Primary server
- Secondary server
- Caching server
- Slave server
- Client

See `bindconfig`(8X) for more information on the BIND Configuration application.

To invoke the BIND Configuration application, log in as root and do the following:

1. Click on the Application Manager icon on the CDE front panel.
2. Double-click on the System_Admin application group icon.
3. Double-click on the Configuration application group icon.
4. Double-click on the BIND Configuration application icon. The BIND Configuration main window is displayed, showing available DNS service types and configured DNS service types.

Note

You must first set up the primary server; then, you can configure the other systems in any order.

To exit the BIND Configuration application, choose File then Exit.

Note

For systems without graphics capabilities, you can use the `bindsetup` utility. See `bindsetup`(8) for more information.

The BIND Configuration application also has an extensive online help system. You can use it instead of the instructions in this section to configure DNS on your system.

6.3.1 Configuring a DNS Server

To configure a server, do the following:

1. Select Server from the Available BIND Services Types field in the BIND Configuration main window.
2. Click on the Configure button to display the Configure Server dialog box.
3. Click on the appropriate radio button in the Scope field. If you click on the Slave radio button, go to step 8.
4. Enter the domain name in the Local Domain input text box.
5. Indicate the order in which to resolve host name queries in the Host Name Resolution field. Click on the first radio button if you want to query DNS before checking the `/etc/hosts` file. Click on the second radio button if you want to check the local `/etc/hosts` file before querying DNS; this is recommended.

   Alternatively, you can run the `svcsetup` script to customize service order selection. See Section 6.4 and `svcsetup`(8) for information on modifying the `svc.conf` file.
6. If your system does not have authority for any zone, go to step 8.
7. For servers that have authority for a zone or zones, do the following:

   a. Click on Zones to display the Zones Served dialog box.

   b. Click on Add to display the Add Zone dialog box.

   c. Enter the domain name in the Domain text box.

   d. Click on the Primary radio button if this system is the primary server for this zone. Click on the Secondary radio button if this system is a secondary server for this zone.

   e. If you are primary or secondary authority for this zone, enter the name of the zone data file in the Data File input text box. If you are using an existing `/etc/hosts` file to create the database, this is the name of the data file you create after you exit the BIND Configuration application.

   f. If this system is a secondary server for this zone, enter the IP address of the primary server in the Server Addr input text box.

g. Click on OK to accept the configuration, add the zone to the list of zones served, and close the Add Zone dialog box. Repeat step 7b and all subsequent steps in this sublist for other zones for which you have authority.

h. Click on OK to accept the configuration and close the Zones Served dialog box.

8. If you want to forward queries to a specific server or servers (forwarders) for resolution, do the following:

   a. Click on Forwarders to display the Forwarders dialog box.

   b. Enter the name or IP address for the new forwarder in the Forwarder input text box. If you enter a host name, it must be included in the `/etc/hosts` file.

   c. Click on Insert to place the new forwarder address at the end of the list. Repeat step 8b and this step for each forwarder.

   d. Click on OK to accept the list of forwarders and close the Forwarders dialog box.

9. If you want to start the `named` daemon, do the following:

   a. Click on Name Daemon to display the Configure Named Daemon dialog box.

   b. Click on OK to accept the configuration, start the `named` daemon, and close the Configure `named` Daemon dialog box.

   If you do not want to start the `named` daemon now, use the following command to start the daemon manually in a terminal window after you are finished with the BIND Configuration application:

   ```
   # /sbin/init.d/named start
   ```

10. Click on Commit to accept the configuration and start the appropriate daemons.

11. Click on Close to close the Configure Server dialog box.

You can also use the BIND Configuration application to modify your server configuration. See the online help and `bindconfig`(8X) for more information.

If your system is a primary authority for information about a zone or domain and you want to create the database from an existing `hosts` file, do the following:

1. Copy the `hosts` file that you want to convert to the DNS `hosts` database into the `/etc/namedb/src` directory.

To create the source file from which the `hosts` database will be created, update the primary server's local `/etc/hosts` file and then copy it into the `/etc/namedb/src` directory. Note that if a system, host1 for example, is in your DNS domain and is running DNS but is not included in the primary server's `hosts` database, other systems in the domain cannot obtain the IP address of host1. See Example 6–1 for a list of sample `/etc/hosts` file entries.

Note

Note that the file that you copy into the `/etc/namedb/src` directory must be named `hosts`.

2. To convert the `hosts` file in `/etc/namedb/src` directory to the appropriate DNS format, enter the following commands:

```
# cd /etc/namedb
# make hosts
```

Example 6–1: Sample /etc/hosts File

```
127.0.0.1 localhost
120.105.5.1 host1
120.105.5.2 host2
120.105.5.3 host3
120.105.5.4 host4
120.105.5.5 host5
```

6.3.2 Configuring a DNS Client

To configure a DNS client, do the following:

1. Select Client from the Available BIND Services Types field in the BIND Configuration main window.
2. Click on Configure to display the Configure Client dialog box.
3. Enter the domain name in the Local Domain input text box.
4. Click on the Host Name text field and enter a host name for the name server.
5. Click on the Address text field and enter the IP address for the name server.

   The addresses are placed in the `/etc/resolv.conf` file, where the resolver uses them to determine the IP addresses of name servers it should query.

6. Click on the appropriate button to add the host name to the list of name servers. If the address is not in the /etc/hosts file, a dialog box appears asking you if you want to add it. To add other name servers, go to step 4 and repeat the steps that follow.
7. Indicate the order in which to resolve host name queries in the Hostname Resolution Order field. Click on the First radio button if you want to query DNS before checking the /etc/hosts file. Click on the Second radio button if you want check the local /etc/hosts file before querying DNS; this is recommended.

   Alternatively, you can run the svcsetup script to customize service order selection. See Section 6.4 and svcsetup(8) for information on modifying the svc.conf file.
8. Click on Commit to accept the configuration and start the appropriate daemons.
9. Click on Close to close the Configure Client dialog box.

You can also use the BIND Configuration application to modify your client configuration. See the online help and bindconfig(8X) for more information.

6.4 Modifying the svc.conf File with svcsetup

You can modify the /etc/svc.conf file without running the BIND Configuration application. To do this, you invoke the svcsetup script using the following command:

```
# /usr/sbin/svcsetup
```

Once invoked, use the following steps to edit the /etc/svc.conf file:

1. Press Return following the informational messages to continue.
2. Press Return to choose the m option from the Configuration Menu.
3. Choose option 2 from the Change Menu. Option 2 corresponds to the hosts database.
4. Enter the number that corresponds to the order in which you want the services running on your system queried for hosts data.

   Listing local first means that the local /etc/hosts file is searched first for the requested information. If the information is not found locally, then DNS servers, NIS servers, or both, are queried, depending on which options you choose.

Note

For better performance, the first service that your system queries for all databases should be local, regardless of what services you are running.

Choose option 3, 4, 5, or 6 to configure the `svc.conf` file so that DNS serves `hosts` information.

The `svcsetup` script indicates that it is updating the `/etc/svc.conf` file. When `svcsetup` is finished updating the file, the script notifies you and the system prompt (#) is displayed.

6.5 Updating DNS Data Files on the Primary Server

Occasionally you may need to update the DNS data files; for example, you may need to add a host to the data files. To do this, use the `bindconfig` application as follows:

1. Select Server from the Available BIND Services Types field in the BIND Configuration main window.
2. Click on Modify to display the Configure Server dialog box.
3. Click on Zones to display the Zones Served dialog box.
4. Click on the zone whose data file you want to modify from the list.
5. Click on Modify to display the Modify Zone dialog box.
6. Click on Resource Record to display the Resource Record dialog box.
7. Click on Add to display the Add Resource Record dialog box.
8. Choose the parameters to change.
9. Click on OK to close the Add Resource dialog box and add the new resource record to the list of resource records.
10. Click on OK to close the Resource Record dialog box.
11. Click on OK to close the Zones Served dialog box.
12. Click on Commit to close the Configure Server dialog box.

Alternatively, to update a data file, you can do the following:

1. Edit the `/etc/namedb/src/hosts` file to add the new host.
2. Change to the `/etc/namedb` directory and enter one of the following commands:

```
# make hosts
# make all
```

After you edit the `hosts` file and enter the `make` command, the DNS conversion scripts (which are in the `/etc/namedb/bin` directory) do the following for you:

1. Create the new hosts databases: `hosts.db` and `hosts.rev`.
2. Place the new databases in the `/etc/namedb` directory.
3. Send a signal to the `named` daemon to reload all databases that have changed.

Note

If you have manually entered mail exchanger (MX) records in the `hosts.db` file, these records are lost. You should edit the `hosts.db` file and add the MX records.

The DNS database conversion scripts also increment the serial number field of the start of authority (SOA) entry in the database file. When the secondary servers poll the primary server and see that the serial number field has changed, they know to refresh their data.

The process is the same for all of the valid files in the primary server's `/etc/namedb/src` directory.

Scripts are provided to create the `hosts.db` and `hosts.rev` databases.

6.6 Obtaining Host Name and IP Address Information

There are several ways that you can obtain information about host names, IP addresses, and user information from a system using the DNS service. The following sections provide an introduction to two commands: `nslookup` and `whois`.

6.6.1 The nslookup Command

You can use the `nslookup` command to noninteractively and interactively query the DNS service for information about hosts on local and remote domains. You can also find information about DNS resource records such as mail exchanger (MX), name server (NS), and so forth.

For a noninteractive query, use the following syntax:

nslookup *hostname*

The output is the server name and address and the host name and address.

For an interactive query, use the following syntax:

nslookup

The output is the default server name and address and the `nslookup` prompt, a greater than sign (>).

For example, to obtain information about MX, you need to query `nslookup` interactively, supplying a valid domain name. The following example shows how to find the mail recipient for the domain `corp.com`:

```
# nslookup
Default Server:  localhost
Address:  127.0.0.1

> set querytype=mx
> corp.com
Server:  localhost
Address:  127.0.0.1
findmx.corp.com      preference = 100, mail exchanger = gateway.corp.com
gateway.corp.com     inet address = 128.54.54.79
>  Ctrl/D
#
```

A good way to learn how to use the `nslookup` command is to experiment with it. To obtain a list of the interactive `nslookup` command options, enter a question mark (?) at the `nslookup` prompt. For further information, see `nslookup`(1).

6.6.2 NIC whois Service

The Network Information Center (NIC) `whois` service allows you to access the following information about a domain:

- The name of the domain
- The name and address of the organization responsible for the domain
- The domain's administrative, technical, and zone contacts
- The host names and network addresses of sites providing the DNS for the domain
- The registered users in the domain

For example, to use the NIC `whois` service to obtain information about a domain named `compaq.com`, use the `whois` command and specify the domain name as follows:

```
# whois compaq.com
Registrant:
Compaq Computer Corporation (COMPAQ-DOM)
   P.O. Box 692000
   Houston, TX 77269

   Domain Name: COMPAQ.COM
.
.
The InterNIC Registration Services database contains ONLY
```

```
non-military and non-US Government Domains and contacts.
Other associated whois servers:
   American Registry for Internet Numbers - whois.arin.net
   European IP Address Allocations        - whois.ripe.net
   Asia Pacific IP Address Allocations    - whois.apnic.net
   US Military                            - whois.nic.mil
   US Government                          - whois.nic.gov
```

To query other `whois` servers, use the `–h` option:

```
# whois -h whois.nic.gov whitehouse.gov
 Whitehouse Public Access (WHITEHOUSE-DOM)
   725 17th Street NW Room NEOB 4208
   Washington, DC 20503

   Domain Name: WHITEHOUSE.GOV
   Status: ACTIVE
   Domain Type: Federal
:
```

7

Network Information Service

The Network Information Service (NIS, formerly Yellow Pages) is a distributed data lookup service for sharing information on a local area network (LAN). NIS allows you to coordinate the distribution of database information throughout your networked environment.

This chapter describes the NIS environment, how to plan for NIS, how to configure your system for NIS, and how to manage NIS servers and clients.

For introductory information on NIS, see `nis_intro`(7).

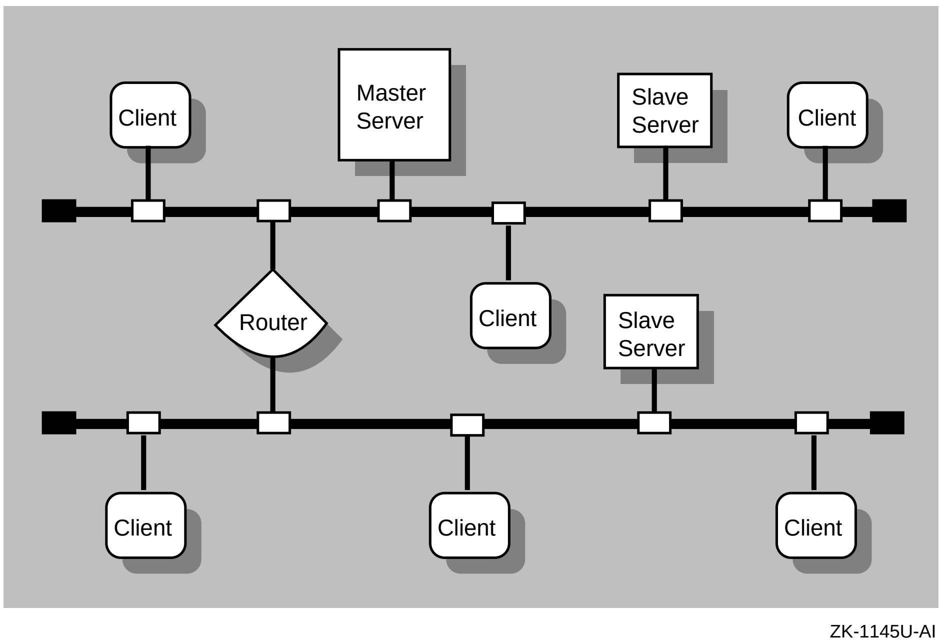
7.1 NIS Environment

In a NIS environment, systems can have the following roles:

- Master server — A system that stores the master copy of the NIS database files, or maps, for the domain in the `/var/yp/`*`DOMAIN`* directory and propagates them at regular intervals to the slave servers. Only the master maps can be modified. Each domain can have only one master server.
- Slave server — A system that obtains and stores copies of the master server's NIS maps. These maps are updated periodically over the network. If the master server is unavailable, the slave servers continue to make the NIS maps available to clients. Each domain can have multiple slave servers distributed throughout the network.
- Client — Any system that queries NIS servers for NIS database information. Clients do not store and maintain copies of the NIS maps locally for their domain.

Figure 7–1 shows a domain in which there is a master server, two slave servers, and some clients.

Figure 7–1: NIS Configuration



By default, NIS distributes the `aliases`, `group`, `hosts`, `netgroup`, `networks`, `passwd`, `protocols`, `rpc`, and `services` databases. (The `mail.aliases` and `netgroup` database are created exclusively for NIS.) You can also create and distribute site-specific customized databases, such as NFS `automount` maps. For information on creating `automount` maps for distribution by NIS, see Appendix C. For information on creating and distributing other site-specific NIS maps, see the Section 7.4.5.

In a C2 secure environment, you can run NIS in a secure mode thereby creating secure and nonsecure versions of the NIS maps. See the *Security* manual for more information.

7.2 Planning NIS

This section describes the tasks you must complete before configuring NIS.

7.2.1 Verifying That the Additional Networking Services Subset is Installed

For NIS servers, verify that the Additional Networking Services subset 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), the *Installation Guide,* or the *System Administration* manual.

7.2.2 Preparing for the Configuration

Appendix A contains a worksheet that you can use to record the information that you need to configure NIS. If you are viewing this manual online, you can use the print feature to print a copy of this part of the worksheet.

Figure 7–2 shows Part 6 of the Configuration Worksheet. The following sections explain the information you need to record in Part 6 of the worksheet.

Figure 7–2: Configuration Worksheet, Part 6

Part 6: NIS Setup

	Domain name:	____________________	
Master Server	/etc/files for maps:	__________	__________
		__________	__________
		__________	__________
		__________	__________
	/var/yp/src/mail.alias file:	☐ Yes ☐ No	
	/ver/yp/src/netgroup file:	☐ Yes ☐ No	
	Setup options:	____________________	
	Slave name:	__________	__________
	IP address:	__________	__________
	Slave name:	__________	__________
	IP address:	__________	__________
Slave Server	Setup options:	____________________	
	Master name:	__________	__________
	IP address:	__________	__________
	Server name:	__________	__________
	IP address:	__________	__________
	Server name:	__________	__________
	IP address:	__________	__________
Client	Setup options:	____________________	
	Server name:	__________	__________
	Server name:	__________	__________

Domain name

The domain name (1 to 31 alphanumeric characters). All systems in the domain must declare the same domain name.

An NIS domain is an administrative entity that consists of a master server, one or more slave servers, and numerous clients. All systems in a domain share the same set of NIS database files.

Note

An NIS domain name is not the same as a DNS domain name. If you configure the system with an incorrect NIS domain name, all NIS-related operations (such as logging in and `ls –l` commands) hang for several minutes, then fail.

Host's role

NIS runs on each system in your network. You must decide what role each system will play within the NIS domain that you are creating. Select one host to be the master server; there can be only one master server for each domain. Select one or more hosts to be slave servers. The rest of the hosts should run as NIS clients.

Note

The master server and all slave servers are also considered to be NIS clients.

7.2.2.1 Master Server

Database files for NIS maps

The files you want to make into NIS maps. Choose from the following list:

- `/var/adm/sendmail/aliases`
- `/etc/group`
- `/etc/hosts`
- `/etc/networks`
- `/etc/passwd`
- `/etc/protocols`
- `/etc/rpc`
- `/etc/services`

/var/yp/src/mail.aliases file

The `mail.aliases` file defines network-wide mail aliases. If you want to define and distribute mail aliases on your network, check Yes; otherwise, check No.

If you choose not to create a `mail.aliases` file, the `nissetup` command issues an informational message that it could not find the `mail.aliases` file while it is building the NIS maps. For information on defining mail aliases, see `aliases`(4).

/var/yp/src/netgroup file

The `netgroup` file defines network-wide groups and is used for permission checking when doing remote mounts, remote logins, and remote shells. If you want to define and distribute `netgroup` information on your network, check Yes; otherwise, check No.

If you choose not to create a `netgroup` file, while it is building the NIS maps, the `nissetup` command issues an informational message that it could not find the `netgroup` file. For information on defining network groups, see `netgroup`(4).

Setup options

The list of setup options for master servers is as follows. Write down the options you want to use in the appropriate place in the worksheet.

- Run the `yppasswdd` daemon (master server only).

  The `yppasswdd` daemon runs on the master server and allows the master copy of the password file to be updated remotely using the `yppasswd` command. Compaq recommends that you run the `yppasswdd` daemon.

- Create secure and nonsecure versions of the NIS maps.

  C2 security, C2 class of trust as defined in the *Trusted Computer System Evaluation Criteria (TCSEC)*, enables you to create secure and nonsecure versions of the NIS maps. Tru64 UNIX provides secure and nonsecure versions of the `passwd` file. For more information, see the *Security* manual and `makedbm`(8).

- Lock the `ypbind` daemon to a particular domain name and server list.

  Normally, hosts broadcast NIS requests on the network and the first available server answers the request. The –S option allows you to lock the `ypbind` daemon to a particular domain and set of servers. Requests are made directly to the specified servers, rather than being broadcast. It is best to run NIS with the –S option configured.

If you choose to run NIS with the –S option configured, you must know the host names and IP addresses of the servers to which you are locking the `ypbind` daemon. You will add them to the local `hosts` file during configuration.

Security Note

When using the `nissetup` script to set up an NIS server that is running with enhanced security, you must answer Yes to the question about locking the domain name and authorized servers (the `ypbind –S` option). For a master server, the server is bound to itself by default.

- Run NIS with the –`ypset` option, the –`ypsetme` option, or with both options set.

  The –`ypset` option allows a user logged in as root on any system in your domain to bind your system to a particular server. The –`ypsetme` option allows `ypbind` to accept –`ypset` requests only from the local system. Compaq recommends that you run NIS with neither the –`ypset` nor the –`ypsetme` options.

Slave name

The name of each slave server in the domain.

IP address

The IP address of each slave server in the domain.

7.2.2.2 Slave Server

Setup options

The list of setup options for slave servers is as follows. Write down the options you want to use in the appropriate place in the worksheet.

- Create secure and nonsecure versions of the NIS maps.

  C2 security, C2 class of trust as defined in the *Trusted Computer System Evaluation Criteria (TCSEC)*, enables you to create secure and nonsecure versions of the NIS maps. Tru64 UNIX provides secure and nonsecure versions of the `passwd` file. For more information, see the *Security* manual and `makedbm`(8).

- Lock the `ypbind` daemon to a particular domain name and server list.

  Normally, hosts broadcast NIS requests on the network and the first available server answers the request. The –S option allows you to lock the `ypbind` daemon to a particular domain and set of servers. Requests are made directly to the specified servers, rather than being broadcast. For security purposes, you should run NIS with the –S option configured.

  If you choose to run NIS with the –S option configured, you must know the host names and IP addresses of the servers to which you are locking the `ypbind` daemon.

Security Note

> When using the `nissetup` script to set up an NIS server that is running with enhanced security, you must answer Yes to the question about locking the domain name and authorized servers (the `ypbind` –S option). For a slave server, the server is bound to itself by default and optionally to the master server and any other slave servers.

- Run NIS with the `–ypset` option, the `–ypsetme` option, or with both options set.

  The `–ypset` option allows a user running as root on any system in your domain to bind your system to a particular server. The `–ypsetme` option allows `ypbind` to accept `–ypset` requests only from the local system. For security purposes, you should not run NIS with the `–ypset` or `–ypsetme` options.

Master name

The host name of the master server in your domain.

IP address

The IP address of the master server in your domain.

7.2.2.3 Client

Setup options

The list of setup options for clients is as follows. Write down the options you want to use in the appropriate place in the worksheet.

- Create secure and nonsecure versions of the NIS maps.

C2 security, C2 class of trust as defined in the *Trusted Computer System Evaluation Criteria (TCSEC)*, enables you to create secure and nonsecure versions of the NIS maps. Tru64 UNIX provides secure and nonsecure versions of the `passwd` file. For more information, see the *Security* manual and `makedbm`(8).

- Lock the `ypbind` daemon to a particular domain name and server list.

  Normally, hosts broadcast NIS requests on the network and the first available server answers the request. The `–S` option allows you to lock the `ypbind` daemon to a particular domain and set of servers. Requests are made directly to the specified servers, rather than being broadcast. Compaq recommends that you run NIS with the `–S` option configured.

  If you choose to run NIS with the `–S` option configured, you must know the host names and IP addresses of the servers to which you are locking the `ypbind` daemon.

- Run NIS with the `–ypset` option, the `–ypsetme` option, or with both options set.

  The `–ypset` option allows a user logged in as `root` on any system in your domain to bind your system to a particular server. The `–ypsetme` option allows `ypbind` to accept `–ypset` requests only from the local system. Compaq recommends that you run NIS with neither the `–ypset` nor the `–ypsetme` options.

- Use the `automount` program.

  The `automount` program, an alternative to mounting remote file systems, allows users to mount remote file systems on an as-needed basis. When NIS is used to distribute `automount` maps, creating and administering the maps for the NIS domain is the responsibility of the administrator of the NIS master server. For information on creating `automount` maps, see Appendix C. For information on administering `automount` maps, see Section 8.1.2.

  Whether or not you use the `automount` program depends on your site's networking environment.

Server name

The name of a slave server in your domain. Specify at least three servers.

7.3 Configuring NIS

To configure NIS, use the the nissetup script. You can configure a master server, slave server, or client. See nissetup(8) for more information.

To invoke nissetup, do the following:

1. Click on the Application Manager icon on the CDE front panel.
2. Double-click on the System_Admin application group icon.
3. Double-click on the Configuration application group icon.
4. Double-click on the NIS Setup application icon.

Note

For systems without graphics capabilities, you can invoke nissetup from the command line.

7.3.1 Configuring a NIS Master Server

You must configure the master NIS server before you can configure the other systems. Prior to using the nissetup script, you must log in as root and complete the following tasks:

1. Copy into the /var/yp/src directory the local /etc files that you intend to make into NIS maps for distribution. If a file is absent from the /var/yp/src directory while it is building the default NIS maps, the nissetup command issues an informational message that it could not find that particular file and continues building the maps.

   Note

   If you copied the passwd file into the /var/yp/src directory, remove the root entry from the file.

2. Optionally, create the /var/yp/src/mail.aliases file.
3. Optionally, create the /var/yp/src/netgroup file.
4. Edit the /var/yp/Makefile file.

   If you are using the NIS master server to serve the /etc/auto.master and /etc/auto.home automount maps, you must remove the comment sign (#) from the beginning of each of the following lines. These lines were added to the Makefile for the automount daemon.

```
    ⋮
#all: passwd group hosts networks rpc services protocols netgroup \
```

```
#     aliases auto.home auto.master
   :
#$(YPDBDIR)/$(DOM)/auto.home.time: $(DIR)/auto.home
#        -@if [ -f $(DIR)/auto.home ]; then \
#              $(SED) -e "/^#/d" -e s/#.*$$// $(DIR)/auto.home | \
#              $(MAKEDBM) - $(YPDBDIR)/$(DOM)/auto.home; \
#              $(TOUCH) $(YPDBDIR)/$(DOM)/auto.home.time; \
#              $(ECHO) "updated auto.home"; \
#              if [ ! $(NOPUSH) ]; then \
#                      $(YPPUSH) auto.home; \
#                      $(ECHO) "pushed auto.home"; \
#              else \
#                      : ; \
#              fi \
#       else \
#               $(ECHO) "couldn't find $(DIR)/auto.home"; \
#       fi
#
#$(YPDBDIR)/$(DOM)/auto.master.time: $(DIR)/auto.master
#        -@if [ -f $(DIR)/auto.master ]; then \
#             $(SED) -e "/^#/d" -e s/#.*$$// $(DIR)/auto.master | \
#             $(MAKEDBM) - $(YPDBDIR)/$(DOM)/auto.master; \
#             $(TOUCH) $(YPDBDIR)/$(DOM)/auto.master.time; \
#             $(ECHO) "updated auto.master"; \
#             if [ ! $(NOPUSH) ]; then \
#                    $(YPPUSH) auto.master; \
#                    $(ECHO) "pushed auto.master"; \
#             else \
#                     : ; \
#             fi \
#       else \
#               $(ECHO) "couldn't find $(DIR)/auto.master"; \
#       fi
   :
#auto.home: $(YPDBDIR)/$(DOM)/auto.home.time
#auto.master: $(YPDBDIR)/$(DOM)/auto.master.time
   :
#$(DIR)/auto.home:
#$(DIR)/auto.master:
```

Place a comment sign (#) in front of the following lines:

```
all: passwd group hosts networks rpc services protocols netgroup \
aliases
```

If you are using the NIS master server to serve other site-specific maps, you must add entries for the maps to the `Makefile`. See the Section 7.4.7.1 for information on adding entries for site-specific NIS maps, other than the `/etc/auto.master` and `/etc/auto.home` `automount` maps, to the `/var/yp/Makefile` file.

5. Copy the `automount` maps, or any other site-specific maps, to the `/var/yp/src` directory. For information on creating `automount` maps, see Appendix C. For information on creating other site-specific maps, see the Section 7.4.7.1.

To continue to set up the master server, log in as root and run the `nissetup` script:

1. Invoke the `nissetup` script either from the CDE Desktop or by entering the following command:

   ```
   # /usr/sbin/nissetup
   ```

   A message reminds you that your network must be established before setting up NIS, and that in order to set up an NIS server you must have the Additional Networking Services subset installed.

2. Enter `c` to continue.

3. Press Return following the script's explanation of `nissetup`, and then press Return again after the script explains the three types of systems in an NIS domain.

4. Enter and confirm your system's NIS domain name.

5. Choose option 1 to indicate that you are configuring the master server.

6. Following the `nissetup` script's explanation that there can be only one master server configured for each NIS domain, enter `c` and indicate whether or not you want to run the `yppasswdd` daemon. Run the `yppasswdd` daemon on the master NIS server.

7. Enter the names of hosts that will be configured as slave servers for this domain. If you enter the name of a host that is not listed in the master server's `/etc/hosts` file, the `nissetup` script prompts you for its IP address.

   ```
   Enter the names of the SLAVE servers in the test_domain domain.
   Press Return to terminate the list.

      Host name of slave server: host2
      Host name of slave server: host3
         Cannot find host3 in the file /etc/hosts.
         To add host3 to the /etc/hosts file you MUST
                   know host3's Internet (IP) address.

     Would you like to add host3 to the /etc/hosts file
           (y/n) [y]? y

     What is host3's Internet (IP) address [no default] ?
        120.105.1.28

     Is 120.105.1.28 correct (y/n) [no default] ? y

      Hostname of slave server:  Return
   ```

   The `nissetup` script displays the list of servers that you entered and gives you the option to redo it to correct errors or to continue with the setup procedure.

   The `nissetup` script then creates the default NIS maps, displaying messages similar to the following as it does:

```
Creating default NIS maps.  Please wait...
updated passwd
updated group
updated hosts
updated networks
updated rpc
updated services
updated protocols
updated netgroup
Finished creating default NIS maps.
```

8. Indicate whether or not you want to use the –s security option.

   If you choose to the –s option, the `ypbind` process runs in a secure mode.

9. Indicate whether or not you want to use the –S security option.

   If you choose to run the –S option, you must enter the names of up to four NIS servers.

   The `nissetup` script places the host name of the server you are configuring first. Press Return when you are done entering server names.

   It is best to use the –S option.

10. Indicate whether or not you want to allow `ypset` requests on your system.

    It is best to disallow all `ypset` requests. Press Return to accept the default, and confirm your choice.

11. Indicate whether or not you want your system to use all of the NIS databases served by the master server.

    It is best to use all of the NIS databases.

    If you choose to use all of the NIS databases (either enter `y` or accept the default), the `nissetup` script edits the `/etc/svc.conf` file to include the string `yp` for each database. It also edits the `/etc/passwd` and `/etc/group` files to include a plus sign followed by a colon (`+:`) at the end of each file. This enables your system to use NIS for each database listed. This symbol enables the files to be distributed by NIS. Continue with step 15.

    If you choose not to use all of the NIS databases enter `n`, continue with the next step.

12. Indicate whether or not you want to add a plus sign followed by a colon (`+:`) to the end of the local `/etc/passwd` and `/etc/group` files.

    For your system to use the NIS served `passwd` database, `group` database, or both, `+:` must be the last line in the file or files you want served by NIS. This applies to the `passwd` and `group` databases only.

Note

The service order selection for the `passwd` and `group` databases is handled by the Security Integration Architecture (SIA). If `BSD` is selected for `passwd` and `group` information in the `/etc/sia/matrix.conf` file, only the `+:` is required for your system to search NIS.

13. Indicate whether or not you want to use NIS to obtain information for all of the default databases (other than the `/etc/passwd` and `/etc/group` which were defined in step 12).

    If you answer yes, `nissetup` edits the `svc.conf` file to include the string `yp` for each database. The `nissetup` script then skips the next question and continues at step 15.

    If you answer no, `nissetup` continues with the next question.

14. Indicate whether or not you want the `nissetup` script to invoke the `svcsetup` script. (Note, if you answered yes to step 13, skip this step.)

    If you answer yes, `nissetup` invokes the `svcsetup` script, which allows you to modify the database services selection file (the `svc.conf` file). See Section 7.3.4 for information on modifying the `svc.conf` file.

    If you answer no, `nissetup` continues with the next question. Note that you must edit the `svc.conf` file if you want your system to use NIS to obtain database information other than `passwd` and `group` information. See `svcsetup`(8) for information on editing the `svc.conf` file with `svcsetup` or manually.

15. Indicate whether or not to start the NIS daemons automatically.

    If you answer yes, `nissetup` starts the daemons.

    If you answer no, use the following command to start the daemons manually after `nissetup` exits and returns you to the system prompt (#):

    ```
    # /sbin/init.d/nis start
    ```

7.3.2 Configuring a Slave Server

To configure a slave server, do the following:

1. Invoke the `nissetup` from the CDE Desktop or by entering the following command:

   ```
   # /usr/sbin/nissetup
   ```

   A message reminds you that your network must be established before setting up NIS, and that in order to set up an NIS server you must have the Additional Networking Services subset installed.

2. Enter c to continue.
3. Press Return following the script's explanation of nissetup, and then press Return again after the script explains the three types of systems in an NIS domain.
4. Enter and confirm your system's NIS domain name.
5. Choose option 2 to indicate that you are configuring a slave server:
6. Enter c to continue following the nissetup script's explanation that the master server's list must include each slave server, and that the master server must be established in order for maps to be copied to the slave server.
7. Enter the name of the master server for your domain.
8. Indicate whether or not you want to use the –s security option.

   If you choose to the –s option, the ypbind process runs in a secure mode.
9. Indicate whether or not you want to use the –S security option.

   If you choose to run the –S option, you must enter the names of up to four NIS servers.

   The nissetup script places the host name of the server you are configuring first. Press Return when you are finished entering server names.

   It is best to use the –S option.

   If you enter the name of a host that is not listed in the slave server's /etc/hosts file, the nissetup script prompts you for its IP address. When you finish entering the list of servers, enter c to continue configuring NIS on your system.
10. Indicate whether or not you want to allow ypset requests on your system.

    It is best to disallow all ypset requests. Press Return to accept the default and confirm your choice.
11. Indicate whether or not you want your system to use all of the NIS databases served by the master server.

    It is best to use all of the NIS databases.

    If you choose to use all of the NIS databases (either enter y or accept the default), the nissetup script edits the /etc/svc.conf file to include the string yp for each database. It also edits the /etc/passwd and /etc/group files to include a plus sign followed by a colon (+:) at the end of each file. This enables your system to use NIS for each database listed. This symbol enables the file to be distributed by NIS. Continue with step 15.

If you choose not to use all of the NIS databases, enter `n`, continue with the next step.

12. Indicate whether or not you want to add `+:` to the end of the local `/etc/passwd` and `/etc/group` files.

    For your system to use the NIS-served `passwd` database, `group` database, or both, `+:` must be the last line in the file or files you want NIS to serve. This applies to the `passwd` and `group` databases only.

    Note

    The service order selection for the `passwd` and `group` databases is handled by the Security Integration Architecture (SIA). If `BSD` is selected for `passwd` and `group` information in the `/etc/sia/matrix.conf` file, the `+:` only is required for your system to search NIS.

13. Indicate whether or not you want to use NIS to obtain information for all of the default databases.

    If you answer yes, `nissetup` edits the `svc.conf` file to include the string `yp` for each database. The `nissetup` script then skips the next question and continues at step 15.

14. Indicate whether or not you want the `nissetup` script to invoke the `svcsetup` script. (Note, if you answered yes to step 13, skip this step.)

    If you answer yes, `nissetup` invokes the `svcsetup` script, which allows you to modify the database services selection file (the `svc.conf` file). See Section 7.3.4 for information on modifying the `svc.conf` file.

    If you answer no, `nissetup` continues with the next question. Note that you must edit the `svc.conf` file if you want your system to use NIS to obtain database information other than `passwd` and `group` information. See `svcsetup`(8) for information on editing the `svc.conf` file with `svcsetup` or manually.

15. Indicate whether or not to start the NIS daemons automatically.

    If you answer yes, `nissetup` starts the daemons.

    If you answer no, use the following command to start the daemons manually after `nissetup` exits and returns you to the system prompt (#):

    ```
    # /sbin/init.d/nis start
    ```

7.3.3 Configuring an NIS Client

To configure an NIS client, do the following:

1. Invoke the `nissetup` script from the CDE Desktop or by entering the following command:

   ```
   # /usr/sbin/nissetup
   ```

   A message reminds you that your network must be established before setting up NIS, and that in order to set up an NIS server you must have the Additional Networking Services subset installed.

2. Enter `c` to continue.
3. Press Return following the script's explanation of `nissetup`, and then press Return again after the script explains the three types of systems in an NIS domain.
4. Enter and confirm your system's NIS domain name.
5. Press Return to accept the default that you are configuring a client.
6. Enter `c` to continue following the `nissetup` script's warning that at least one server must be configured for this domain.
7. Indicate whether or not you want to use the `–s` security option.

   If you choose to the `–s` option, the `ypbind` process runs in a secure mode.

8. Indicate whether or not you want to use the `–S` security option.

   If you choose to run the `–S` option, you must enter the names of up to four NIS servers.

   If you enter the name of a server that is not listed in the client's `/etc/hosts` file, the `nissetup` script prompts you for its IP address. After you finish entering the list of servers, enter `c` to continue configuring NIS on your system.

9. Indicate whether or not you want to allow `ypset` requests on your system.

   It is best to disallow all `ypset` requests. Press Return to accept the default, and confirm your choice.

10. Indicate whether or not you want your system to use all of the NIS databases served by the master server.

    It is best to use all of the NIS databases.

    If you choose to use all of the NIS databases (either enter `y` or accept the default), the `nissetup` script edits the `/etc/svc.conf` file to include the string `yp` for each database. It also edits the `/etc/passwd` and `/etc/group` files to include a plus sign followed by a colon (`+:`) at the end of each file. This enables your system to use NIS for each database listed. This symbol enables the file to be distributed by NIS. Continue with step 14.

If you choose not to use all of the NIS databases, enter `n` and continue with the next step.

11. Indicate whether or not you want to add `+:` to the end of the local `/etc/passwd` and `/etc/group` files.

    For your system to use the NIS served `passwd` database, `group` database, or both, `+:` must be the last line in the file or files you want served by NIS. This applies to the `passwd` and `group` databases only.

    Note

    The service order selection for the `passwd` and `group` databases is handled by the Security Integration Architecture (SIA). If `BSD` is selected for `passwd` and `group` information in the `/etc/sia/matrix.conf` file, only the `+:` is required for your system to search NIS.

12. Indicate whether or not you want to use NIS to obtain information for all of the default databases.

    If you answer yes, `nissetup` edits the `svc.conf` file to include the string `yp` for each database. The `nissetup` script then skips the next question and continues at step 14.

    If you answer no, `nissetup` continues with the next question.

13. Indicate whether or not you want the `nissetup` script to invoke the `svcsetup` script. (Note, if you answered yes to step 12, skip this step.)

    If you answer yes, `nissetup` invokes the `svcsetup` script, which allows you to modify the database services selection file (the `svc.conf` file). See Section 7.3.4 for information on modifying the `svc.conf` file.

    If you answer no, `nissetup` continues with the next question. Note that you must edit the `svc.conf` file if you want your system to use NIS to distribute database information other than `passwd` and `group` information. See `svcsetup`(8) for information on editing the `svc.conf` file with `svcsetup` or manually.

14. Indicate whether or not to start the NIS daemons automatically.

    If you answer yes, `nissetup` starts the daemons.

    If you answer no, use the following command to start the daemon manually after `nissetup` exits and returns you to the system prompt (#):

/sbin/init.d/nis start

7.3.4 Modifying the svc.conf File with svcsetup

If you choose not to use NIS for all of the default databases, the `nissetup` script provides the option of editing the `/etc/svc.conf` file with the `svcsetup` script. If you answer yes when `nissetup` asks if you want to run `svcsetup`, it invokes the `svcsetup` script. Use the following procedure to edit the `/etc/svc.conf` file:

1. Press Return to choose the `m` option from the Configuration Menu.
2. Enter the numbers from the Change Menu that correspond to the databases whose entries you want to modify.
3. Enter the number that corresponds to the order in which you want to query the services on your system.

   If you choose the default (2), the local `/etc` files will be searched first for the requested information. If the information is not found locally, then an NIS server will be queried. This choice is valid for all of the databases that NIS serves.

   To have NIS serve `hosts` information if your system is also having `hosts` information served by DNS, choose either option 5 (`local,bind,yp`) or option 6 (`bind,local,yp`) for the `hosts` database. Note that options 3 (`local,bind`), 4 (`bind,local`), 5, and 6 are valid for the `hosts` database only.

7.3.5 Modifying or Removing an NIS Configuration

If you configure NIS and run the `nissetup` script, you can modify or remove the NIS configuration.

If you choose to modify the NIS configuration, the `nissetup` script proceeds as described in Section 7.3.1 to Section 7.3.3, resulting in a new configuration.

If you choose to remove the NIS configuration, the `nissetup` script prompts you to verify your choice, then removes the NIS information from the following files:

- `/etc/rc.config`
- `/etc/passwd`
- `/etc/group`
- `/etc/svc.conf`
- `/var/yp/`*`DOMAIN`* (where *`DOMAIN`* is the name of the current NIS domain)

This directory and its contents are deleted (for NIS master and slave servers only).

7.4 Managing an NIS Server

This section describes how to perform the following NIS server tasks:

- Add an NIS slave server to a domain
- Remove an NIS slave server from a domain
- Add a user to an NIS domain
- Update an NIS map
- Add an NIS map to a domain
- Remove an NIS map from a domain
- Modify the `/var/yp/Makefile` file
- Restrict access to NIS data

7.4.1 Adding an NIS Slave Server to a Domain

Adding a slave server to a domain enables the slave server to receive updated NIS maps from the master server and serve them to NIS clients in a domain.

To add an NIS slave server to a domain, do the following:

1. Set up the system as a slave server. See Section 7.3.2 for information on setting up a slave server.
2. Log in to the NIS master server as root.
3. Change to the `/var/yp` directory by using the `cd` command.
4. Undo the `ypservers` map and direct the output to a file by using the following command:

   \# **makedbm –u *domainname/*ypservers > *filename***
5. Edit the file and add the host name of the new server.
6. Build a new `ypservers` map by using the `makedbm` command as follows:

   \# **makedbm *filename* ypservers**

   You can combine steps 4, 5, and 6 into one command line. See the example at the end of these steps.
7. Move the `ypservers.dir` and `ypservers.pag` map files to the domain subdirectory.

8. Distribute the updated `ypservers` map to the slave servers by using the `yppush` command.

9. Edit the NIS master server's master `hosts` file and add an entry for the slave server, if it is not already in the `hosts` file. Then update the map by entering the `make` command. The `make` command also distributes the updated map.

See `makedbm`(8) for more information on building maps.

The following example (illustrating steps 3 through 9) shows how to add slave server `host8` to domain `market`:

```
# cd /var/yp
# /var/yp/makedbm -u market/ypservers ; echo host8\  [1]
 |/var/yp/makedbm - tmpmap
# mv tmpmap.dir market/ypservers.dir [2]
# mv tmpmap.pag market/ypservers.pag
# yppush ypservers [3]
# vi /var/yp/src/hosts  [4]
   .
   .
   .
# make hosts  [5]
```

[1] Represents the combination of steps 4, 5, and 6 in the preceding procedure. The output from the `makedbm` command with the `-u` option is displayed and the new server name, `host8`, is echoed on standard output to add it to the file. Then, the output is piped back into the `makedbm` command to build a new map named `tmpmap`.

Note

You can type the first and second lines as one command even if the line wraps on your screen, or you can use the backslash escape character (\), as shown.

[2] Moves the `tmpmap.dir` and `tmpmap.pag` map files to the domain `market` subdirectory and renames them as `ypservers` map files.

[3] Distributes the updated map to the slave servers.

[4] Adds a new host to the `hosts` NIS map on the master server.

[5] Updates the map and distributes the updated map to the slave servers.

Section D.1 contains a sample script you can copy that performs the steps involved in adding a slave server to a domain. You still have to set up the slave server and edit the master server's `hosts` file, adding a slave server entry, if necessary. The script does not do those steps.

7.4.2 Removing an NIS Slave Server from the Domain

Removing a slave server from a domain means that the system will no longer receive updated NIS maps from the master server and serve them to NIS clients in a domain.

To remove an NIS slave server from the domain, do the following:

1. Log in to the NIS slave server.

   If the system is going to be an NIS client, configure it as an NIS client by using `nissetup`. See Section 7.3.3 for more information.

   If the system will no longer use NIS, turn off the NIS configuration flag in the `/etc/rc.config` file by using the following command:

   ```
   # /usr/sbin/rcmgr set NIS_CONF NO
   ```

2. Log in to the NIS master server as root.
3. Change to the `/var/yp` directory by using the `cd` command.
4. Undo the `ypservers` map and direct the output to a file by using the following command:

   ```
   # makedbm -u ypservers > filename
   ```

5. Edit the file and remove the host name of the new server.
6. Build a new map by using the `makedbm` command as follows:

   ```
   # makedbm filename ypservers
   ```

   You can combine steps 4, 5, and 6 into one command line. See the following examples.

7. Move the `ypservers.dir` and `ypservers.pag` map files to the domain subdirectory.
8. Distribute the updated `ypservers` map to the slave servers by using the `yppush` command.

See `makedbm`(8) for more information on building maps.

The following example (illustrating steps 4 through 8) shows how to remove slave server `host4` from domain `market`:

```
# /var/yp/makedbm –u market/ypservers |\ [1]
 grep –v host4 | /var/yp/makedbm - tmpmap
# mv tmpmap.dir market/ypservers.dir  [2]
# mv tmpmap.pag market/ypservers.pag
# yppush ypservers [3]
```

[1] Represents the combination of steps 4, 5, and 6 in the preceding procedure. The output from the `makedbm` command with the `–u` option

is piped into `grep` with the `–v` option to display all lines except the one containing the slave server name (`host4`). Then, the output is piped back into the `makedbm` command to build a new map named `tmpmap`.

Note

You can type the first and second lines as one command even if the line wraps on your screen, or you can use the backslash escape character (\), as shown.

---

2 Moves the `tmpmap.pag` and `tmpmap.dir` map files to the domain `market` subdirectory and renames them as `ypservers` map files.

3 Distributes the updated map to the slave servers.

Section D.2 contains a sample script you can copy that performs the steps involved in removing a slave server from a domain. You still have to reconfigure the slave server as an NIS client or as a system that does not use NIS. The script does not do that for you.

7.4.3 Adding a New User to an NIS Domain

Adding a new user to an NIS domain includes the user in the `passwd` map and allows the user to participate in the NIS environment. A user has only one password on all systems that use NIS for their `passwd` map.

To add a user to an NIS domain, do the following:

1. Log in to the NIS master server as root.
2. Edit the NIS master server's master password file, `/var/yp/src/passwd`, and add an entry for the new user.

   The master `passwd` file is a readable ASCII file with a one-line entry for each valid user on the system. Here is a sample `passwd` file entry for a user named Jane Doe:

   ```
   doe:fnuTqqab.6yec:444:10:Jane Doe:/usr/staff/doe:/bin/csh
   ```

   See *System Administration* for a description of how to edit the `passwd` file to add a new user.

Note

The remote systems on the network recognize a user by the user identification (UID) number. Therefore, it is important that each user have the same UID number on each of the systems on the network.

---

3. Change to the `/var/yp` directory by using the `cd` command.
4. Update the `passwd` map by using the `make` command.
5. Create a home directory for the new user on the user's system, using the same directory name that you specified in the master `passwd` file.
6. Set up the new user's environment.

   You can define login environments for new users in several ways. For example, you can give new users a copy of the `.login` and `.cshrc` files if they use the C shell (`/bin/csh`), or the `.profile` file if they use the Bourne shell (`/bin/sh`). Copies of the default environment files are stored in the `/usr/skel` directory. See *System Administration* and `csh`(1) and `sh`(1) for further information about setting up a new user's environment.

   If the new user is a member of any groups at your site, add the user's login name to the master `group` and `netgroup` files on the NIS master server as necessary. See `group`(4), `netgroup`(4), and `groups`(1) for more information about user groups.
7. Change ownership of the directory to the new user by using the `chown` command.
8. Have the user set the NIS password by using the `yppasswd` command.

The following example (illustrating steps 2 through 4) shows how to add a new user to a domain:

```
# vi /var/yp/src/passwd  [1]
   ⋮
# cd /var/yp  [2]
# make passwd  [3]
```

[1] Opens the `/var/yp/src/passwd` file for editing.

[2] Changes to the `/var/yp` directory.

[3] Updates the NIS `passwd` map and distributes the updated map to the slave servers.

You would then set up the new user's environment and have the user set the NIS password to complete the task.

7.4.4 Updating an NIS Map

Updating an NIS map involves making changes to an NIS map's master file, updating the `Makefile` file (if the map is not listed), and building and distributing the new map. Entries for the following standard maps are included in the `Makefile` file:

- passwd
- group
- hosts
- networks
- rpc
- services
- protocols
- netgroup
- aliases

The master files are located in /var/yp/src on the NIS master server.

To update an NIS map, do the following:

1. Log in to the NIS master server as root.
2. Change to the /var/yp directory by using the cd command.
3. Modify the Makefile file, if no entry exists in the /var/yp/Makefile file for the map you want to update.

   See Section 7.4.7 for information on modifying the Makefile file.
4. Change to the /var/yp/src directory by using the cd command.
5. Edit the master file of the map you want to update and make your changes.
6. Change to the /var/yp directory by using the cd command.
7. Update and distribute the map by using the make command as follows:

   ```
   # make map_name
   ```

The following example (illustrating steps 4 through 7) shows how to update the hosts map:

```
# cd var/yp/src  1
# vi hosts   2
    ⋮
# cd /var/yp  3
# make hosts  4
```

1 Changes to the /var/yp/src directory.

2 Opens the /var/yp/src/hosts file for editing.

3 Changes to the /var/yp directory.

4 Updates the map and distributes it to the slave servers.

7.4.5 Adding an NIS Map to a Domain

Adding an NIS map to a domain allows the database information to be distributed throughout an NIS domain. You can create and distribute maps for any information you want to distribute.

To add an NIS map to a domain, do the following:

1. Log in to the NIS master server as root.
2. Create a master file for your new map.

   A master file is an ASCII text file containing individual entries. Each entry has fields separated by spaces. Some of these fields are used to build a key to each entry. Review some of the master files in the `/var/yp/src` directory to better understand the structure of a master file.
3. If you are using NIS to distribute NFS automount maps, create a file named `auto.master` in the `/var/yp/src` directory. If the file exists, add an entry for the NFS automount map you want to distribute.

   See Section 8.1.2 and Appendix C for more information on the `auto.master` map.
4. Edit `/var/yp/Makefile` file to include the new map in the default set of maps.

   See Section 7.4.7 for information on modifying the `Makefile` file.
5. Change to the `/var/yp` directory by using the `cd` command.
6. Update the map by using the `make` command as follows:

   ```
   # make map_name
   ```

The following example adds the `phonelist` map to a domain:

```
# vi /var/yp/src/phonelist  1
   ⋮
# vi /var/yp/Makefile  2
   ⋮
# cd /var/yp  3
# make phonelist  4
```

1 Creates a `phonelist` master file on the master server.

2 Modifies the `Makefile` file and adds `phonelist` entries.

3 Changes directory.

4 Updates the map and distributes the updated map to the slave servers.

7.4.6 Removing an NIS Map from a Domain

Removing an NIS map from a domain prevents the database information from being distributed throughout an NIS domain.

To remove an NIS map from a domain, do the following:

1. Log in to the NIS master server as root.
2. If you are using NIS to distribute NFS automount maps, delete the entry for the NFS map you no longer want distributed in the `auto.master` file in the `/var/yp/src` directory.

   See Section 8.1.2 and Appendix C for more information on the `auto.master` map.
3. Edit the `/var/yp/Makefile` file to remove the map from the default set of maps.

   See Section 7.4.7 for information on modifying the `Makefile` file.

7.4.7 Modifying the /var/yp/Makefile File

Modifying the `Makefile` file means adding or deleting database entries in the `/var/yp/Makefile` file on the NIS master server. By adding a database entry to the `Makefile` file, you indicate that you want a map produced for the specific database when you use the `make` command. By deleting a database entry, you indicate that you do not want a map produced for the specific database.

7.4.7.1 Adding an Entry

To add an entry to the `Makefile` file, do the following:

1. Log in to the NIS master server as root.
2. Edit the `/var/yp/Makefile` file and add the database name to the line beginning with `all:`. Next, add a line with the following format to the end of the file:

   database_name:*database_name*.time

   Finally, add an entry with the following format to the middle of the file:

   database_name.time: *various_commands*

   To simplify the creation of this entry, copy the `auto.home.time:` entry in the file and make the necessary database name changes.

3. If you are using NIS to distribute NFS automount maps, uncomment any line that contains the `auto.master` string by deleting the comment character (#) that precedes it.

The following example shows the `phonelist` database added to the `/var/yp/Makefile` file. There is a tab character preceding the `netgroup` database name in the `all:` line.

```
all: passwd group hosts networks rpc services protocols \
        netgroup aliases phonelist
    ⋮
$(YPDBDIR)/$(DOM)/phonelist.time: $(DIR)/phonelist
       -@if [-f $(DIR)/phonelist ]; then \
            $(SED) -e "/^#/d" -e s/#.*$$// $(DIR)/phonelist | \
            $(MAKEDBM) - $(YPDBDIR)/$(DOM)/phonelist; \
            $(TOUCH) $(YPDBDIR)/$(DOM)/phonelist.time; \
            $(ECHO) "updated phonelist"; \
            if [ ! $(NOPUSH) ]; then \
                    $(YPPUSH) phonelist; \
                    $(ECHO) "pushed phonelist"; \
            else \
                    : ; \
            fi \
 else \
            $(ECHO) "couldn't find $(DIR)/phonelist"; \
 fi
    ⋮
phonelist: phonelist.time
```

7.4.7.2 Deleting an Entry

To delete an entry from the `Makefile` file, do the following:

1. Log in to the NIS master server as root.
2. Edit the `/var/yp/Makefile` file, delete the database name from the line beginning with `all:`, and delete the line beginning with the database name (`database_name:`).

   Instead of deleting the database line, you could comment out the line by adding a number sign (#) to the beginning of the line.

7.4.7.3 Makefile Editing Guidelines

As you edit the `/var/yp/Makefile` file, remember the following:

- The order of entries in the line that begins with `all:` is not important. However, in continuation lines, the blank space preceding the line must be a tab character; do not use spaces.

• Variables are defined at the top of the `Makefile` file.

7.4.8 Restricting Access to NIS Data

By default, the `ypserv` and `ypxfrd` daemons provide NIS information to anyone with network access to an NIS server who makes a request. However, you can restrict NIS database access to only those hosts in subnets you specify by completing the following steps:

1. Log in to the NIS server as root.
2. Create a `/var/yp/securenets` file.
3. Edit the `/var/yp/securenets` file and add an entry for each subnet from which the NIS server is to accept NIS requests. The format of each file entry is as follows:

   subnet_mask subnet_ip_address

   For example:

   ```
   255.255.0.0  128.30.0.0      1
   255.255.255.0 128.211.10.0    2
   255.255.255.255 128.211.5.6   3
   ```

   1 Allows IP addresses that are within the subnet `128.30` range to access the NIS files. The network mask is `255.255.0.0` and the corresponding network address is `128.30.0.0`.

   2 Allows IP addresses that are within the subnet `128.211.10` range to access the NIS files.

   3 Allows one host with the IP address `128.211.5.6` to access the NIS files.

4. Save the file.

If the file does not exist or contains no entries, the server accepts any NIS request.

If the file exists and contains entries, the `ypserv` and `ypxfrd` daemons read the `/var/yp/securenets` file during initialization. When an NIS request is received, the requester's IP address is compared to the subnets in the `/var/yp/securenets` file. If it matches, the request is processed. If it does not match, the NIS request is rejected and the rejection is recorded in the NIS server's log file. For example:

```
ypxfrd: An attack by non-trusted host, 128.40.16.122
```

On the system making the NIS request, NIS commands such as `ypcat` terminate with no error message. If a user is trying to log in to a system, the login times out after many retries.

Note

If the /var/yp/securenets file is modified, the you must kill and restart the ypserv and ypbind daemons.

You can also use a /var/yp/securenets file to restrict access to NIS data on a slave server. However, the NIS slave server's IP address must be in the authorization range of entries in the /var/yp/securenets file of the NIS master.

7.5 Managing an NIS Client

This section describes how to perform the following NIS client management tasks:

- Change an NIS password
- Obtain NIS map information

7.5.1 Changing an NIS Password

To change a user's password that is stored in the NIS passwd map, use the yppasswd command. If you receive an error message, ask the system administrator on the master server to verify that the rpc.yppasswdd daemon on the NIS master server is running.

If you try to change your password with the passwd command, you might receive the following error message:

```
Not in passwd file.
```

This message means your password is stored and distributed in NIS. You must change your password by using the yppasswd command.

To change the root password, use the passwd command. This password is local and not in the NIS file.

See yppasswd(8) and rpc.yppasswdd(8) for further information.

7.5.2 Obtaining NIS Map Information

Obtaining NIS map information enables you to see the following:

- Map names
- Map values
- Map keys
- Map master server

To obtain NIS map information, issue one of the commands listed in Table 7–1.

Table 7–1: NIS Map Information Commands

Command	Action
`ypcat`	Prints values from an NIS database
`ypwhich`	Prints the name of the master server for an NIS map
`ypmatch`	Prints the values of one or more keys from an NIS map

Use the `–x` option with any of the commands shown in Table 7–1 to list all the map nicknames.

See `ypcat`(1), `ypwhich`(1), and `ypmatch`(1) for more information about these commands.

The following command lists all available maps and their master servers:

```
# ypwhich –m
```

The following command lists all values in the `hosts` map:

```
# ypcat hosts
```

The following command lists all occurrences in the `hosts` map that have the key `apple`:

```
# ypmatch apple hosts
```

The following command lists all occurrences in the `hosts` map that have the name `jones` associated with them. The name `jones` is not a key in this map.

```
# ypcat hosts | grep jones
```

8
Network File System

The Network File System (NFS) is a facility for sharing files in a heterogeneous environment. This chapter describes the NFS environment, how to plan for NFS, how to configure your system for NFS, and how to manage NFS servers and clients, including how to export and import file systems.

For introductory information on NFS, see `nfs_intro`(7).

8.1 NFS Environment

In the NFS environment, systems can have the following roles:

- Client — A system that imports file systems. A client can mount file systems by using either the `/etc/fstab` file or the `automount` daemon. Both methods are explained in this chapter.
- Server — A system that exports file systems.

Your system can be set up as an NFS server, an NFS client, or both.

8.1.1 Distributing the hosts Database

If your network is running NIS or Berkeley Internet Name Domain (BIND) to distribute host information, you do not need to list each server that is referenced in a client's `/etc/fstab` file in the client's local `/etc/hosts` file. However, the server's host information must be in the NIS or BIND database.

Similarly, if your network is running NIS or BIND to distribute host information and the client information is listed in the `hosts` database, you do not have to list each client that is referenced in a server's `/etc/exports` file in the server's local `/etc/hosts` file.

8.1.2 Automount and NFS

The `automount` daemon offers an alternative to mounting remote file systems with the `/etc/fstab` file, allowing you to mount them on an as-needed basis.

When a user on a system using the `automount` daemon invokes a command that must access a remotely mounted file or directory, the `automount` daemon mounts that file system or directory and keeps it mounted for as long as the user needs it. When a specified amount of time elapses (the default is 5 minutes) without the file system or directory being accessed, the `automount` daemon unmounts it.

You specify the file systems to be mounted in automount maps. These maps may be customized to suit your environment and administered in the following ways:

- Use NIS to create and distribute the automount maps.
- Administer the automount maps locally.
- Use a combination of both methods.

See Appendix C for information on writing automount maps.

8.1.2.1 NIS and automount Maps

NIS allows you to create and distribute customized maps and, typically, is used to distribute automount maps. Therefore, if NIS is used on your network to distribute automount maps, your system must be an NIS client. When NIS is used to distribute automount maps, the administrator of the NIS master server creates and administers the maps for the NIS domain.

If many clients in an environment remotely mount a file system by specifying it in their `/etc/fstab` file, that file system is a good candidate for inclusion in a map distributed by NIS. Carefully constructed automount maps can allow client systems to eliminate a large part of their `/etc/fstab` files. If the location of a file system that is included in a distributed automount map changes, or its server changes, the administrator of automount maps changes the map on the NIS master server. The change is then propagated throughout the domain without users on the client systems having to edit their `/etc/fstab` files.

See Section 7.3.1 for information on configuring a master NIS server to serve automount maps.

8.1.2.2 Local automount Maps

Local automount maps might be useful to you under the following circumstances:

- Your system mounts remote file systems that are not typically mounted by other NIS clients.
- Your network is not running NIS.

• You need to test an automount map.

Administering the `automount` daemon locally is the same as administering it when NIS distributes the maps, except that you, as administrator of your system, create and manage automount maps.

A local `auto.master` map serves the same function as one distributed in an NIS domain. If a local `auto.master` is specified, the `automount` daemon consults it for the location of other maps, their local mount points, and the mount options. You can use an `auto.master` map that is distributed by NIS, a local `auto.master` map, both, or neither, if the `automount` daemon is invoked correctly.

8.2 Planning NFS

Appendix A contains a worksheet that you can use to record the information that you need to provide to configure NFS. If you are viewing this manual online, you can use the print feature to print a copy of this part of the worksheet.

Figure 8–1 shows Part 7 of the Configuration Worksheet. The following sections explain the information you need to record in Part 7 of the worksheet.

Figure 8–1: Configuration Worksheet, Part 7

Part 7: NFS Setup

Server

Number of nfsd threads: TCP: ________ UDP: ________
NFS locking: ☐ Yes ☐ No
PCNFS daemon: ☐ Yes ☐ No
Allow nonroot mounts: ☐ Yes ☐ No

Path name:	Permissions:	Network group/ Node name:

Client

Number of I/O threads: ________
NFS locking: ☐ Yes ☐ No
Automount: ☐ Yes ☐ No
Remote server name: ________ ________
Directory path: ________ ________
Local mount point: ________ ________
Readonly mount: ☐ Yes ☐ No ☐ Yes ☐ No

8.2.1 Server

Number of nfsd TCP server threads

Enter the number of `nfsd` TCP server threads to run. These threads service requests from NFS clients. The default number of 8 is adequate for an average work load. You can configure a combined total of 0 to 128 TCP and UDP server threads. See `nfsd`(8) for information on starting the `nfsd` daemon from the command line.

Number of nfsd UDP server threads

Enter the number of UDP server threads to run. The default number of 8 is adequate for an average work load. You can configure a combined total of 0 to 128 TCP and UDP server threads. See `nfsd`(8) for information on starting the `nfsd` daemon from the command line.

NFS locking

If you want to run the NFS lock manager (`rpc.lockd`) and status monitor (`rpc.statd`), check Yes. Running these daemons allows users to use `fcntl`(2) and `lockf`(3) to lock file regions on NFS files (in addition to local files). If you do not run these daemons, users can use advisory locking primitives only on local files.

PCNFS daemon

If you want to run the PC-NFS daemon (`rpc.pcnfsd`), check Yes; otherwise, check No.

Allow nonroot mounts

If you allow nonroot mounts, users on client systems who do not have root privileges can still mount the file systems or directories exported from this system. If you do not allow nonroot mounts, only the superusers on the client systems can mount file systems from this host. The default setting does not allow nonroot mounts.

Path name

The path name of the file systems or directories that you intend to export.

Permissions

The permissions to assign for each exported file system or directory. You can specify whether a file system or directory is exported with read-write (rw) or read-only (ro) permission, and you can map client superuser access to a root user ID (UID) number other than the default of –2. For more information on assigning permissions to exported file systems or directories and on specifically mapping the root UID for clients, see exports(4).

Network group/Node name

The network groups or individual host names to which you will export these file systems or directories. If you want to limit the hosts that can import a file system or directory, you must explicitly specify the individual hosts or network groups in the /etc/exports file. If you do not specify individual hosts or network groups, all hosts can import that file system or directory. For information on defining network groups, see netgroup(4).

8.2.2 Client

Number of I/O threads

The number of I/O threads to run. The default number of 7 is recommended for optimum load generation on servers. You can configure from 0 to 64 nfsiod threads.

In addition, you can start nfsiod threads from the command line. See nfsiod(8) for information on starting nfsiod threads from the command line.

NFS locking

If you want to run the NFS lock manager (rpc.lockd) and status monitor (rpc.statd), check Yes. Running these daemons allows users to use fcntl(2) and lockf(3) to lock file regions on NFS files (in addition to local files). If you do not run these daemons, users can use advisory locking primitives only on local files.

Automount

If the client is to run the automount daemon and use automount maps, check Yes. If the network is running the Network Information Service (NIS), the automount maps are better administered and served from the master NIS server. The format of the maps is the same whether they are local or served by the NIS master server. For information on creating automount maps, see Appendix C.

Otherwise, check No.

Remote server name

The host names of the servers from which you are importing file systems or directories.

Directory path

The complete pathnames of the file systems or directories that you want to import.

Local mount point

The mount point on the local system where you want the imported file systems or directories to reside.

Read-only mount

The permissions for the imported file systems or directories

Note

If you mount your user area from a server, make sure that your UID on the client is the same as your UID on the server. NFS uses your client UID to check against file access permissions on the server. If your UID is different on the client and server, you cannot modify your own NFS mounted files (assuming that you have the permissions on the mounted files set so that only you can modify them). Since the server does the access checking, the only UID allowed to modify the files is the one that the server knows.

8.3 Configuring NFS

Use the NFS Configuration application of the Common Desktop Environment (CDE) Application Manager for configuring NFS on systems with graphics capabilities. You can configure clients, servers, and designate imported and exported filesystems.

See `nfsconfig`(8X) for more information on the NFS Configuration Application.

To invoke the NFS Configuration application, log in as root and do the following:

1. Click on the Application Manager icon on the CDE front panel.
2. Double-click on the System_Admin application group icon.
3. Double-click on the Configuration application group icon.

4. Double-click on the NFS Configuration application icon. The NFS Configuration main window is displayed, showing available NFS service types and configured NFS service types.

To exit the NFS Configuration application, choose File then Exit.

Note

For systems without graphics capabilities, you can use the `nfssetup` utility. See `nfssetup`(8) for more information.

The NFS Configuration application has an extensive online help system that you can use instead of the instructions in this section to configure NFS on your system.

8.3.1 Configuring an NFS Server

To configure an NFS server, do the following:

1. Select NFS Server Setup from the Available NFS Services list box in the NFS Configuration main window.
2. Click on the Configure button to display the NFS Server Setup dialog box.
3. Enter the number of server TCP daemons to be run in the input text field, or use the Increment or Decrement buttons to specify the appropriate number of server TCP daemons.
4. Enter the number of server UDP daemons to be run in the input text field, or use the Increment or Decrement buttons to specify the appropriate number of server UDP daemons.
5. Set the Configure for Locking check button to the on position to specify locking configuration if the status of the `lockd` daemon is Stopped. If the status of the daemon is Running, locking is already set.
6. Click on the Configure PC NFS check button to run the PC-NFS `rpc.pcnfsd` daemon.

   If you run the PC-NFS daemon, you must export to the client the directories you want to mount on the PC client. Also, you must export the `/usr/spool/pcnfs` directory to the PC client to enable the client to utilize network printing. For information on exporting directories, see Section 8.4.1.
7. Set the Allow Non-Root Mounts check button to the on position to allow file systems to be mounted by users other than root if the status

of the daemon is Stopped. If the status of the daemon is Running, mounting by non-root users is already set.

8. Click on Commit to accept the configuration and start the appropriate daemons.
9. Click on Close to close the NFS Server Setup dialog box.

If your system will be an NFS client, see Section 8.3.2 for information on configuring an NFS client. If your system will export or import directories, go to Section 8.4.1 or Section 8.5.1, respectively.

8.3.2 Configuring an NFS Client

To configure an NFS client, do the following:

1. Select NFS Client Setup from the Available NFS Services list box in the NFS Configuration main window.
2. Click on the Configure button to display the NFS Client Setup dialog box.
3. Enter the number of client daemons to be run in the input text field, or use the Increment or Decrement buttons to specify the appropriate number of client daemons.
4. Set the Configure for Locking check button to the on position to specify locking configuration if the status of the `lockd` daemon is Stopped. If the status of the daemon is Running, locking is already set.
5. Set the Configure for Automount check button to the on position to configure the `automountd` daemon if the status of the daemon is Stopped. If the status of the daemon is Running, automounting is already configured. See Section 8.1.2 for information on automount and Appendix C for information on automount maps.
6. Enter appropriate arguments to the `automountd` daemon in the Automount arguments input text field.

   You can later change the `automount` daemon argument list by using the `rcmgr` command to set the `AUTOMOUNT_ARGS` variable.

   For more information, see `automount`(8) and `rcmgr`(8).
7. Click on Commit to accept the configuration, start the appropriate daemons, and update the status of the daemons.
8. Click on Close to close the NFS Client Setup dialog box.

If you want to import directories, go to Section 8.5.1.

8.4 Managing an NFS Server

This section describes how to perform the following NFS server tasks:

- Export a directory or file system
- Halt export of a directory or file system
- Enable a superuser on a client system to access files as root
- Send mail to superuser (root) across NFS
- Enable port monitoring
- Monitor the NFS load

You might have to reconfigure NFS on your system, whether to make a client system a server system or to increase the number of NFS threads. See Section 8.2 for this information.

8.4.1 Exporting a File System or Directory

Exporting a file system or directory makes it available for client systems on the network to mount remotely. If you want your system to be an NFS server and to export file systems and directories, be aware that your system will be less secure. However, depending on how you export your files, you can minimize the security risks.

Note

To make a change to the exports file and have it take effect immediately, send the `mountd` daemon a HUP signal. Otherwise, `mountd` will reread the `exports` file the next time it receives a mount request from an NFS client or a `showmount -e` request.

8.4.1.1 Exporting a File System with the NFS Configuration Application

To export a file system using the NFS Configuration Application, do the following:

1. Log in as root.
2. Click on the File Sharing button in the NFS Configuration main window to display the File System Sharing main window.
3. Select FileShare then Share Local File to display the Share Local File dialog box.
4. Enter the full pathname of the directory to be exported in the Directory input text field.

5. Select whether the file has read only or read/write access and whether all hosts or only selected hosts can have access. By default, the file is exported read/write to all hosts.
6. Click on the down arrow icon to display the expert options. By default, no root access is allowed and the anonymous UID is 2.
7. Click on Apply. Repeat steps 3, 4, and 5 if you want to export additional files.
8. Click on OK to close the Share Local File dialog box.
9. Select File Sharing then Exit to close the File System Sharing main window.

8.4.1.2 Exporting a File System Manually

To export a file system or directory, do the following:

1. Edit the `/etc/exports` file on your system and create an entry for the file system or directory to be exported. The following example shows entries from a sample `/etc/exports` file:

```
/usr/local    1
/usr/staff/doe host3     2
/usr/staff –ro host7     3
/usr2 host7 host3 host1     4
/usr/scratch –rw=host2    5
/usr/src –rw=host1:host2 host5 host7    6
```

1	Exports the `/usr/local` file system. It can be mounted remotely (read-write) by any NFS client on the network.
2	Exports the `/usr/staff/doe` subdirectory. It can be mounted remotely (read-write) only by host3.
3	Exports the `/usr/staff` file system. It can be mounted remotely (read-only) only by host7. Client host7 also has read-only access to `/usr/staff/doe` exported in the second entry.
4	Exports the `/usr2` directory. It can be mounted remotely (read-write) only by host7, host3, and host1.
5	Exports the `/usr/scratch` file system to all hosts. Only host2 is allowed read-write access.
6	Exports the `/usr/src` file system to host1, host2, host5, and host7. Only host1 and host2 are allowed read-write access.

   See `exports`(4) for more information on the `/etc/exports` file.

2. Check that the NFS server daemons `mountd`, `portmap`, and `nfsd` are running, using the `ps` command as follows:

```
# ps -e | grep daemon_name
```

If they are running, go to the next step. If they are not running, start them by using the following commands:

```
# /sbin/init.d/nfs start
# /sbin/init.d/nfsmount start
```

3. Verify the exported files by using the `showmount -e` command.

   The file system or directory is exported automatically when a mount request is received.

NFS servers use the standard operating system file access protection scheme. This scheme protects files from all users except root. An NFS client sends user and group IDs to the server along with an NFS file access request. The server uses this information to allow or disallow the request.

8.4.1.3 Export Guidelines

The `/etc/exports` file defines an export list for each file system and directory that a client can mount. When creating entries in the `/etc/exports` file, remember the following:

- Make only one entry for each exported file system or directory; multiple entries are not supported.
- Each entry exports that directory and all subdirectories in it, except for those subdirectories that reside in a file system (disk partition) different from the exported directory.
- File systems and directories are exported with read-write access by default.
- If no remote system (client) names are specified for a file system or directory, any client on the network can mount that file system or directory.
- If one or more client names are specified for a file system or directory, only those clients can mount the exported file system or directory.
- If you start the `mountd` daemon with the `-i` option, only those hosts in the server's host database are allowed mount access. If you start the `mountd` daemon with the `-d` or `-s` option, only those clients in the same domain or subdomain, respectively, are allowed mount access.
- Exporting specific directories to specific clients provides more security than does exporting an entire file system to all clients.
- Protect sensitive exported data on the server by making the data files owned and accessible only by root, and do not allow superusers on client systems root access over NFS.

- The -public option can only be specified by one exported file system.

8.4.2 Halting Export of a Directory or File System

Halting export of a directory or file system prevents client systems from accessing the particular directory or file system. You can still export other directories or file systems.

To halt the export of a directory or file system, do the following:

1. Delete from the /etc/exports file the entry for the directory or file system you do not want to export.
2. Verify that the entry is no longer in the exports list by using the following command:

   ```
   # showmount –e
   ```

3. If you do not want to export any directories or file systems, stop the nfsd daemon by using the following commands:

   ```
   # ps –e | grep nfsd
   # kill –9 process_id
   ```

8.4.3 Enabling Client Superuser Access to Files

By default under NFS, a superuser (root) on a client system does not have superuser privileges on the server and cannot do the following:

- Access remotely mounted files and directories whose permissions do not allow world access.
- Change the ownership of remotely mounted files (run the chown command).

For security reasons, you typically should not allow a remote superuser access to your system as superuser unless both the remote host and superuser are trusted. However, in a friendly network environment, you can explicitly allow superuser access over the network.

To allow a superuser on a client access to your server system, edit the /etc/exports file on your server and add the –root=0 option to the entry you want to make available. The –root=0 option maps the remote superuser's identification to UID 0. All future mount requests will be honored with root mapping. By default, this option allows superuser access from any client system on the network. To restrict the superuser access to specific systems, use the –root=*host_list* option, where *host_list* is a list of host names. See exports(4) for more information.

By default, NFS servers regard superusers and those users without UNIX authentication (personal computer systems) as anonymous users. This class of users can only access files that are accessible to the world. To prevent anonymous users from accessing file systems or directories, use the –anon=–1 option. If you still want to allow client superusers access to the file systems or directories, specify the –root option in addition to the –anon option. The –root option overrides the –anon option for client superusers only.

A superuser on a client system can assume the identity of any other user on the client system by substituting the UID number. The client superuser could then have the access rights of another user on the server. Therefore, to protect sensitive exported data on the server, make root the owner of the data files and do not export the directory or file system with root mapping. This is useful if you need to export other files in the file system.

The following example shows entries in an /etc/exports file:

```
/usr/games –root=0 host8      1
/usr/templates –root=host8    2
```

1 Exports the /usr/games file system. It can be mounted remotely (read-write) only by the client system host8. However, the client superuser has superuser access to the file system. The superuser's UID is 0 (zero).

2 Exports the /usr/templates file system. It can be mounted remotely (read-write) by any client in the network. However, only the superuser on host8 has superuser access to the file system.

8.4.4 Sending Mail to Superuser (root) Across NFS

If the /usr/spool/mail directory is remotely mounted from the server, you might not be able to send mail to superuser (root) on the server. The reason is most systems do not export the /usr/spool/mail directory with the root=0 option. To enable clients to send mail to root, set the root and admin aliases to the login name or names of the system administrators for that system. Then, users can address all mail intended for the administrators of that system as follows:

admin@*system*

To enable clients to send mail to root, follow these steps:

1. Edit the /var/adm/sendmail.cf file and add the alias name admin to the following line:

   ```
   CN MAILER-DAEMON postmaster
   ```

The line should then look as follows:

```
CN MAILER-DAEMON postmaster admin
```

This adds the name admin to the class N.

Alternatively, you can run the Mail Configuration application and add admin as a local user. See Chapter 11 for more information.

2. Edit the `/var/adm/sendmail/aliases` file, add the login names of the system administrators, and redefine (alias) the name root to be admin.

3. Restart the `sendmail` daemon by using the following command:

   ```
   # /sbin/init.d/sendmail restart
   ```

If you are enabling clients to send mail to root, remember the following:

- All systems in the local area network (LAN) should follow this convention. Mail for root or admin on any system can be automatically directed to any user login on any system.
- A `/usr/spool/mail/root` mailbox is not created or used.

The following example shows the steps involved in enabling clients to send mail to root.

```
# vi /var/adm/sendmail/sendmail.cf  1
   ⋮
# vi /var/adm/sendmail/aliases  2
   ⋮
# /sbin/init.d/sendmail restart  3
```

1 Opens the `/var/adm/sendmail/sendmail.cf` file to add the admin alias.

2 Opens the `/var/adm/sendmail/aliases` file to add the login names and root alias.

3 Restarts the `sendmail` daemon.

The following example shows entries in the `/var/adm/sendmail/aliases` file for the system administrators John, Mary, and Joe:

```
admin:john,mary,joe
root:admin
```

8.4.5 Enabling Port Monitoring

Only privileged users can attach to Internet domain source ports known as privileged ports. By default, NFS does not check to see if a client is bound to a privileged port. You might want to activate NFS server port monitoring to be sure that file access requests were generated by the client kernel rather than forged by an application program.

Although this operating system enforces the privileged port convention, some operating systems do not. If hosts running a different operating system are on your network, activating port checking might not improve security, but could prevent those systems from functioning properly as NFS client systems.

To start NFS server port monitoring, enter the following command:

```
# /usr/sbin/nfsportmon on
```

To stop source port monitoring, enter the following command:

```
# /usr/sbin/nfsportmon off
```

8.4.6 Monitoring the NFS Load

Monitoring the NFS load allows you to see the number of NFS requests, both client and server, being executed on the local machine. You should periodically monitor NFS requests to determine whether you need additional NFS server threads.

To monitor NFS requests, use the `nfsstat` command with the following syntax:

nfsstat –n

See `nfsstat`(8) for more information on monitoring NFS load.

The following example shows the client and server activity on a local machine:

```
# /usr/bin/nfsstat -n
 nfs:
calls      badcalls
69228      0

Server nfs V2:
null       getattr    setattr    root       lookup     readlink   read
1  0%      24 0%      0  0%      0  0%      60  0%     0  0%      5  0%
wrcache    write      create     remove     rename     link       symlink
0  0%      58030 83%  20  0%     0  0%      0  0%      0  0%      0  0%
mkdir      rmdir      readdir    statfs
0  0%      0  0%      0  0%      2  0%

Server nfs V3:
null       getattr    setattr    lookup     access     readlink   read
0  0%      667  0%    1009  1%   2598  3%   101  0%    200  0%    1408  2%
write      create     mkdir      symlink    mknod      remove     rmdir
1280  1%   376  0%    71  0%     200  0%    0  0%      676  0%    70  0%
rename     link       readdir    readdir+   fsstat     fsinfo     pathconf
100  0%    100  0%    468  0%    0  0%      1750  2%   2  0%      0  0%
commit
```

```
10  0%

Client nfs:
calls      badcalls   nclget     nclsleep
224664     0          224664     0

Client nfs V2:
null       getattr    setattr    root       lookup     readlink   read
0  0%      51328 22%  1069  0%   0  0%      41643 18%  455  0%    28793  12%
wrcache    write      create     remove     rename     link       symlink
0  0%      64665 28%  589  0%    1052  0%   352  0%    250  0%    250  0%
mkdir      rmdir      readdir    statfs
171  0%    170  0%    2689  1%   1814  0%

Client nfs V3:
null       getattr    setattr    lookup     access     readlink   read
0  0%      2038  0%   2180  0%   8534  3%   430  0%    450  0%    3136  1%
write      create     mkdir      symlink    mknod      remove     rmdir
3158  1%   1048  0%   243  0%    450  0%    1  0%      1848  0%   242  0%
rename     link       readdir    readdir+   fsstat     fsinfo     pathconf
452  0%    350  0%    1240  0%   0  0%      3506  1%   3  0%      0  0%
commit
75  0%
```

8.5 Managing an NFS Client

Your system can be an NFS client if the following conditions exist:

- Your system can reach an NFS server over the network.
- Your system's host or network group name is included in the server's `/etc/exports` file, or the server is exporting a file system to all systems on the network.

This section describes how to perform the following NFS client tasks:

- Mount a remote file system or directory
- Unmount a remote file system or directory

8.5.1 Mounting a Remote File System or Directory

You can mount a remote file system or any subdirectory within a remote file system onto a local mount point. While mounted, it is treated as a file system by the local system. The file system or subdirectory must also be entered in the remote system's `/etc/exports` file.

8.5.1.1 Mounting a Remote File System Using NFS Configuration Application

To mount a remote file system or directory on a system with graphics capabilities, use the NFS Configuration Application as follows:

1. Click on the File Sharing button in the NFS Configuration main window to display the File System Sharing main window.

2. Select FileShare then Share Remote File to display the Share Remote File dialog box.
3. Enter the full pathname of the directory in the Get Directory input text box.
4. Enter the host name of the system from which you are importing the directory in the From Host input text box.
5. Enter the local mount point in the Put Files In input text box. If the local mount point does not exist, one is created by default.

Note

Place mount points to different servers in separate directory trees. Some directories (such as `/usr`) in complex production environments might be too large for you to adhere strictly to this recommendation. In such cases, try to minimize the number of mount points to different servers that occur in any given directory.

6. If you want to mount this file at each time your system starts, click on Make Permanent. This creates an entry in the `/etc/fstab` file.
7. If you want to select NFS options, click on the down arrow icon. By default, files are imported with the following options: read-write, hard, retry in foreground, and noninterruptable. See Section 8.5.1.4 for list of some options.
8. Click on Apply. If you want to import additional files, go to step 3 and repeat the succeeding steps.
9. Click on OK to close the Share Remote File dialog box.
10. Select File Sharing then Exit to close the File System Sharing main window.

See `nfsconfig`(8X) for more information.

8.5.1.2 Mounting a Remote File System Manually

To manually mount a remote file system or directory, do the following:

1. Create a directory (mount point) on the local system.

   Typically, people create a directory with the same name as the remote host because it is easier to remember where the remotely mounted file systems and directories reside.
2. Mount the remote file system or directory by using either of the two following mount command formats:

mount –t nfs [*server_name*:/ *filesystem* / *mount_point*]

mount –t nfs [*filesystem*@ *server_name* / *mount_point*]

The following example mounts the reference pages from the remote host host2 onto the local directory `/mnt`:

```
# mount –t nfs host2:/usr/ref /mnt
```

3. Verify that the file system or directory is mounted by entering the `mount` command with no arguments. The mounted file systems and directories are displayed as in the following example:

```
# /usr/sbin/mount
/dev/ra0a on / (rw)
/dev/ra0g on /usr (rw)
host2:/usr/ref on /mnt type nfs (rw, hard, intr)
host7:/usr on /host7 type nfs (rw, hard, nintr)
```

4. If you are mounting a remote layered product for the first time, create the necessary symbolic links by executing the appropriate linking script or scripts. Ask the server administrator for the location of the linking scripts and the command syntax to use to invoke them.

   Use this step for Compaq layered products and third-party layered products that have been created in accordance with Compaq guidelines. See *Programming Support Tools* for information on creating linking scripts for layered products.

8.5.1.3 Mounting a Remote File System Automatically

To automatically mount a remote file system or directory at startup time, do the following:

1. Log in as root.
2. Edit the `/etc/fstab` file and create an entry for each file system or directory to be mounted. For example, the following entry in the `/etc/fstab` file causes the `/usr` file system on the remote host host7 to be mounted automatically at startup time on the local system in the `/host7` directory:

```
/usr@host7 /host7 nfs rw,bg 0 0
```

   The `bg` option causes remote mount requests to be tried once in the foreground and then retried in the background if the initial mount fails. See Section 8.5.1.4 for a list of some of the options. See `fstab`(4) for information on the `/etc/fstab` file format.

3. Mount the new directory or file system by entering the `mount –a` command.

The files will be mounted automatically each time the system is rebooted.

4. If you are mounting a remote layered product for the first time, create the necessary symbolic links by executing the appropriate linking script or scripts. Ask the server administrator for the location of the linking scripts and the command syntax to use to invoke them.

   Use this step for Compaq layered products and third-party layered products that have been created in accordance with Compaq guidelines. See *Programming Support Tools* for information on creating linking scripts for layered products.

8.5.1.4 Mount Command Options

Occasionally, a server system will go down or be slow to respond to client NFS requests; when you mount the file system, choose one of the following mount command options to control how NFS operations are to proceed under those conditions:

- `bg` — Remote mount requests are tried once in the foreground. If they fail, the requests are then retried in the background. The default is to retry remote mount requests in the foreground. If any server listed in the `/etc/fstab` file is not currently available, your system will not finish booting until the server becomes available.
- `soft` — Operations fail with an error code (ETIMEDOUT). Do not use this option with file systems mounted as read-write or when running executable files, unless you are sure the application is testing return codes.
- `hard` — Operations do not fail: they continue to try until they either succeed or are stopped. This is the default.

  When you use the interrupt option, `intr`, with the `hard` option, you can type an interrupt character and prevent your system from indefinitely attempting to reach an unreachable server system. The `intr` option is the default with the `hard` option.

See `mount`(8) for further information on `mount` command options.

8.5.1.5 Using automount to Mount a Remote File System

The `automount` daemon allows you to automatically mount a remote file system or directory at the time of access. If you are using `automount`, determine whether you are using local automount maps or NIS-distributed automount maps. See Section 8.1.2 for a description of local and NIS-distributed automount maps.

To use local automount maps, do the following:

1. Log in as root.
2. Create a local `auto.master` map in the `/etc` directory. See Appendix C for information on creating automount maps.

Note

If you are modifying an existing `auto.master` map, you must stop and restart the `automount` daemon in order to read the revised map.

3. Create the local maps for your system.
4. Start the `automount` daemon by using the NFS Configuration Application. See Section 8.3.2 for information on starting the `automount` daemon.

   When the `automount` daemon starts, it uses the local `auto.master` file to determine the location of other maps, their local mount points, and the mount options.

   If the NFS Configuration Application indicates that the `automount` daemon is already running, do the following:

   a. Set the Configure for Automount check box to the off position.
   b. Select Commit.
   c. Set the Configure for Automount check box to the on position.
   d. Select Commit.
   e. Select Close to close the NFS Client Setup dialog box.

To use NIS-distributed automount maps, do the following:

1. Set up your system as an NIS client. See Section 7.3.3 for information on setting up an NIS client.
2. Start the `automount` daemon by using the NFS Configuration Application. See Section 8.3.2 for information on starting the `automount` daemon.

   All automount maps are served from the NIS master server in the domain. When the `automount` daemon starts, it uses the master `auto.master` file to determine the location of other maps, their local mount points, and the mount options.

If the NFS Configuration Application indicates that the automount daemon is already running, do the following:

a. Set the Configure for Automount check box to the off position.

b. Select Commit.

c. Set the Configure for Automount check box to the on position.

d. Select Commit.

e. Select Close to close the NFS Client Setup dialog box.

See automount(8) for information on the automount command and its arguments.

8.5.1.6 Specifying automount Arguments

You can specify arguments for the automount daemon from the command line, in a local auto.master map, in an NIS-distributed auto.master map, or some combination of the three. However, it is important to know that the automount daemon reads and carries out its instructions in the following order:

1. Command line information, such as additional mount points or replacements to entries in a master map, are read first. Command line information takes precedence over instructions in any maps — local or NIS-distributed.
2. Instructions in a local auto.master map (specified with the –f option) are read next. The information in the local master map overrides information in an NIS-distributed master map.
3. Information in the NIS-distributed master map is read last.

When you invoke the automount daemon without any options, it looks for a distributed NIS map called auto.master. If it finds one, it checks the master map for information about the location of other maps, their local mount points, and the mount options. If it does not find one, and if no local auto.master is specified, the automount daemon exits.

You can pass command arguments to the automount daemon from the NFS Configuration Application, the command line, or from an entry in the /etc/rc.config file in one of the following ways:

- Specify all of the arguments to the automount command on the command line. For example:

```
# automount /net -hosts \
  /home /etc/auto.home -rw,intr \
  /- /etc/auto.direct -ro,intr
```

- Include the information in the previous example in an NIS-distributed `auto.master` map:

  ```
  /net    –hosts
  /home   /etc/auto.home          –rw,intr
  /-      /etc/auto.direct        –ro,intr
  ```

  If this NIS `auto.master` map is distributed, typing the `automount` command at the superuser prompt (#) produces the same results as the previous command line.

- Include the `automount` command information in a local `auto.master` file and use the `–f` option to instruct the `automount` daemon to consult the local `auto.master` file first for instructions. The `–f` option instructs the `automount` daemon to consult the local master map first and then the NIS-distributed master map. (The `–m` option instructs the `automount` daemon to ignore the NIS-distributed master map completely, if there is one.) For example:

  ```
  # automount –f /etc/auto.master
  ```

- Specify mount points on the command line, in addition to those included in the local `auto.master` file. For example:

  ```
  # automount –f /etc/auto.master \
    /src /etc/auto.src –ro,soft
  ```

- Nullify one of the entries in the local `auto.master` map. For example:

  ```
  # automount –f /etc/auto.master /home –null
  ```

- Replace an entry in the local `auto.master` map with one of your own. For example:

  ```
  # automount –f /etc/auto.master \
   /home /mine/auto.home –rw,intr
  ```

See `automount`(8) for more information on the `automount` command and its options.

8.5.2 Unmounting a Remote File System or Directory

To unmount a remote file system or directory, do the following:

1. Unmount the remote file system or directory by using the `umount` command with the following syntax:

   umount {|| *filesystem* | *directory*||}

2. Verify that the file system or directory is unmounted by entering the `mount` command with no arguments.

   The mounted file systems and directories are displayed.

See umount(8) for more information on umount command options.

The following command unmounts the /mnt local directory, containing the reference pages mounted in Section 8.5.1.2:

```
# umount /mnt
```

The following command unmounts all NFS file systems:

```
# umount -A -t nfs
```

The following command unmounts all file systems exported from host2:

```
# umount -h host2
```

9
UNIX-to-UNIX Copy Program

The UNIX-to-UNIX Copy Program (UUCP) is a group of programs that enables batched, error-free file transfer and remote command execution between two UNIX systems. UUCP is typically used to transfer electronic mail, network nets, and public domain software over low-speed, low-cost communications links. Tru64 UNIX implements the HoneyDanBer version of UUCP.

This chapter describes:

- The UUCP Environment
- How to plan for UUCP
- How to configure your system for UUCP
- How to manage UUCP

For general information about UUCP see `uucp_intro`(7). For information on how to use UUCP, see the *Command and Shell User's Guide*.

9.1 UUCP Environment

In the UUCP environment, systems can be connected to each other in the following ways:

- Directly connected to each other, if they are in close proximity.
- Connected through modems and a telephone network, if they are not in close proximity.
- Connected through a local area network (LAN), if they are not in close proximity.

Figure 9–1 shows two simple UUCP configurations. Figure 9–2 shows a sample UUCP configuration on a LAN in which Host A has a TCP/IP connection with Host C.

Figure 9–1: Sample Simple UUCP Configuration

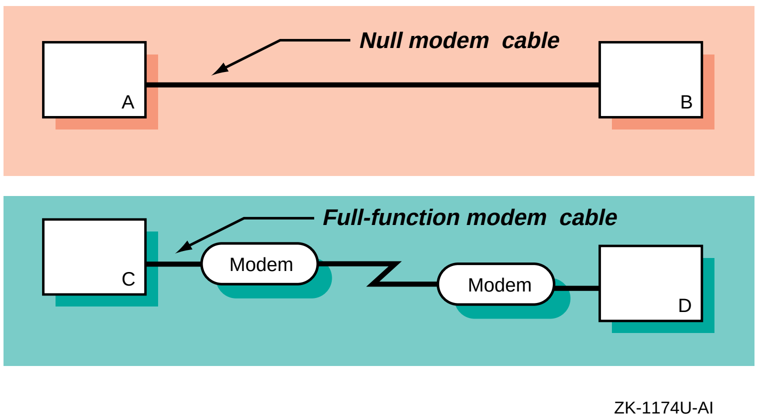
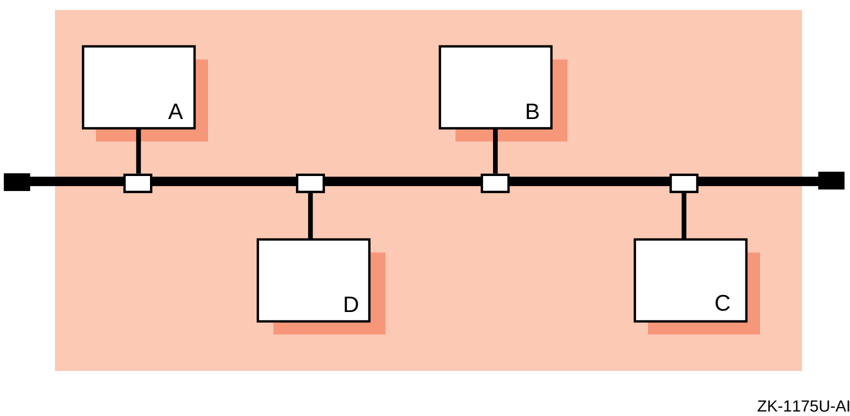


Figure 9–2: Sample UUCP Over TCP/IP Configuration



9.2 Planning UUCP

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

9.2.1 Verifying the Correct Hardware

In verifying the correct hardware, you are verifying both the cables and modems, if used.

Make sure you are using 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 4–1.

If the two systems are connected through modems and telephone lines, see Table 4–6 for a list of modem cables to use. When using modems with UUCP, make sure that both the local and the remote modems are correctly configured.

UUCP can also be configured to run over TCP/IP local area networks (LANs). For information on running UUCP over a LAN, see `uucp_manual_setup`(7).

9.2.2 Preparing for the Configuration

UUCP configuration consists of the following parts:

- Defining connection information for your system
- Defining dial-up information for outgoing calls
- Defining information for receiving incoming calls

The type of information you need depends on the types of connections you plan to set up and use.

Appendix A contains a worksheet that you can use to record the information that you need to provide to configure UUCP.

9.2.2.1 Information for Connections

Figure 9–3 shows Part 8A of the Configuration Worksheet.

Figure 9–3: Configuration Worksheet, Part 8A

Part 8A: UUCP Setup

Connections

Type of connection: ☐ Modem ☐ Direct link ☐ TCP/IP

Modems:

Modem type:	________________
Baud rate:	________________
Device name:	________________
inittab entry ID:	________________

Direct links:

Remote system name:	________________	☐ Direct
Baud rate:	________________	
Device name:	________________	
/etc/inittab entry ID:	________________	

TCP/IP:

Outgoing connections:	☐ Yes ☐ No
Incoming connections:	☐ Yes ☐ No

If you are viewing this manual online, you can use the print feature to print a copy of this part of the worksheet. The sections that follow explain the information you need to record in Part 8A of the worksheet.

Type of connection

The types of connections you want to configure. You can configure one or all of the following connections:

- Modems — Modems enable you to use UUCP over analog transmission facilities, which include telephone lines.
- Direct (hardwired) links — Direct hardwired links connect systems with cables.
- TCP/IP — Connections using the TCP/IP protocol.

For modem connections, supply the following information:

Modem type

The type of modem you want to use. The supported devices are listed in the `/usr/lib/uucp/Devices` file. For more information, see `uucp_manual_setup`(7).

Baud rate

The speed at which the modem is to operate; for example: 1200, 2400, 9600, or any.

Device name

The name of the tty device that you want the modem to use, as listed in the `/dev` directory. If you are unsure of the terminal device, see `port`(7).

/etc/inittab entry ID

The process ID for the `uugetty` process entry in the `/etc/inittab` file. The `uugetty` process sets up speed, terminal flags, and the line discipline for managing terminals. For more information, see `uugetty`(8).

Note

The `uugetty` command should only be run on RS-232 lines, not printer or console lines.

For direct link connections, supply the following information:

Remote system name or Direct

The type of direct link. If you want to connect to a specific remote system, enter the name of the remote system. This restricts connections to that system only.

If you want to connect to any system to which you have a direct hardwired connection, check Direct.

Baud rate

The speed at which the direct link is to operate; for example: 1200, 2400, or 9600.

Device name

The name of the tty device that you want the direct link to use, as listed in the `/dev` directory. If you are unsure of the terminal device, see `port`(7).

/etc/inittab entry ID

The process ID for the `uugetty` process entry in the `/etc/inittab` file. The `uugetty` process sets up speed, terminal flags, and the line discipline for managing terminals. For more information, see `uugetty`(8).

> **Note**
>
> The `uugetty` command should only be run on RS-232 lines, not printer or console lines.

For TCP/IP connections, supply the following information:

Outgoing connections

If you want to configure UUCP to place outgoing calls over TCP/IP, check Yes. When you enable UUCP to place outgoing calls over TCP/IP, an entry for TCP/IP is added to the `/usr/lib/uucp/Devices` file.

Otherwise, check No.

Incoming connections

If you want to configure UUCP to accept incoming calls over TCP/IP, check Yes. When you enable UUCP to accept incoming calls over TCP/IP, the `/etc/inetd.conf` file is modified. In addition, you must stop and restart the `inetd` daemon to be able to accept UUCP calls over TCP/IP.

Otherwise, check No.

9.2.2.2 Information for Outgoing Systems

Figure 9–4 shows Part 8B of the Configuration Worksheet.

Figure 9–4: Configuration Worksheet, Part 8B

Part 8B: UUCP Setup

Outgoing System

Remote system name:	____________
Mode of connection:	☐ Modem ☐ Direct link ☐ TCP/IP
TCP/IP conversation protocol:	☐ g ☐ t ☐ e ☐ f
Calling times:	____________
Baud rate:	____________ Any ☐
Phone number (for modem):	____________
Login ID:	____________
For modem/direct links, expect send string:	☐ Carriage returns ☐ None ☐ Prompt

If you are viewing this manual online, you can use the print feature to print part of the worksheet. The following sections explain the information you need to record in Part 8B of the worksheet.

Remote system name

The name of the remote system to which you plan to connect.

Mode of connection

The mode of the connection. Check modem, direct hardwired, or TCP/IP. You must configure the mode of the connection with the information from Section 9.2.2.1.

TCP/IP conversation protocol

For TCP/IP connections, the TCP/IP conversation protocol, which can be one of the following:

- `g` — Specifies to use the default protocol, which provides error checking.
- `t` — Presumes an error-free channel and therefore is not reliable for use with modem connections.
- `e` — Used to communicate with sites that are running both Tru64 UNIX and other UNIX versions of UUCP.
- `f` — Relies on flow control of the data stream. It is meant for working over links that can virtually be guaranteed to be error free, specifically X.25/PAD links.

Calling times

The times when your system is allowed to connect to the remote host. You can select the following times:

- Any time of any day
- Evenings — Monday to Friday 5 p.m. to 8 a.m.; Saturday and Sunday, all day
- Any three nights — You can choose the three nights from the following:
  - Monday to Friday, 11 p.m. to 8 a.m.
  - Saturday, all day
  - Sunday, until 5 p.m.
- Never

Baud rate or Any

The baud rate that corresponds to a device you configured in the /usr/lib/uucp/Devices file, or you can specify any, if the device can be used at any speed.

Phone number (for modem)

For modem connections, the telephone number of the remote system. You can enter the complete telephone number or a dialing prefix and the telephone number.

A dialing prefix is defined in the /usr/lib/uucp/Dialcodes file. The /usr/lib/uucp/Dialcodes file contains dial code abbreviations and partial phone numbers that complete the telephone entries in the /usr/lib/uucp/Systems file. Entries in the /usr/lib/uucp/Dialcodes file contain an alphabetic prefix attached to a partial phone number that can include, for example, access codes, area codes, and exchange numbers.

If you know the dialing prefix, enter it on the worksheet. If none is defined, enter it and the sequence of numbers to be associated with the prefix.

Login ID

The login name for your system on the remote system. This must match the information in the /etc/passwd file on the remote system. Ask the administrator of the remote system for the login name and password that is assigned to your system on the remote system. The administrator of the remote system must include the login name and password for your system in the remote system's /etc/passwd file.

Note

Although the password for the login ID on the remote system is required in order to configure UUCP, do not write the password on this worksheet to protect system security.

For modem/direct links, expect-send string

The *expect-send* string to be used immediately before performing the login on the remote system. You can choose one of the following:

- To send a series of carriage returns before expecting any characters from the remote system
- To specify no *expect-send* strings

- To be prompted to enter *expect-send* strings

  Modems usually use a series of carriage returns as an *expect-send* string.

  For more information on *expect-send* strings, see `Systems`(4).

9.2.2.3 Information for Incoming Systems

Figure 9–5 shows Part 8C of the Configuration Worksheet.

Figure 9–5: Configuration Worksheet, Part 8C

Part 8C: UUCP Setup

Incoming System

Remote system name: ____________________

Local system name: ____________________

Login ID: ____________________

Alternative login ID: ____________________

Options: ____________________

REQUEST option: ☐ Yes ☐ No

SENDFILES option: ☐ Yes ☐ No

Additional READ/WRITE locations: ____________________

Additional NOREAD/NOWRITE locations: ____________________

Commands: ____________________

____

____

VALIDATE option: ☐ Yes ☐ No

CALLBACK option: ☐ Yes ☐ No

Phone number (for modem): ____________________

If you are viewing this manual online, you can use the print feature to print a copy of this part of the worksheet. The sections that follow explain the information you need to record in Part 8C of the worksheet.

Remote system name

The name of the remote system you want to allow to establish incoming UUCP connections.

Local system name

The name of your system. The default provided is the name that you assigned your system at installation.

Login ID

The login ID for the remote system. The login ID is automatically added to the `/etc/passwd` file on your system.

By convention, the login ID that you assign to a remote system establishing incoming connections is the system name prefixed with a U (uppercase u). For example, if you specify machine1 for incoming connections, the login ID, by convention, is Umachine1; however, you can select any login ID.

You also have the option of adding a comment to the `/etc/passwd` file for this login ID.

You have the option to assign more than one login ID for each incoming system. Assigning multiple logins to a remote system allows you to maintain better access control for users on the remote system. With multiple logins, you can grant privileged users on the remote system more access on your system than you do to nonprivileged users. With multiple logins, you can assign multiple sets of permissions.

You must provide this information to the administrator of each remote system that will connect to your system as an incoming system.

REQUEST option

If you want a remote system to ask for any queued work on the local system that is meant for that remote system, check Yes; otherwise, check No.

If you check Yes, remote system users can transfer files to and execute commands on a local system more easily. If security is a consideration, you can restrict this access so that the local system retains control of file transfers and command executions initiated by remote systems.

SENDFILES option

If you want the local system to try to send queued work to the calling remote system after the remote computer finishes transferring files to or executing commands on the local system, check Yes; otherwise, check No.

Security considerations at your site might require that you limit a remote system's access to the local system by using the default value `CALL` for this option.

Additional READ/WRITE locations

If you do not specify pathnames in the `READ` and `WRITE` options, `uucp` permits files to be transferred only to the `/usr/spool/uucppublic` directory. However, if you specify pathnames in these options, you must enter the pathname for every source and destination. If you enter a pathname in either option, you must also explicitly specify the public directory if you want the `uucico` daemon to be allowed to place files in that location.

Additional NOREAD/NOWRITE locations

These options allow you to explicitly specify directories and files on the local system to which the remote system cannot transfer data. These are exceptions to the `READ` and `WRITE` options.

Commands

A list of commands the remote system is allowed to run on the local system. If you list a set of commands, that list comprises the new default command set for the systems listed in the MACHINE entry of the `/usr/lib/uucp/Permissions` file. The default is the command `rmail` only.

VALIDATE option

If you want the calling remote system to use a specific ID and password, check Yes; otherwise, check No.

If you use this option, no other ID from the remote system can call in. Several systems, however, can use the same ID. The `VALIDATE` option is meaningful only when the login ID and password are protected.

CALLBACK option

If you want the local system to contact the remote system before the remote system can transfer any files to the local system, check Yes; otherwise, check No.

If both systems use the `CALLBACK` option in their respective `Permissions` files, they will never be able to communicate with each other.

Phone number (modems only)

For modem connections, the phone number and speed of the modem attached to the local system. You must provide this information to the administrator of each remote system that will connect to your system as an incoming system.

9.3 Configuring UUCP

After you complete the required UUCP planning (Section 9.2), use the `uucpsetup` script to configure UUCP. To invoke the `uucpsetup` script, log in as superuser and either choose the UNIX-to-UNIX Copy Program (UUCP) option from the Setup Menu or enter the following command:

/usr/sbin/uucpsetup

The uucpsetup script prompts you for information required to configure connections, incoming systems, and outgoing systems.

Table 9–1 summarizes the various uucpsetup command syntaxes:

Table 9–1: Additional uucpsetup Commands

Use this command:	If you want to:
uucpsetup	Configure connections, incoming systems, and outgoing systems
uucpsetup –i	Configure the incoming systems only
uucpsetup –m	Modify UUCP connections
uucpsetup –o	Configure the outgoing systems only
uucpsetup –p	Configure the Poll file

For information about other options, see uucpsetup(8).

The following sections provide information on how to configure connections, incoming systems, outgoing systems, and the Poll file.

9.3.1 Configuring Connections

After you invoke uucpsetup, use the the information you gathered in Section 9.2.2.1 to configure UUCP connections. The following guidelines explain how to answer some of the script questions:

- Device names — The script lists the available device names. Enter the last letter or number of the device that you want the modem to use. For example, if you want to use tty01, enter 1.
- /etc/inittab entry ID — The script prompts you for the *Identifier* field and asks if this entry will be used in shared mode. It automatically supplies information for the other fields. No two processes can have the same ID.

  The following example illustrates how to select the process ID (PID) u4:

  ```
  Select an ID for the process in /etc/inittab file
  For example type 'u1': u4
  ```

  The ID that you select is checked against those that exist in the /etc/inittab file. If the ID that you assign exists, the uucpsetup script prompts you to enter another ID.

  You must also indicate whether the system will use the modem or direct line in shared mode.

For more information on the `/etc/inittab` file, see `inittab`(4).

9.3.2 Configuring Outgoing Systems

After you invoke the `uucpsetup` script, use the the information you gathered in Section 9.2.2.2 to configure UUCP for outgoing systems. This enables you to use UUCP to connect to other remote systems.

If you are doing a complete UUCP setup, the `uucpsetup` script prompts you for information on outgoing systems when you finish configuring connections. The following guidelines explain how to answer some of the script questions:

- Phone number — If you choose a dialing prefix and the telephone number, the script prompts you to enter a prefix to be defined in the `/usr/lib/uucp/Dialcodes` file. After you enter the prefix, the script prompts you for the meaning of the prefix. Enter the sequence of numbers that you want the system to substitute for the prefix. The following example illustrates how to define the prefix btown to be the dialing sequence 1617772:

  ```
  Enter the prefix for the Dialcodes file; for example "boston"
         stands for 9=16171234 : btown
  What telephone number does the prefix stand for; Please include
         the long distance access code, area, or country codes;
         for example type 9=1617123 : 9=1617772
  ```

  The 9 in this example is used to obtain a secondary dial tone. The 9 is site specific; it can be different for your site. The equal sign (=) is used with the 9, or number for your site, and means “wait for the dial tone.” Following the equal sign (=) is the rest of the number. Enter the rest of the number.

- Password — For security considerations, the password is not written on the worksheet. However, when the script prompts for it, you must enter it.

If you define an outgoing TCP system, edit the `/etc/uucp/Systems` file and add an entry for the remote system. The remote system name must be the fully qualified name.

9.3.3 Configuring Incoming Systems

After you invoke the `uucpsetup` script, use the the information you gathered in Section 9.2.2.3 to configure UUCP for incoming systems. This enables specific remote systems to connect to your system using UUCP.

If you are doing a complete UUCP setup, the script prompts you for information on incoming systems when you are done configuring outgoing systems.

The first time you add an incoming system, the Incoming Systems Configuration menu prompts you for the name of the system you want to add. After you add an incoming system, this menu offers you the following choices:

- Specify a remote system name.
- Specify options for all the other systems not specified in the `Permissions` file but listed in the `Systems` file.
- Neither. If you choose this option, the script terminates and the defaults for the options are not entered in the `Permissions` file.

The following guidelines explain how to answer some of the script questions:

- Password — The script invokes the `vipw` command for you to enter the password for the incoming system. Press Return and, after viewing the entry in the `/etc/passwd` file, exit the editing session by entering `:wq`. Then supply a password for the new entry:

  ```
  Invoking 'vipw'.
  Press RETURN to continue...
   Return
  root:fQPPWjF20Dfso:0:1:Charles Root:/:/bin/csh
  nobody:*Nologin:4294967294:4294967294:anonymous NFS user:/:
  daemon:*:1:1:Mr Background,,,:/:
  uucp:No Login:2:2:UNIX-to-UNIX Copy:/usr/spool/uucppublic:\
       /usr/lib/uucp/uucico
  bin:*:3:4:Mr Binary:/bin:
  marcy:5jW0VXKeP6n1E:1242:15:Marcy Darcy,,,:\
       /usr/users/marcy:/bin/false
  Umachine1:H/kj951Fq12ub:2:2:uucp login:/usr/spool/uucppublic:\
      /usr/lib/uucp/uucico
  ~
  ~
  ~
  "/etc/ptmp" 15 lines, 933 characters
  :wq
  15 password entries, maximum length 100

  You must enter a password
  Changing password for Umachine1.
  New password:
  Retype new password:
  ```

  You must provide this information to the administrator of each remote system that will connect to your system as an incoming system.

- Commands — The script prompts you for each command separately.

If you define an incoming UUCP system and your system uses NIS, edit the /etc/passwd file and add the wildcard (+:) as the last line (if not there already).

9.3.4 Configuring the Poll File

After you invoke the uucpsetup script with the –p option, you configure the /usr/lib/uucp/Poll file by completing the following steps:

1. Enter 1 (Configure the Poll file) from the Poll File Configuration Menu.
2. Enter the name of the remote system, which has been configured in the /usr/lib/uucp/Systems file as an outgoing system.
3. Enter the sequence of hourly intervals. For example, to have the system polled every 4 hours, enter 0 4 8 12 16 20.

   Press Return to update the Poll file.
4. To add another system to the Poll file, enter y; otherwise, press Return to exit uucpsetup.

9.4 Monitoring the File Transfer Queue

Monitoring the file transfer queue enables you to determine the status of several types of networking operations, including jobs that have been queued on a local system for transfer to a remote system. General users and system administrators can monitor the file transfer queue.

9.4.1 Getting Queue Status Manually

To get queue status manually, use the uustat –q command.

This command lists the jobs (waiting to execute or currently executing) queued for all systems. If a status file exists for a system, its date, time, and status information are reported.

The uustat command also allows you to do the following:

- Get information about the status of mail activities.
- Control uucp jobs queued to run on remote systems.
- Check the status of uucp connections to other systems, using the –m flag.
- Cancel transfer requests, using the –k flag.
- Monitor requests for file transfers generated with the uucp and uuto commands, and requests for command executions generated with the uux command.

See uustat(1) for more information on uustat flags.

The following example shows all jobs in the current queue: one command file for system host4, three command files for system host6, and two command files for system host8. The command files for system host6 have been in the queue for 2 days.

```
# uustat –q
host4 1C Sat May 9 11:12:30 1992 SUCCESSFUL
host6 3C(2) Sat May 9 11:02:35 1992 CAN'T ACCESS DEVICE
host8 2C Sat May 9 10:54:02 1992 NO DEVICES AVAILABLE
```

9.4.2 Getting Queue Status Automatically

You can automatically receive status information about the uucp file transfer queue. To enable this mechanism, edit the /usr/spool/cron/crontabs/uucp file and delete the comment character (#) from the beginning of the following line:

```
# 48 8,12,16 * * * /usr/lib/uucp/uudemon.admin > /dev/null
```

In the preceding example:

48	Represents minutes
8,12,16	Represents hours based on 24-hour clock notation
* * *	Three asterisks are placeholders representing the day of the month, the month of the year, and the day of the week

The cron daemon will run the uudemon.admin shell script daily at 48 minutes past the hours 8, 12, and 16; that is, at 8:48 a.m., 12:48 p.m., and 4:48 p.m. The uudemon.admin script sends mail to the uucp login ID containing queue status information.

Note

These times are the defaults. You can change the time to fit the needs of your site by editing the line in the /usr/spool/cron/crontabs/uucp file.

You can also manually run the uudemon.admin script. If you do, you should run it at least once a day.

9.4.3 Guidelines for Checking Queue Status

When examining queue status, check the number and age of the file-transfer and command execution requests queued in the

`/usr/spool/uucp/`*`system_name`* directory. In some cases, queued jobs remain in the queue for some time, essentially going undelivered. The status information you need to check includes:

- The age in days of the oldest request in each queue
- The number of times the local system has tried and failed to reach the specified computer
- The reason for the failure to contact the specified system

  See Appendix F for error messages and solutions.

If necessary, delete the files in the queue, either manually or automatically. See Section 9.5 for information on deleting files.

9.5 Cleaning Up the Spooling Directories

Each system connected by UUCP has the following spooling directories:

- The `/usr/spool/uucp/`*`system_name`* directory is the UUCP spooling directory. It contains queued local requests for file transfers and command executions on remote systems. These files are removed by the `uucp` program after they are transferred to the designated system.
- The `/usr/spool/uucppublic` directory is the UUCP public directory. When a user transfers a file to a remote system or issues a request to execute a command on an other system, the files generated by these UUCP commands are stored in the public directory on the designated system.

Depending upon the size of your installation and the number of files sent to the local `/usr/spool/uucppublic` directory by users on remote systems, the public directory can become quite large. Similarly, if requests are not transferred to remote systems for whatever reasons, the spooling directory could also become quite large. Therefore, part of UUCP management is to clean up the spooling directories and conserve disk resources.

9.5.1 Cleaning Up Directories Manually

To clean up the spooling directories manually, do the following:

1. Log in as root.
2. Remove files by using the `uucleanup` command, with the following syntax:

   uucleanup [*options...*]

The `uucleanup` program performs the following tasks:

- Informs the system manager of requests to send files to and receive files from remote systems that the local system cannot contact.
- Warns users about requests that have been waiting in the spooling directory for a given period of time. The default is 1 day.
- Returns to the original sender mail that cannot be delivered.
- Removes all other files older than a specified number of days from the spooling directory.

Note

Depending on the size of your installation and the available storage space on the local system, you can set the age limit for any length of time. However, you should allow files to remain in the spooling directory for at least the default number of days.

See `uucleanup`(8) for more information on the `uucleanup` command options.

The following example deletes all old files in the UUCP spooling and public directories for system host2 on the local system:

```
# uucleanup –shost2
```

9.5.2 Cleaning Up Directories Automatically

Although automatic cleanup is not enabled when UUCP is installed, you can enable it by doing the following:

1. Log in as root.
2. Edit the `/usr/spool/cron/crontabs/uucp` file and delete the comment character (#) from the beginning of the following line:

   ```
   # 45 23 * * * ulimit 5000; /usr/lib/uucp/uudemon.cleanu > /dev/null
   ```

   In the preceding example:

`45`	Represents minutes
`23`	Represents hours based on 24-hour clock notation
`* * *`	Three asterisks are placeholders representing the day of the month, the month of the year, and the day of the week

   The `cron` daemon will start the `uudemon.cleanu` shell script daily at 45 minutes after hour 23; that is, at 11:45 p.m. The shell script in turn starts the uucleanup program. This time is the default. You can change the time to fit the needs of your site by editing the line in the /usr/spool/cron/crontabs/uucp file.

You can instruct the cron daemon to run the uudemon.cleanu shell script daily, weekly, or at longer intervals, depending on the number of uucico and uuxqt transactions that occur on the local system.

The uudemon.cleanu script incorporates the actions of the uucleanup program and performs the following additional tasks:

- Locates and deletes empty directories and files older than 30 days from the /usr/spool/uucppublic directory. This helps keep the local file system from overflowing when users send files to the public directory. If the local system does not have enough storage space to accommodate a large /usr/spool/uucppublic directory, you can change the 30-day default to a shorter time period by modifying the uudemon.cleanu shell script.
- Cleans up all the uucp spooling directories, including the public directories, unless you direct it to clean up only the directories of a specific system by issuing the uucleanup –s *system_name* command.
- Updates archived log files, removing log information more than 2 days old. The script removes log files for individual computers from the /usr/spool/uucp/.Log directory, merges them, and places them in the /usr/spool/uucp/.Old directory, which contains old log information.
- Mails a summary of the status information gathered during the current day to the UUCP login ID. You can modify the script to send status information to other login IDs, such as root.

The operating system allots UUCP a specified amount of storage space for any one log file; the number of blocks is determined by the default ulimit value. If the uudemon.cleanu script fails to execute because the ulimit value is set too low for the requirements of the local system, you should increase the value.

See uudemon(8) for more information on command options.

9.5.3 Guidelines for Removing Files

When removing files from the queue, observe the guidelines for the following files:

- Execute files — Usually, you can remove execute files that have been in the queue for at least 2 days, using either the uucleanup or uudemon.cleanu script. The execute files are still queued because the data files required to execute the specified command on the designated system were not transferred. Since data files are generally sent at the same time as execute files, the transfer probably failed at the point of destination. Execute files are named `X.`*`filename`* and data files are named `D.`*`filename`*.

- Command files — Before removing old command files, make every possible effort to establish the connection and transfer the files. You can then remove these files by using either the `uucleanup` or `uudemon.cleanu` script. Command files are named `C.`*`filename`*.

9.6 Viewing Log Files

The `uucp` program creates a log file for each remote system with which your local system communicates. Each time you use the networking utilities facility, `uucp` places status information about each transaction in the appropriate log file. Log file names can be in either of the following forms:

`/usr/spool/uucp/.Log/`*`daemon_name/system_name`*

`/usr/spool/uucp/.Log/`*`command_name/system_name`*

In the preceding example:

daemon_name	Represents either `uucico` (called by the `uucp` and `uuto` commands) or `uuxqt` (called by the `uux` command)
command_name	Represents either `uucp` or `uux`
system_name	Represents the name of the system with which your local system is communicating

To display individual log files, use the `uulog` command, with the following syntax:

uulog [*options...*]

You can use the `uulog` command to display a summary of `uucp` and `uux` requests by user or by system. See `uulog`(1) for more information on the `uulog` command and its options.

Instead of viewing the log files individually, you can have the `uudemon.cleanu` script automatically append these log files to one primary log file, and then view only the one log file.

The `uudemon.cleanu` script combines the `uucico`, `uuxqt`, `uux`, and `uucp` log files on a system and stores them in a directory named `/usr/spool/uucp/.Old`. By default, the `uudemon.cleanu` script saves log files that are up to 2 days old.

You can change the default by modifying the –o2 option in the following line in the uudemon.cleanu script:

```
uucleanup –D7 –C7 –X2 –o2 –W1
```

If storage space is a problem on a particular system, consider reducing the number of days that the files are kept in the individual log files. See Section 9.5 for information on setting up the uudemon.cleanu script.

The following command displays the log file for uucico requests for system host2:

```
# uulog –s host2
```

The following command displays the log file for uuxqt requests for system host1:

```
# uulog –x host1
```

The following command displays the last 40 lines of the file transfer log for system host6 and executes a tail –f command. Press Ctrl/C to terminate the command.

```
# uulog –f host6 –40
```

9.7 Cleaning Up sulog and cron/log Files

The following two system log files are affected by the uucp program:

- The /usr/adm/sulog file contains a history of superuser (su) command usage. The uudemon entries in the /usr/spool/cron/crontabs/uucp file each use the su command.
- The /usr/adm/cron/log file contains a history of all the processes generated by the cron daemon.

Both files can grow quite large over a period of time. Purge these files periodically to keep them at a reasonable size. See *System Administration* for information on these files.

9.8 Limiting the Number of Remote Executions

The Maxuuxqts file, located in the /usr/lib/uucp directory, limits the number of uuxqt processes running simultaneously on a local system. Typically, the file requires no configuration or maintenance unless the system on which it is installed is utilized frequently and heavily by users on remote systems.

To change the number of uuxqt processes on the system, edit the Maxuuxqts file and change the ASCII number to meet the needs of your installation; the default is 2. In general, the larger the number, the greater the potential load on the local system.

9.9 Scheduling Work in the Spooling Directory

When users issue uucp commands to copy files and execute remote commands, the files containing these work requests are queued for transfer in the local /usr/spool/uucp/*system_name* directory. The uucp daemon uusched schedules the transfer of these files.

9.9.1 Starting uusched Manually

To schedule jobs, start the uusched daemon by using the uusched command, with the following syntax:

uusched [*options...*]

The following options are available:

–x *debug_level*	Produces debugging information about the progress of the uusched activity. The valid range for the debugging level is 0 to 9, with a default of 5. Higher numbers produce more detailed debugging information.
–u *debug_level*	Passes the –x *debug_level* specification on to the uucico daemon, which then produces debugging output about the file-transport activities.

9.9.2 Starting uusched Automatically

Although you can start the uusched daemon manually, the preferred method is to start it automatically at specified intervals by using the uudemon.hour shell script, which is stored in the /usr/lib/uucp directory. The shell script, in turn, is started periodically by the cron daemon, based on instructions in the /usr/spool/cron/crontabs/uucp file.

The /usr/lib/uucp/Maxuuscheds file limits the number of remote systems that the uucico program can contact at any one time. This file is used in conjunction with the uusched daemon and the lock files in the /usr/spool/locks directory to determine the number of systems currently being polled.

The Maxuuscheds file requires no configuration or maintenance unless the system on which it is installed is utilized frequently and heavily by users on remote systems. You use this file to help manage system resources and load averages.

The Maxuuscheds file contains an ASCII number that you can change in order to meet the needs of your installation; the default is 2. In general, the larger the number, the greater the potential load on the local system.

See uusched(8) for more information on the uusched command and its options.

The following command starts the uusched daemon manually as a background process:

```
# /usr/lib/uucp/uusched &
```

9.10 Calling File Transfer Programs (uudemon.hour)

The uudemon.hour shell script is used in conjunction with the Poll file, the uudemon.poll shell script, and the /usr/spool/cron/crontabs/uucp file to initiate calls to remote systems. Specifically, uudemon.hour calls programs involved in transferring files between systems at specified hourly intervals.

You can instruct the cron daemon to run the uudemon.hour shell script at specified hourly intervals. The frequency at which you run the script depends on the amount of file transfer activity originating from the local computer.

Although the uudemon.hour shell script is not enabled when UUCP is installed, you can enable it by doing the following:

1. Log in as root.
2. Edit the /usr/spool/cron/crontabs/uucp file and delete the comment character (#) from the beginning of the following line:

   ```
   # 25,55 * * * * /usr/lib/uucp/uudemon.hour > /dev/null
   ```

   In the preceding example:

25,55	Represents minutes past the hour
* * * *	Four asterisks are placeholders representing the hour interval, the day of the month, the month of the year, and the day of the week

   The cron daemon will run the uudemon.hour script at 25 minutes past the hour and again at 55 minutes past the hour; for example, at 8:25 a.m. and 8:55 a.m., 9:25 a.m. and 9:55 a.m., and so on.

   These times are the defaults. You can change the time to fit the needs of your site by editing the line in the /usr/spool/cron/crontabs/uucp file.

If users on the local system initiate a large number of file transfers, you might need to specify that the `cron` daemon should start the `uudemon.hour` script several times an hour. If the number of file transfers originating from the local system is low, you can probably specify a start time once every 4 hours, for example.

9.11 Polling Remote Systems (uudemon.poll)

The `uudemon.poll` shell script is used in conjunction with the `Poll` file, the `uudemon.hour` shell script, and the `/usr/spool/cron/crontabs/uucp` file to initiate calls to remote systems. The `uudemon.poll` shell script polls the systems listed in the `/usr/lib/uucp/Poll` file. In addition, it creates command files for the systems listed in the `Poll` file.

The time at which you run the `uudemon.poll` script depends on the time at which you run the `uudemon.hour` script. You generally schedule the polling shell script to run before the hourly script. This schedule enables the `uudemon.poll` script to create any required command files before the `cron` daemon runs the `uudemon.hour` script.

Although the `uudemon.poll` script is not enabled when UUCP is installed, you can enable it by doing the following:

1. Log in as root.
2. Edit the `/usr/spool/cron/crontabs/uucp` file and delete the comment character (#) from the beginning of the following line:

   ```
   # 20,50 * * * * /usr/lib/uucp/uudemon.poll > /dev/null
   ```

   In the preceding example:

`20,50`	Represents minutes past the hour
`* * * *`	Four asterisks are placeholders representing the hour interval, the day of the month, the month of the year, and the day of the week

   The `cron` daemon will run the `uudemon.poll` script at 20 minutes past the hour and again at 50 minutes past the hour, for example, at 8:20 a.m. and 8:50 a.m., 9:20 a.m. and 9:50 a.m., and so on.

   These times are the defaults. You can change the times at which the `cron` daemon executes the `uudemon.poll` script to correspond to the times you set up for the `uudemon.hour` script. You should instruct the `cron` daemon to run the `uudemon.poll` script about 5 to 10 minutes before running the `uudemon.hour` script.

10

Network Time Protocol

The Network Time Protocol (NTP) provides accurate, dependable, and synchronized time for hosts on both wide area networks (WANs) like the Internet network and local area networks (LANs). In particular, NTP provides synchronization traceable to clocks of high absolute accuracy, and avoids synchronization to clocks keeping bad time. The Tru64 UNIX NTP subsystem is derived from the University of Maryland's implementation, `xntpd` Version 3.4m. The `/etc/ntp.conf` file is the configuration file for the `xntpd` daemon.

This chapter describes:

- The Tru64 UNIX NTP subsystem and its components
- Guidelines for configuring and administering NTP
- NTP configuration planning
- NTP configuration
- Day-to-day NTP management tasks

For introductory information on NTP, see the `ntp_intro`(7) reference page. Also, for information about the latest releases of NTP, more examples of how to configure NTP subnets, and extensive NTP troubleshooting information, visit the NTP website at `http://www.eecis.udel.edu/~ntp`.

As an alternative to NTP, you can set your system time by using the `rdate` command or the `timed` daemon.

Note

> You should use NTP for time synchronization. The `timed` daemon is provided only for compatibility. If you plan to run both the `timed` daemon and NTP, you should configure NTP first and run the timed daemon with the `–E` option.

For more information on the `rdate` command, see `rdate`(8) and `ntp_manual_setup`(7).

For more information on the `timed` daemon, see `timed`(8) and `timedsetup`(8).

10.1 NTP Environment

In the NTP environment, systems can have the following roles:

- Client — An NTP client system is a system that synchronizes its time with local NTP servers.
- Server — An NTP server is a local system that synchronizes its time with an Internet NTP server or with a local reference clock, or both for better accuracy.

Figure 10–1 shows a sample NTP configuration on a LAN in which Host D is an NTP server that uses a local reference clock as its time source. Hosts A, B, C, E, F, and G are NTP clients, synchronizing their time with Host D.

Figure 10–1: Sample NTP Configuration (Local Clock)

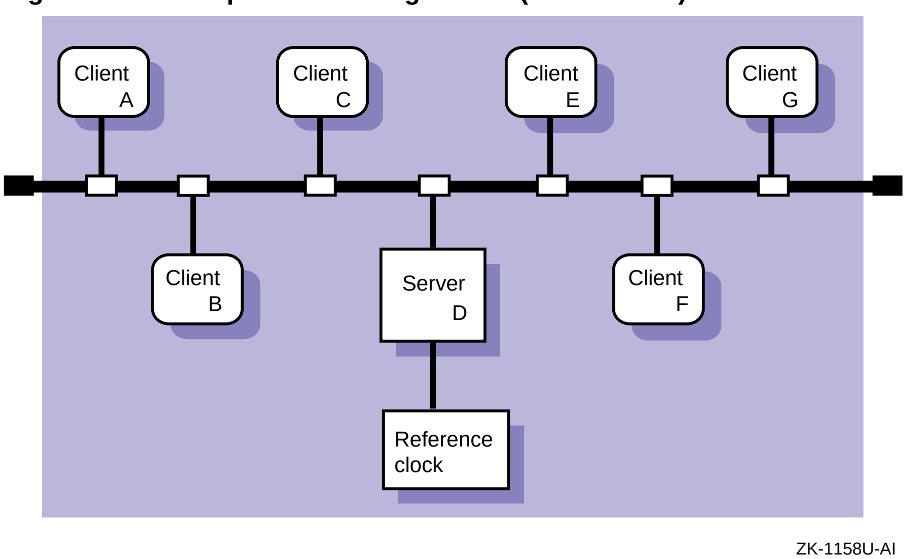
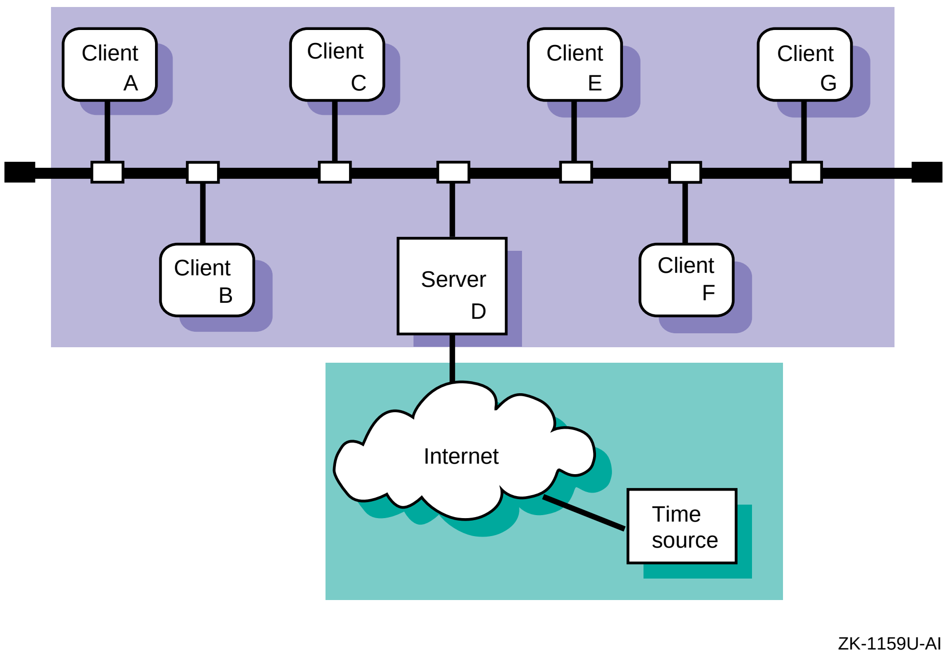


Figure 10–2 shows a sample NTP configuration in which host D is an NTP server that uses an Internet time server as its time source. Hosts A, B, C, E, F, and G are NTP clients, synchronizing their time with Host D.

Figure 10–2: Sample NTP Configuration (Internet Source)



10.2 NTP Planning

Appendix A contains a worksheet that you can copy and use to record the information that you need to complete the tasks in this book. If you are viewing this manual online, you can use the print feature to print a copy of this part of the worksheet.

Figure 10–3 shows Part 9 of the Configuration Worksheet. The following sections explain the information you need to record in Part 9 of the worksheet.

Figure 10–3: Configuration Worksheet, Part 9

Part 9: NTP Setup

Server

Time source: ______________________________

Server Internet address:	Server name:	NTP version:
____________	____________	____________
____________	____________	____________
____________	____________	____________

Client

Local NTP server address:	Server name:	NTP version:
____________	____________	____________
____________	____________	____________
____________	____________	____________

System's role

Your system can be a local NTP server or an NTP client, or both.

10.2.1 Server Information

Time source

Your system's time source. For local NTP servers, the time source is one of the following:

- Internet NTP servers — If your system is connected to the Internet, you can obtain a list of possible NTP Internet servers from `http://www.eecis.udel.edu/~ntp` on the World Wide Web. You should select a minimum of three systems from the server list with which to synchronize the time on your local NTP servers. Obtain permission from the contact person listed for each Internet server before specifying it as a server for your local NTP servers.
- A local reference clock — If your network is not connected to the Internet network, you must select a system on your network to be the local reference clock. A local reference clock is a lightly loaded and highly available system that keeps good time. See `ntp_manual_setup`(7) and `ntp.conf`(4) for information on setting up a local reference clock.

Server Internet address

The IP address of the Internet NTP server or the local reference clock. Local NTP servers are the time sources for NTP clients.

Server name

The host name of the Internet NTP server.

NTP version

The version of NTP daemon running on the Internet NTP server or the local reference clock. This can be Version 1 (the `ntpd` daemon), Version 2 (the `xntpd` daemon), or Version 3 (the `xntpd` daemon).

10.2.2 Client Information

Local NTP server address

The local NTP server IP address. Local NTP servers are the time sources for NTP clients.

Server name

The local NTP server name.

NTP version

The version of NTP daemon running on the local NTP server. This can be Version 1 (the `ntpd` daemon), Version 2 (the `xntpd` daemon), or Version 3 (the `xntpd` daemon). Servers running Version 3.2 or earlier of the Tru64 UNIX operating system run Version 2 (the `xntpd` daemon); servers running Version 4.0 of the Tru64 UNIX operating system run Version 3 (the `xntpd` daemon).

10.3 Configuring NTP

After you gather the necessary information and select your Internet servers, configure NTP by using the `ntpsetup` script. The script enables you to configure all NTP clients and local NTP servers, provided they use Internet NTP servers as their time source.

Note

Do not use the `ntpsetup` script to configure NTP on local NTP servers that use a local or external reference clock as a time source. Instead, see `ntp_manual_setup`(7) for instructions.

If you plan to use both NTP and the `timed` daemon, set up NTP prior to setting up the `timed` daemon.

To configure NTP, do the following:

1. Invoke the `ntpsetup` script by choosing the Network Time Protocol (NTP) option from the Setup Menu or by entering the following command:

   ```
   # /usr/sbin/ntpsetup
   ```

   An explanation of the `ntpsetup` script appears on your screen.

   If the `timed` daemon has been configured on the system, the following message is displayed:

   ```
   NOTE: timed has been configured on this system.

   In order to insure proper coexistence, timed must be
   configured AFTER NTP.

   If you continue with the NTP configuration, the timed
   configuration data will be removed (which means timed cannot run).
   You will need to re-enable timed by using timedsetup again.

   Do you wish to continue (yes/no) [no default]?
   ```

2. Press Return following the script's explanation of what `ntpsetup` does.

3. Indicate whether or not you want to run the `xntpd` daemon with authentication.

   Authentication enables you to verify the authenticity of received NTP packets and indicate authenticity of outgoing NTP packets. If you want to use authentication, enter Y. If you do not want to use authentication, press Return.

4. Enter the names of the NTP servers for this system.

   For clients, enter the names of your site's local NTP servers. For servers, enter the names the Internet NTP servers. (See Section 10.2.1 for information on selecting Internet servers.) In either case, you should specify a minimum of three NTP servers.

   For NTP clients, entries in the `ntp.conf` file are designated server entries because clients can only synchronize their time with them. NTP servers, however, can contain server and peer entries. The term peer indicates that the local server can be synchronized to the remote server and that the remote server can be synchronized by the local server.

   If you enter the name of a host that your system cannot find an address for in the local `/etc/hosts` database or through BIND or NIS, the `ntpsetup` script prompts you for its IP address. For example:

   ```
   Hostname of NTP server/peer [no default]: host1
   Mode for host1 (server or peer) (s/p) [s]:  Return
   ```

```
Looking up host host1
             Cannot find an address for "host1".
             To add "host1" to the /etc/hosts file, you must know
             "host1"’s internet (IP) address.
Would you like to add "host1" to the /etc/hosts
          file (y/n) [y]?  Return
What is host1’s internet (IP) address [no default] ? 120.105.1.2
Is 120.105.1.2 correct (y/n) [no default] ? y
Is host1 running ntpd (V1) or xntpd (V2/V3) (V1/V2/V3) [V3] ? V2  Return
Hostname of NTP server/peer [no default]: host2
Mode for host1 (server or peer) (s/p) [s]:  Return
Looking up host host2 ...found.
Is host2 running ntpd (V1) or xntpd (V2/V3) (V1/V2/V3) [V3] ? V2  Return
Hostname of NTP server/peer [no default]: host3
Mode for host1 (server or peer) (s/p) [s]:  Return
Looking up host host3 ...found.
Is host3 running ntpd (V1) or xntpd (V2/V3) (V1/V2/V3) [V3] ? V2  Return
Hostname of NTP server/peer [no default]:  Return
```

If you have selected to use authentication, the `ntpsetup` script prompts you for a key number to be associated with each server after it prompts you for the mode, as follows:

`Enter the key number to be associated with` *`hostname:`*

The `ntpsetup` script then displays the list of servers that you entered. If the list is correct, enter c to continue. If the list in incorrect or incomplete, enter r to redo it.

5. Press Return following the script’s explanation that if any of your NTP servers are not on your subnet you must run either the `routed` or the `gated` daemon to access them.

   For information on running the `gated` or `routed` daemon, see Chapter 2.

6. Indicate whether or not you want to run the `xntpd` daemon with the `–g` option.

   The `–g` option allows `xntpd` to correct time differences of more than 1000 seconds between your system and that of your system’s NTP servers that occur after the `xntpd` daemon is started. Initial time differences are corrected before the `xntpd` daemon is started by the `ntpdate` command, which is run at boot time by the `/sbin/init.d/settime` script. If your system is sensitive to security threats, do not use the `–g` option. If you do not use the `–g` option, time differences of more than 1000 seconds will cause the `xntpd` daemon to log a message to the `syslog` daemon and exit.

7. Indicate whether you want to run the `xntpd` daemon with the `–x` option.

   The `–x` option prevents the `xntpd` daemon from setting the system time backward. The default is to allow `xntpd` to set the system time backward.

8. If you have chosen authentication, enter the authentication keys and key identifiers to be put in the /etc/ntp.keys file. Enter as many as you need. For example:

```
Enter key and key identifiers separated by one space (no default):1 defcd
Enter key and key identifiers separated by one space (no default): Return
```

   The ntpsetup script then displays the list of keys and key identifiers that you entered. If the list is correct, enter c to continue. If the list in incorrect or incomplete, enter r to redo it.

   The ntpsetup script displays a message similar to the following and exits:

```
Configuring your system to run NTP...done.

Starting the NTP daemon (xntpd)...
Setting kernel timezone variable
Setting the current time and date with ntpdate
Ntpdate succeeded.
Network Time Service started
To monitor NTP, type "/usr/bin/ntpq –p".
```

10.4 Enabling the High-Resolution Clock

The operating system includes an optional high-resolution clock that can be used for time-stamping and for measuring events that occur on the order of microseconds, such as the time spent in a critical code path. Programmers might be able to use this information to find the source of a bug or to determine where a program should be optimized to improve performance.

To enable the high-resolution clock, add the following line to the kernel configuration file and rebuild the kernel:

```
options MICRO_TIME
```

The system clock (CLOCK_REALTIME) resolution as returned by the clock_getres function does not change, nor does the timer resolution. However, the time as returned by the clock_gettime routine is extrapolated between the clock ticks, and the granularity of the time returned is in microseconds. The resulting time values are SMP-safe, they are monotonically increasing, and they have an apparent resolution of 1 microsecond.

10.5 Monitoring Hosts Running the xntpd Daemon

You can monitor the hosts running the xntpd daemon by using either the ntpq command or the xntpdc command.

To monitor the local host's NTP status using the ntpq command, use the following syntax:

ntpq [*options...*]

To monitor remote hosts' NTP status using the `ntpq` command, use the following syntax:

ntpq [*options...*] *host1 host2...*

Table 10–1 shows the `ntpq` command options.

Table 10–1: Options to the ntpq Command

Option	Function
`–c command`	Interprets *command* as an interactive format command and adds it to a list of commands to be executed on the specified host or hosts
`–i`	Forces `ntpq` to operate in interactive mode
`–p`	Prints a list of peers and a summary of their state

In interactive mode, use the `host` command to set the host to use as a reference for the other options; the local host is the default. Use the `peers` option to display the offsets between the current host and its `xntpd` servers. See `ntpq`(8) for more information.

The following example shows normal output from the `ntpq` command with the `–p` option:

```
% ntpq –p
     remote           refid      st  when poll reach  delay  offset   disp
==========================================================================
*host2.corp.com  host121.corp.co  2    47   64  377    31.3   93.94   16.5
+host4.corp.com  host2.corp.com   3   212 1024  377    33.8   89.58   16.9
 host8.corp.com  host2.corp.com  16 never   64    0     0.0    0.00  64000
```

The last line of the previous example shows that host8 is either not running NTP or cannot be reached.

To monitor the local host's NTP status using the `xntpdc` command, use the following syntax:

xntpdc [*options...*]

To monitor remote hosts' NTP status using the `xntpdc` command, use the following syntax:

xntpdc [*options...*] *host1 host2...*

Table 10–2 shows some of the `xntpdc` command options.

Table 10–2: Options to the xntpdc Command

Option	Function
`–c command`	Interprets *command* as an interactive format command and adds it to a list of commands to be executed on the specified host or hosts.
`–i`	Forces `xntpdc` to operate in interactive mode.
`–l`	Prints a list of peers that are known to the server.
`–p`	Prints a list of peers and a summary of their state. This is similar in format to the `ntpq –p` command.

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

The following example shows normal output from the `xntpdc` command with the `–p` option:

```
% xntpdc -p
     remote           refid      st  when poll reach  delay  offset   disp
==========================================================================
*host2.corp.com host121.corp.co   2    47   64   377   31.3   93.94   16.5
+host4.corp.com host2.corp.com    3   212 1024   377   33.8   89.58   16.9
.host5.corp.com host12.usc.edu    2   111 1024   377   39.1   46.98   17.7
```

10.6 Monitoring Hosts Running the ntpd Daemon

You can monitor the hosts running the `ntpd` daemon by using the `ntpdc` command; however, you should use the `xntpdc` command because it works with all versions of NTP and provides additional features.

10.7 Querying Servers Running NTP

You can query time by using the `ntp` and `ntpdate` commands. However, you should use the `ntpdate` command because it works with all versions of NTP and provides additional features to those provided by the `ntp` command.

11 Mail System

The Tru64 UNIX mail system enables users to send mail to other users, whether on the same system, same network, or the other side of the world. This chapter describes:

- The Tru64 UNIX mail system and its components
- Guidelines for configuring and administering mail on a standalone system or across an enterprise
- Mail configuration planning
- Mail configuration
- Day-to-day management tasks

For additional introductory information on mail, see also `mail_intro`(7), the *sendmail* book by O'Reilly & Associates, and the *Sendmail Installation and Operation Guide*. The guide is located at the following URL: `http://uwsg.ucs.indiana.edu/usail/mail/op/op.html`.

The implementation of mail in Tru64 UNIX is based on `sendmail` Version 8.8.8 from Sendmail, Inc.

11.1 Mail Environment

In the mail environment, systems can have the following roles:

- Standalone — A mail standalone system is one that processes, sends, and delivers mail locally. This is useful for configurations of from 1 to 6 systems. In small LAN configurations of two or more systems, one system serves the mailbox to the other systems using NFS. In this case, NFS must be configured on all systems.
- Client — A mail client system is a system that sends all mail to a mail server for processing and delivery. If the addressee is on the client system, the mail is delivered there. If not, the mail is forwarded to the destination system.
- Server — A mail server system is a system that receives mail from clients in a local domain for processing and delivery to other domains, the Internet, or other networks. In addition, the server also receives mail from other domains for delivery.

Figure 11–1 shows a sample standalone configuration on a LAN in which all hosts are configured as mail standalone systems. Host B is also an NFS server, exporting the `/var/spool/mail` directory to hosts A, C, D, and E. Hosts A, C, D, and E are also NFS clients, importing the `/var/spool/mail` directory from host B.

The hosts must also have identical information in their `passwd` and `aliases` files. This information can be distributed either by using NIS or by manually editing the files on each system.

Figure 11–1: Sample Mail Standalone Configuration

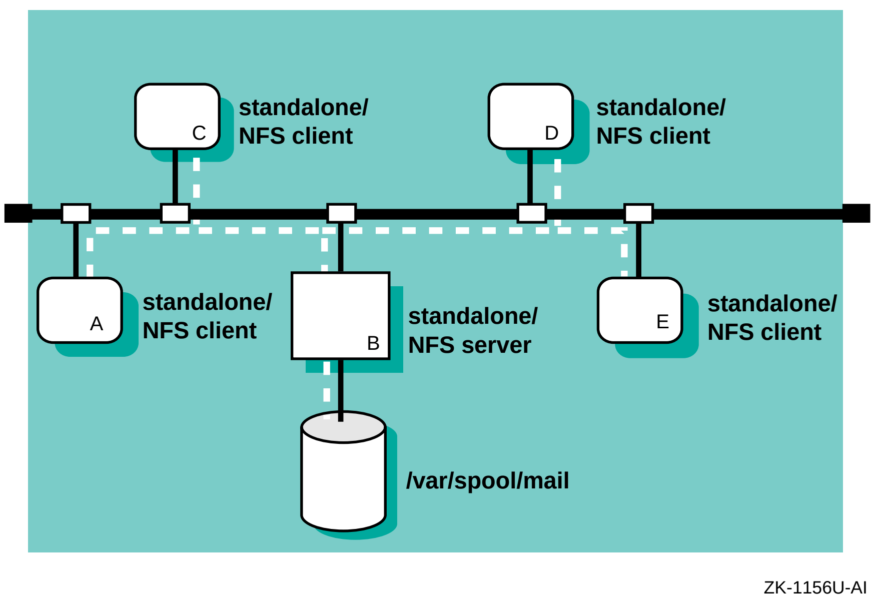
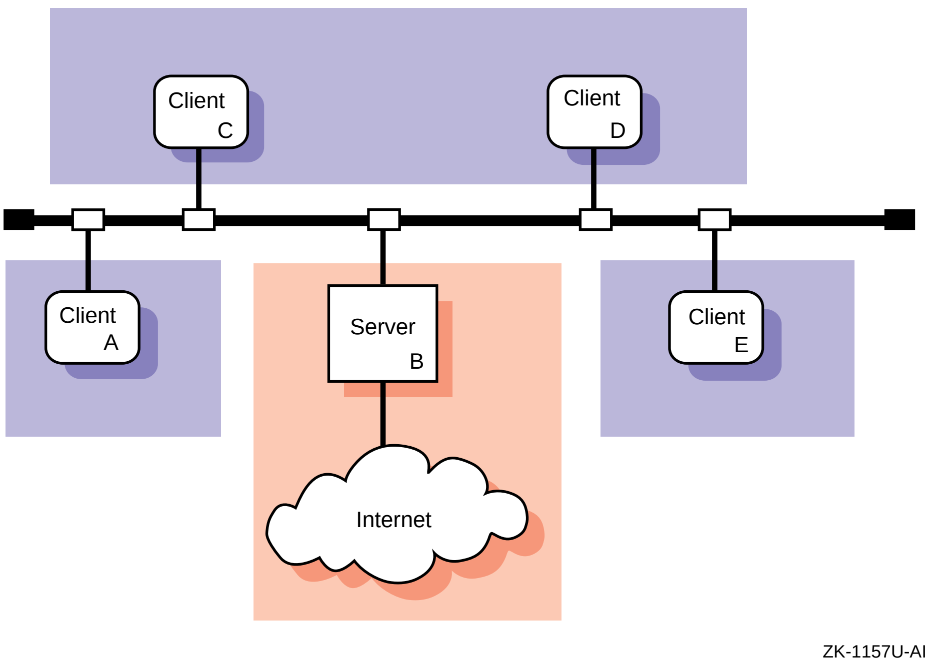


Figure 11–2 shows a sample client/server configuration in which host B is configured as a mail server and hosts A, C, D, and E are configured as mail clients. This is useful in larger enterprise networks that consist of multiple domains and connections to the Internet or other networks.

This configuration also provides for the creation of a natural hierarachy of mail servers in large enterprise networks with multiple domains. Mail clients in each domain would direct all traffic to one or more mail servers, depending on the number of clients in the domain. Each domain's servers would then forward mail to the enterprise's top domain servers for forwarding to the Internet. Since almost all of your local domain's mail traffic goes through the servers, this simplifies administration and problem resolution in that you only have to manage the servers.

The connection to the Internet in Figure 11–2 could be direct or through a local access provider. Business configurations would typically use firewalls and dedicated mail servers. If using a firewall, ensure the firewall and the mail server are configured to work with each other. See the documentation for your firewall product for more information.

Figure 11–2: Sample Mail Client/Server Configuration



If users need to send mail between systems that use different mail protocols, such as DECnet, SMTP, and UUCP, you should designate specific server systems in your network to perform that function. These server systems are also known as mail relays.

Additional mail configurations are possible, but they require more effort to plan for and to configure. See the *sendmail* book by O'Reilly and Associates and the *Sendmail Installation and Operation Guide* for more information.

In implementing a client/server mail environment, you need to decide how to:

- Direct outgoing mail to the servers
- Handle incoming mail to the domain
- Deliver mail to clients
- Handle DECnet mail

This section describes each of these topics.

11.1.1 Directing Outgoing Mail to Servers

To direct outgoing mail to a server, you include the DNS mail exchanger (MX) entry in the `/etc/namedb/hosts.db` file. This entry specifies a system in the local domain that can deliver mail to systems not directly connected to the local network. Using MX to route mail has the following benefits:

- You can define an MX record to point to all of the mail servers in your local domain. If a mail server is inaccessible, mail can be delivered to another host listed in the MX record.
- You can use MX records to define a system to be a mail exchanger for an inaccessible remote system. Then, if you send mail to the remote, inaccessible host, instead of being queued on your local system and periodically resent, the mail is sent to the mail exchanger and queued there until the host is restored.

For information on adding entries to the `/etc/namedb/hosts.db` file, see Section 6.5 and `bind_manual_setup`(7).

11.1.2 Handling Incoming Mail to the Domain

To simplify the handling of incoming mail to a domain and to ensure reliability, you should use domain-based addresses in your environment. Mail sent over the Internet is usually addressed in the following format:

username@hostname.domain

For example:`joe@host1.nyc.big.com`

Using domain-based addresses, this address appears as follows:

`joe@nyc.big.com`

Mail is sent to the local domain `nyc.big.com` instead of to the specific host within that domain `host1.nyc.big.com`; the return address is also `@nyc.big.com`. Then, the mail servers within the local domain decide how to deliver the mail to the user's account.

Domain-based addresses make it easier to manage your mail environment. You can change your mail system (that is, move user accounts and replace or move systems) without disrupting your mail delivery. These changes are transparent to users sending mail to your systems.

11.1.3 Delivering Mail to Clients

Once mail is delivered to the domain, you can deliver it to clients using either of the following mechanisms:

- Deliver the mail to the `/var/spool/mail` directory on each client.
- Deliver the mail to the server and use NFS to serve the mail directory to each client.

In order to deliver mail to each of the clients, each server in the domain must have an aliases file that contains entries for each user on the client. For example:

```
username1: username1@client1
username2: username2@client1
```

See Section 11.1.4 for information on distributing the aliases file.

11.1.4 Distributing the aliases File

For standalone and server systems, use the Network Information System (NIS) to distribute the mail aliases file from one machine. In a LAN environment with standalone systems, distribute the mail aliases file from the system that is also the NFS server. In a client/server environment, distribute the aliases file to the servers in the domain. In either case, sharing the aliases file among systems simplifies administration in that you only need to update one aliases file, instead of several.

For client systems, do not distribute the aliases file and, if using NIS, do not put a `yp` entry for the aliases database in the `svc.conf` file.

11.1.5 Distributing the passwd File

If you are using multiple server systems in a domain, make sure that the `passwd` information is identical on each system. For security reasons and to ensure correct mail delivery, you should do this by manually editing the `passwd` file on each server system.

11.1.6 Handling DECnet Mail

When you set up a mail server system, you must consider that the mail address formats for DECnet Phase IV and DECnet/OSI are different from those for TCP/IP. Therefore, you need to establish a mapping scheme to translate mail addresses when sending mail between a DECnet node and a TCP/IP node.

The mapping scheme used by the Tru64 UNIX version of the `sendmail` program for DECnet Phase IV is to encapsulate DECnet addresses inside a pseudodomain. For example, a typical DECnet Phase IV address has the following format:

nodename:: *username*

Mail addressed in this format is mapped to an address in the following format:

username@ *nodename.pseudodomain.top.domain*

The variables represent the following:

username
The user name.

nodename
The DECnet node name.

pseudodomain
An arbitrary string that specifies the DECnet pseudodomain. The pseudodomain can be an arbitrary string, but it must be used consistently throughout your organization. All of your mail systems must be configured to use the same string for the pseudodomain.

top.domain
Usually, your company's domain name; for example, `abc.com`.

The mapping for DECnet/OSI uses a similar scheme. A typical DECnet/OSI address has the following format:

username@ *namespace*:. *site.nodename*

Mail addressed in this format is mapped as follows:

username@ *nodename.site.namespace.pseudodomain.top.domain*

As with DECnet Phase IV, the pseudodomain can be an arbitrary string. However, if you use both DECnet Phase IV and DECnet/OSI within your organization the pseudodomain names should be different.

Some environments that support both DECnet Phase IV and DECnet/OSI use the DECnet Phase IV syntax for handling DECnet-based mail. This simplifies the mail administration task. In order to implement this, all DECnet-OSI nodes must have a unique Phase IV Synonym and must be configured to use the Phase IV Synonym. You can reconfigure a DECnet/OSI host by typing the following command line:

```
# ncl set session control application mail11 Node Synonym=true
```

See the DECnet/OSI documentation for more information.

11.2 Planning Mail

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

11.2.1 Verifying that Required Protocols are Installed

Depending on the protocols supported by your mail server, verify that the following required subsets are installed and configured:

- DECnet
- DECnet/OSI
- X.25 (PSInet)
- UUCP

See the documentation for each product for installation and configuration instructions. For UUCP, verify that the UUCP subset is installed by entering the following command:

```
# setld -i | grep OSFUUCP
```

If it is not installed, install it by using the `setld` command. For more information on installing subsets, see `setld`(8), the *Installation Guide*, or the *System Administration* manual.

11.2.2 Verifying that Required Services are Configured

The following table lists specific mail configurations and the network service required:

If you want to:	Configure this service:
Distribute the aliases file	NIS
Use domain-based addressing	DNS/BIND

To verify if NIS is needed, enter the following command as root:

```
# grep NIS_CONF /etc/rc.config
```

If the output returns as `NIS_CONF="YES"`, then NIS is required to distribute the `aliases` file. To see if DNS is needed, enter the following command as root:

```
# grep BIND_SERVERTYPE /etc/rc.config
```

If the output returns as `BIND_SERVERTYPE="CLIENT"` or `BIND_SERVERTYPE="SERVER"`, then DNS is needed to use domain-based addressing.

11.2.3 Preparing for the Configuration

After you install and configure the required protocols and services, you configure mail using the Mail Configuration application. Appendix A contains a worksheet that you can use to record the information that you need to complete the tasks in this book. If you are viewing this manual online, you can use the print feature to print a copy of this part of the worksheet.

Mail configuration consists of:

- Defining the standalone, client, or server system
- Defining the protocol information (server systems only)

11.2.3.1 General System Information

Figure 11–3 shows Part 10A of the Configuration Worksheet. The following sections explain the information you need to record in Part 10A of the worksheet.

Figure 11–3: Configuration Worksheet, Part 10A

Part 10A: Basic Mail Setup Information			
Mail server (clients only):	______		
Top domain (servers only):	______		
Mailbox directory:	☐ Local	☐ NFS client	☐ NFS server
Locking:	☐ lockf	☐ Lock file	☐ Both
Mailbox server:	______		

Mail server (clients only)

The fully qualified name of your mail server; for example, `foo.dec.com`.

Top domain (servers only)

The name of the highest level domain in your organization that uniquely identifies your organization. For example, if the server domain name is `nyc.big.com`, the top domain is `big.com`. If the server domain name is `cs.big.univ.ac.uk`, the top domain is `big.univ.ac.uk`.

Mailbox directory

The location of the mailbox directory.

For standalone and client systems, if the mailbox directory is on the local system, check Local. If it is on a remote system and is to be mounted on the local system using NFS, check NFS Client. If the local system is to export mail boxes to NFS clients, check NFS Server.

For server systems, check Server to make the mailbox directories available to other systems. If you do not want to share the mailbox directories, check Local. In this case, use the `aliases` command to send each user's mail to the appropriate system. See Section 11.7 and `aliases`(4) for more information.

Locking

The type of file locking to use on the mailbox.

For standalone and client systems, if the host with the mailbox directory is a Tru64 UNIX system, check lockf; this provides the best performance. If you are not sure what operating system the host with the mailbox directory is running, check Lock file. If you want to use both, check Both.

Note

The locking mechanism you select must match the mechanism used by the NFS server. If you are not sure how the locking mechanisms are set on the NFS server, ask the administrator of the NFS server.

For server systems, if you checked Local as the mailbox location, check lockf. If you checked Client as the mailbox location, check Lock file. If you checked Server as the mailbox location, check Both.

Mailbox server

The name of the system that exports the mailbox to your local system.

11.2.3.2 Protocol Information

Figure 11–4 shows Part 10B of the Configuration Worksheet. The following sections explain the information you need to record in Part 10B of the worksheet.

Figure 11–4: Configuration Worksheet, Part 10B

Part 10B: Mail Protocol Information

Internet (SMTP)

Forward:	☐ None ☐ Internet ☐ Nonlocal ☐ Local
Relay's host name:	________
Relay's protocol:	________
Pseudo domain:	________
Pseudo domain aliases:	________
Host aliases:	________

Others

Protocol:	☐ DECnet ☐ DECnet/OSI ☐ MTS ☐ UUCP ☐ X.25
Routing:	☐ Internet ☐ Direct ☐ Relay
Relay's host name:	________
Relay's protocol:	________
Node address (DECnet):	________
DNS name space (DECnet/OSI):	________
Pseudo domain:	________
Pseudo domain aliases:	________
Host aliases:	________

Forward

For SMTP only, the type of mail that must be forwarded to a relay. If the local host has direct access to the Internet and does not forward any mail, check None. If the local host must forward all mail addressed outside of the top domain, check Internet. If the local host must forward all messages addressed outside of the local Internet domain, check Nonlocal. If the local host must forward all mail, including local domain mail, check Local.

Relay's host name

The name of the host that will process mail for the protocol.

Relay's protocol

The name of the protocol the relay host uses to forward messages to the gateway.

Pseudodomain

An arbitrary string that specifies the pseudodomain (DECnet, DECnet/OSI, and MTS only). The pseudodomain name must be unique for each protocol and must be used consistently throughout your enterprise.

Pseudodomain aliases

Any synonyms for your pseudodomain (DECnet, DECnet/OSI, UUCP, and MTS only).

Host aliases

The alternative names that other systems might use to direct mail to your host.

If your system will serve as a gateway for other mail protocols, you will need to collect the following information:

Protocol

The type of mail protocol to support. Available protocols include the following:

- DECnet
- DECnet/OSI
- Internet Mail Protocol (SMTP) (required)
- Message Transport System (MTS)
- UUCP
- X.25 (PSInet)

Routing

For DECnet, DECnet/OSI, UUCP, MTS, and X.25 only. If mail for the particular protocol is to be forwarded over the Internet to an unspecified gateway, check Internet. The Internet depends on DNS to select an appropriate relay; therefore, do not specify a relay hostname for the Internet.

If the particular protocol is installed on this server, check Direct. If mail requiring the particular protocol is to be forwarded to another system for processing, check Relay. Complete the Relay's Hostname and Relay's Protocol fields.

Relay's host name

The name of the host that will process mail for the protocol.

Relay's protocol

The name of the protocol the relay host uses to forward messages to the gateway.

Node address

The DECnet address for this machine (DECnet only).

DNS name space

The complete DNS name space name for this node (DECnet/OSI only). The syntax of the DNS name space is as follows:

namespace:.site.nodename

11.3 Configuring Mail

Use the Mail Configuration application of the Common Desktop Environment (CDE) Application Manager to configure mail on systems with graphics capabilities. You can configure the following systems:

- Standalone systems
- Client systems
- Server systems

To start the Mail Configuration application, do the following:

1. Log in as root.
2. Click on the Application Manager icon on the CDE desktop.
3. Double-click on the System_Admin application group icon.
4. Double-click on the Configuration application group icon.
5. Double-click on the Mail Configuration application icon in the Configuration group. The Mail Configuration main window is displayed, showing available Mail service types and configured Mail service types.

To exit the Mail Configuration application, choose File then Exit. See `mailconfig`(8X) for more information.

Note

For systems without graphics capabilities, you can use the `mailsetup` utility. See `mailsetup`(8) for more information.

The Mail Configuration application has an extensive online help system that you can use, instead of the instructions in this section, to configure mail on your system.

11.3.1 Configuring a Standalone Mail System

To configure mail for a standalone system, do the following:

1. Select Standalone from the Available Mail Service Types list box in the Mail Configuration window
2. Click on the Configure button to display the Standalone Setup dialog box.
3. If your system will not use NFS to import or export the mailbox file (`/var/spool/mail`), go to step 8; the default settings are applicable for your mail configuration.

   Otherwise, click on Mailbox Setup button to display the Mailbox Setup dialog box.
4. If your system imports its mailbox using NFS, click on NFS Client for Mailbox Directory. If your system distributes its mailbox to NFS clients, click on NFS Server.
5. If you clicked on Client for Mailbox Directory, do the following:

   a. Click in the Mailbox Server field and enter the server name.
   b. Click one of the following Locking mechanisms: lockf, Lock Files, or Both.

6. If you clicked on Server for Mailbox Directory, click on Both as the Locking Mechanism setting.
7. Click on OK to complete the mailbox setup and close the Mailbox Setup dialog box.
8. Click on Commit to save the changes.
9. Click on Restart in the confirmation dialog box to restart the `sendmail` daemon and enable the changes.
10. Click on Close to close the Standalone Setup dialog box.

11.3.2 Configuring a Mail Client

To configure a mail client, do the following:

1. Select Client from the Available Mail Service Types list box in the Mail Configuration window.
2. Click on Configure to display the Client Setup dialog box.
3. Click in the Mail Server input text box and enter the name of the mail server.

4. Click on the Mailbox Setup button to display the Mailbox Setup dialog box.
5. Click on NFS Client for the Mailbox Directory.

   If your site does not use NFS to share system mailbox directories, click on Local instead of NFS Client, and then go to step 7.
6. Click in the Mailbox Server input text field and enter the name of the mailbox server.
7. Click on one of the following Locking mechanisms: lockf, Lock Files, or Both.
8. Click on OK to complete the mailbox setup and close the Mailbox Setup dialog box.
9. Click on Commit to save the changes.
10. Click on Restart in the confirmation dialog box to restart the `sendmail` daemon and enable the changes.
11. Click on Close to close the Client Setup dialog box.

11.3.3 Configuring a Mail Server

Mail servers are part of a network and process mail for clients. To configure a mail server, do the following:

1. Select Server from the Available Mail Service Types list box in the Mail Configuration window.
2. Click on the Configure button to display the Server Setup dialog box.
3. Select the mail protocol you want to configure from the Available Protocols list box. The Internet Mail Protocol (SMTP) protocol is the only required protocol configuration. Configure additional protocols as necessary.
4. Click on the Configure button to display the protocol setup dialog box for the protocol you selected.
5. For the SMTP protocol, click on the type of forwarding for this server. If you click on None, go to step 11; otherwise, go to step 7.
6. For the DECnet, DECnet/OSI, MTS, UUCP, and X.25 protocols, click on a Routing type. If you click on Internet or Direct, go to step 9. If you click on Relay, go to step 7.
7. If you are forwarding mail to another system for processing, click in the Relay's Hostname input text field and enter the name of the host that will process mail requiring this protocol; otherwise, continue with step 9.

8. If you are forwarding mail to another system for processing, click on the Relay's Protocol option menu and select the protocol used to communicate with the Relay.
9. For the DECnet, DECnet/OSI, and MTS protocols, in the Pseudo Domain text input field, enter the domain name used to identify mail that requires the selected protocol.
10. For the DECnet, DECnet/OSI, MTS, UUCP, and X.25 protocols, to add aliases for the pseudodomain, click on Pseudo Domain Aliases and do the following:
    a. Enter the alias name in the Pseudo Domain Alias text input field and click on Add.
    b. Repeat the previous step as many times as necessary.
    c. Click on Commit to close the Pseudo Domain Aliases dialog box.
11. To add aliases for this mail server, click on Host Aliases and do the following:
    a. Enter the alias name in the Host Alias text input field and click on Add.
    b. Repeat the previous step as many times as necessary.
    c. Click on Commit to close the Host Aliases dialog box.
12. For the DECnet protocol, click in the Node Address field and enter the DECnet node address (area.node) for this server, for example, `32.958`.
13. For the DECnet/OSI protocol, click in the DNS Name Space field and enter the name space of the node, which is usually the token before the colon (:) in a DECnet Phase V address.
14. Click on OK to close the Setup dialog box for the protocol you selected. The Server Setup dialog box is active.
15. Configure another protocol if necessary. Repeat steps 3 through 15 for each additional protocol.
16. Click on the Mailbox Setup button to display the Mailbox Setup dialog box.
17. Click on a Mailbox Directory.

    If your site does not use NFS to distribute the system mailbox directories, click on Local instead of NFS Server, and then go to step 19.
18. If you selected NFS Client as a Mailbox Directory, enter the name of the mail server in the Mail Server input text box. Be sure to include the domain. For example, for a server named `mailhub`, the server name with domain might be `mailhub.nyc.dec.com`.

19. Click on one of the following Locking mechanisms: lockf, Lock Files, or Both.
20. Click on OK to complete the mailbox setup and close the Mailbox Setup dialog box.
21. Click on Commit to save the changes.
22. Click on Restart in the confirmation dialog box to restart the `sendmail` daemon and enable the changes.
23. Click on Close to close the Server Setup dialog box.

11.4 Mail Utilities

The operating system includes the following mail utilities:

- The `mail`, `binmail` utility (the default) — Used by the `sendmail` utility to deliver mail locally. Because the `mail` utility has root setuid permission, it handles delivery of all mail to a user's local mailbox located in the `/var/spool/mail` directory. See the *Command and Shell User's Guide* and `mail`(1).
- The `mailx`, `Mail` utility — A combination of the Berkeley Software Distribution's (BSD) and UNIX System Laboratories, Inc.'s System V Release 4 (SVIDI) mail utilities. The `mailx` utility depends on the `binmail` utility for delivery to a user's mailbox. It has more user features than the `binmail` utility. See the *Command and Shell User's Guide* and `mail`(1).
- The `dtmail` utility — The default mail program in the Common Desktop Environment. This utility uses `sendmail` as the transport and stores information in much the same way as the `mailx` utility. In addition, it offers MIME support. See the *Common Desktop Environment: User's Guide* and `dtmail`(1).
- The message handler utility `mh` — It and its associated commands are included in the optional RAND Corporation Mail Handler subset. The message handler is composed of several shell commands where each command handles a specific function. For example, the `inc` command reads new mail and the `comp` command creates a message. Like the `mailx` utility, `mh` depends on the `mail` utility for delivery to a user's mailbox. The `mh` utility provides a graphical interface with the `xmh` command. It also provides the Post Office Protocol (POP). See `xmh`(1X) and `mail_manual_setup`(7) for more information on `xmh` and POP, respectively.

For more information on `sendmail`, see `sendmail`(8), `sendmail.cf`(4), and `sendmail.m4`(8).

11.5 Monitoring the Mail Queue

Monitoring the mail queue enables you to determine the status of several types of networking operations, including jobs that have been queued on a local system for transfer to a remote system. General users and system administrators can monitor the mail queue.

To display the contents of the mail queue, use the `mailq` command. This command lists the number of requests and the queue ID, the message size, the date the message entered the queue, and the sender and recipient for each request. Alternatively, you can also use the `sendmail –bp` command.

See `mailq`(1) for more information.

If a major host is off line for a period of time, the number of entries in the queue might be quite large, causing the performance of the mail environment to suffer. To remedy this, you will have to archive the queue. See Section 11.6 for information.

The following example shows two requests in the mail queue:

```
# mailq
               Mail Queue (2 requests)
--QID-- --Size-- -----Q-Time----- ------------Sender/Recipient------------
AA04956     1442 Tue Aug 24 10:12 <blaise>
                (Deferred)
                               <corcoran@host1.corp.com>
AA08618* (no control file)
```

11.6 Archiving the Mail Queue

When a major host is off line for a number of days, the mail queue might grow to be quite large. As a result, the `sendmail` utility spends a lot of time sorting the large queue, severely affecting the mail environment performance. Archiving the mail queue enables your mail environment to function normally while the major host is off line. To archive the mail queue, do the following:

1. Log in as root.
2. Change to the `/var/spool` directory by using the `cd` command.
3. Move the `mqueue` directory to the `old.mqueue` directory by using the `mv` command.
4. Make a new `mqueue` directory by using the `mkdir` command.
5. Change the directory's permission code to 775 by using the `chmod` command.
6. Restart the `sendmail` utility by using the following command:

   ```
   # /sbin/init.d/sendmail restart
   ```

After the major host returns on line, process the old mail queue by using the following command:

```
# /usr/sbin/sendmail -oQ/var/spool/old.mqueue -q
```

When the queue is empty, remove it by using the following command:

```
# rm -r /var/spool/old.mqueue
```

11.7 Administering and Distributing Alias Information

Depending on how you choose to administer and distribute alias information on standalone or server systems, there are two ways to provide alias information for use in the mail environment:

- `/var/adm/sendmail/aliases` file
- NIS aliases database

By default, the `/var/adm/sendmail/aliases` file permissions code is 644. This means that global users cannot change and write the changes to the file. While this creates a reasonably secure system, it leaves the maintenance of the list of global users up to the system administrator.

You can distribute responsibility for maintenance by doing the following:

1. Create a local alias file for a global maintainer in a directory. Both the file and the directory must be accessible by another maintainer.
2. Create an entry in the `/var/adm/sendmail/aliases` file that includes the additional alias file. The entry has the following form:

   alias_name`: :include:`*filename*

   The *filename* is the full path name and file name of the alias file.
3. Build a new version of the alias file by using the `newaliases` command.

See `aliases`(4) for more information.

You can also use the NIS aliases database to administer and distribute alias information for use in the mail environment. To use the NIS aliases database, do the following:

1. Install and configure NIS, if this is not already done, by using the `nissetup` script.
2. Edit the `svc.conf` file by using the `svcsetup` script, and modify the aliases entry to include `yp` (NIS).
3. Edit the NIS aliases map to include the alias information you want.

See Chapter 7 for information on configuring NIS and Section 7.4.4 for information on updating an NIS map.

11.8 Display Mail Statistics

You can display statistics about mail traffic on your system by using the `mailstats` command as follows:

```
# /usr/sbin/mailstats
```

At any time, you can initialize the statistics file by issuing the following commands:

```
# cp /dev/null /var/adm/sendmail/sendmail.st
# chmod 666 /dev/null /var/adm/sendmail/sendmail.st
```

11.9 Adding a New Mail Host

To add a new host to the mail environment, do the following:

1. Configure the network and network services. See Chapter 2 for more information.
2. If you are using DNS MX records in your environment, you must update the DNS data files. See Section 6.5 for more information.

12

Simple Network Management Protocol Agent

This chapter describes the Simple Network Management Protocol (SNMP) implementation on a Tru64 UNIX system.

12.1 SNMP Environment

The Simple Network Management Protocol (SNMP) is the de facto industry standard for managing Transmission Control Protocol/Internet Protocol (TCP/IP) networks. The protocol defines the role of a Network Management Station (NMS) and the SNMP Agent, allowing remote users on an NMS to monitor and manage TCP/IP network entities.

Note

Tru64 UNIX does not implement the NMS software.

Tru64 UNIX provides the `snmpd` daemon as the SNMP agent. This daemon is started at boot time. For information on how to set up and configure the `snmpd` daemon, see `snmp_manual_setup`(7) and `snmpd`(8).

See Appendix H for a description of the Host Resources MIB implementation.

See *Network Programmer's Guide* for information on registering applications with the SNMP agent.

Part 2

Problem Solving Information

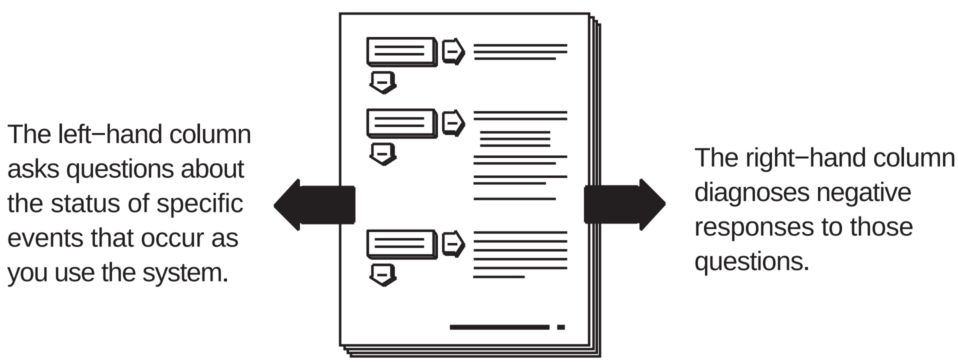
13

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 documentation to solve as many problems as possible at your level.

13.1 Using the Diagnostic Map

Network and network service problems can occur for a number of reasons. The diagnostic map in this chapter should help you isolate the problem. The following figure explains how to use the diagnostic map:



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. Each layered product has its own manual or set of manuals, which might describe solutions not provided here.

13.2 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 *System Administration* for information on starting event logging and for descriptions of the event messages.

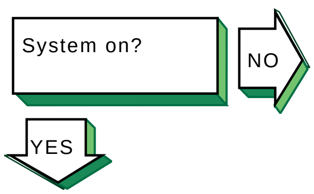
Also check the product release notes for up-to-date information on known problems.

Table 13–1 helps you identify a starting point in the diagnostic map.

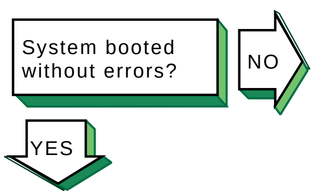
Table 13–1: Problem Solving Starting Points

If your problem is:	Start here:
`uucp` command error	Section 13.10
Network command error	Section 13.12, if using a SLIP connection Section 13.13, if using a PPP connection Section 13.3
Inaccurate system time and you are using NTP	Section 13.11
Getting host name information	Section 13.5, if you are using DNS/BIND Section 13.7, if you are using NIS
Accessing files	Section 13.9, if you are using NFS Section 13.3
Connecting to a host using LAT	Section 13.14
Unknown errors	Section 13.3
Sending or receiving mail	Section 13.15

13.3 Solving 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 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 `ypbind` 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 LAT problem, the following message is displayed:

```
getty: cannot open "/dev/ttyxx"
```

See the solutions for solving LAT problems in Section 13.14.

If your system is an NFS client and it hangs while mounting a remote file system or directory, complete the following steps:

1. Check 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 Chapter 8 for the correct format of an `fstab` entry with the `bg` option.

   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.

Network configured?

NO

Check the `/etc/rc.config` file for the following entry:

```
NUM_NETCONFIG
```

If the entry is `0`, run the Network Configuration application. See Section 2.3 for more information.

YES

Network daemons started?

NO

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
```

YES

Network reachable?

NO

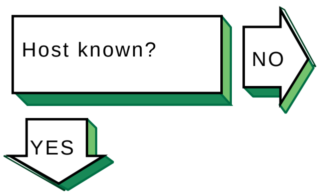
If a remote host's network is not reachable, the following message is displayed:

```
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. Check the routing tables on the local host, using the `netstat -r` command.
3. Trace the path looking at each Internet Protocol (IP) router's routing tables to make sure there is 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. Check that the local host's address-to-name translation for the remote host is correct. See the solutions for Host known?.
5. Check 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.

YES

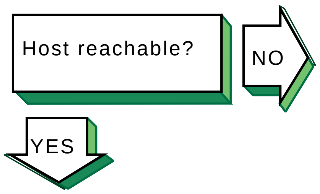


If a remote host is not known, the following message is displayed:

```
unknown host
```

Complete the following steps:

1. Check if the user is trying to reach the remote host using a valid host name.
2. Check if 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, check 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 if the DNS service has information about the remote host. See the solutions for solving DNS/BIND client problems in Section 13.5.
4. If your site uses NIS name service for name-to-address translation, check 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 solutions for solving NIS client problems in Section 13.7.
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.

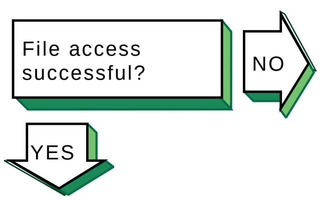


If a remote host is not reachable, the following message is displayed:

```
host is unreachable
```

Complete the following steps:

1. Check 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. Check the routing tables on the local host using the `netstat -r` command. Use the `ping` command to determine whether the IP router is reachable.
5. Check that the local host's address-to-name translation for the remote host is correct. See the solutions for Host known?.
6. Check 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.



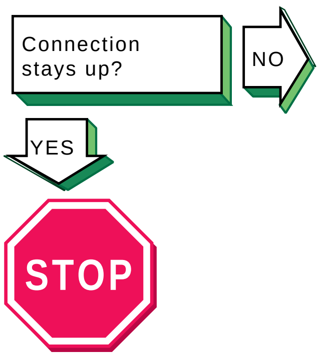
If a file cannot be accessed using the `rcp` or `rsh` commands, the following message is displayed:

```
permission denied
```

Complete the following steps:

1. Check that the user is intended to have access to the remote host. The remote host might be intentionally preventing remote access.
2. Check that the correct host and user definitions exist in the user's `.rhosts` file on the remote host.
3. Check that the `/etc/hosts.equiv` file is set up correctly.
4. Check 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, go to Section 13.9.



Problem still exists? Report your problem to Compaq. See Chapter 16.

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 14 for more information on testing the network.
2. After you identify the host with the problem, do the following:

   a. Verify that the broadcast address and address mask for the local host are properly set up in the `/etc/rc.config` file.

   b. Confirm that the network device is properly configured.

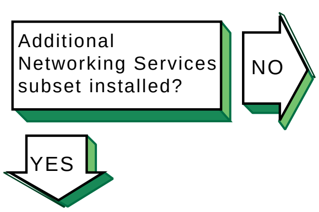
   c. Make sure the local host's `/etc/hosts` file has the correct IP address for the local host.

   d. Make sure the cabling from the local host to the network is intact and properly connected.

   e. If connected over a local area network (LAN), check the Address Resolution Protocol (ARP) entries and LAN connections.

   f. If connected over a wide area network (WAN), check WAN connections and modems.

13.4 Solving DNS/BIND Server Problems



Verify whether the Additional Networking Services subset is installed. Enter the following command:

```
# setld -i | grep OSFINET
```

The following message should be displayed:

```
OSFINETnnn installed Additional Networking Services
  (Network-Server/Communications)
```

If the subset is not installed, install it by using the `setld` command. See *System Administration* for more information on installing the subset.

DNS configured?

NO

Check the `/etc/rc.config` file for the following entry:

```
BIND_SERVERTYPE=
```

If no type is specified (double quotation marks), run the BIND Configuration application. See Section 6.3 for more information.

YES

DNS daemons started?

NO

Verify that the BIND daemon (`named`) is running. Enter the following command:

```
# ps -e | grep named
```

If no `named` process is running, start the `named` daemon, using the following command:

```
# /sbin/init.d/named start
```

YES

nslookup command returns valid information?

NO

If the `nslookup` command does not return information for any host or the host specified in the client `nslookup` command, check the `/etc/rc.config` file for the following entry:

```
BIND_SERVERTYPE=
```

If the type is:	Go to:
`CLIENT`	Stop. This system is not a DNS/BIND server and cannot provide name resolution to clients.
`PRIMARY`	Section 15.3
`SECONDARY`	Section 15.3
`SLAVE`	Section 15.4
`CACHE`	Section 15.8

YES

STOP

Problem still exists? Report your problem to Compaq. See Chapter 16.

13.5 Solving DNS/BIND Client Problems

Additional Networking Services subset installed?

NO

Verify whether the Additional Networking subset is installed. Enter the following command:

```
# setld -i | grep OSFINET
```

The following message should be displayed:

```
OSFINETnnn installed Additional Networking Services
  (Network-Server/Communications)
```

If the subset is not installed, install it by using the `setld` command. See *System Administration* for more information on installing the subset.

YES

DNS configured?

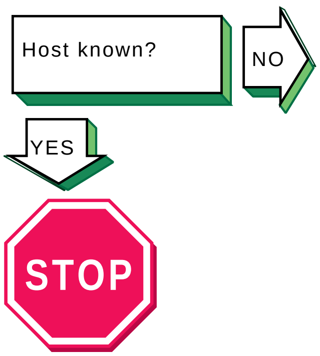
NO

Check the `/etc/rc.config` file for the following entry:

```
BIND_CONF="YES"
```

If the entry does not exist, run the BIND Configuration application. See Section 6.3 for more information.

YES



Problem still exists? Report your problem to Compaq. See Chapter 16.

If you attempted to use one of the network commands (for example, `telnet`, `rlogin`, and `rsh` commands) and the remote host is not known, the following message is displayed:

```
unknown host
```

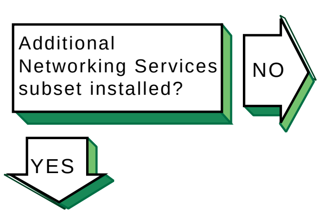
Complete the following steps:

1. Check the `/etc/svc.conf` file to determine whether DNS is being used for the `hosts` database lookup. If it is, go to step 2. If it is not, add it to the file by using the `/etc/svcsetup` command.
2. Retrieve information about the remote host with which you tried to communicate by using the `nslookup` command. Enter the following command:

   ```
   # nslookup hostname
   ```

   If the command succeeds, the client is set up correctly; try the network command again. If the command fails, go to step 3.
3. View the `/etc/resolv.conf` file and retrieve the addresses for the `nameserver` entries.
4. Verify whether the servers are reachable by using the `ping` command. If no servers are reachable, contact your network administrator. If any name server fails to respond to the `ping` command, delete the name server entry from the `resolv.conf` file.
5. Try the `nslookup` command again. If the command fails, see the solutions for solving DNS/BIND server problems in Section 13.4.

13.6 Solving NIS Server Problems



Verify whether the Additional Networking Services subset is installed. Enter the following command:

```
# setld -i | grep OSFINET
```

The following message should be displayed:

```
OSFINETnnn installed Additional Networking Services
   (Network-Server/Communications)
```

If the subset is not installed or is corrupt, install it by using the `setld` command. See *System Administration* for more information on installing the subset.

NIS configured?

NO

Check the /etc/rc.config file for the following entry:

```
NIS_CONF="YES"
```

If the entry does not exist, run the nissetup script. See Section 7.3 for more information.

YES

portmap daemon started?

NO

Verify that the portmap daemon is running. Enter the following command:

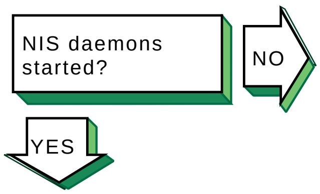
```
# ps -e | grep portmap
```

If you do not find the portmap daemon, stop and restart NIS, using the following commands:

```
# /sbin/init.d/nis stop
# /sbin/init.d/nis start
```

If the portmap daemon does not start, reboot the server.

YES



To verify that a ypserv process is running, enter the following command:

```
# ps -e | grep yp
```

If no ypserv process is running, stop and start NIS, using the following commands:

```
# /sbin/init.d/nis stop
# /sbin/init.d/nis start
```

If a ypserv process is running, execute a ypwhich command. Enter the following command:

```
# ypwhich
```

If the ypwhich command does not return an answer, find the process ID (PID) of the portmap process and kill it. Enter the following commands:

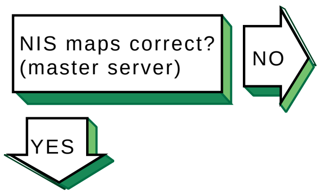
```
# ps -e | grep portmap
# kill -9 portmap_PID
```

Note

Since other network services use the portmap daemon, stopping it can affect network service. Therefore, notify your users of potential disruptions.

Stop and start NIS by using the following commands:

```
# /sbin/init.d/nis stop
# /sbin/init.d/nis start
```



Verify the information in the map. Enter the following command:

```
# ypcat map_name
```

The *map_name* variable is the name of the NIS map. If the information is incorrect, create a new map. Enter the following commands:

```
# cd /var/yp
# make map_name
```

The `make` command returns the following message:

```
map_name updated
```

If the `make` command indicates that the database is not updated, complete the following steps:

1. Remove the `database_name.time` file in the `/var/yp` and `/var/yp/domainname` directories.
2. Create a new map by using the `make` command. Enter the following commands:

   ```
   # cd /var/yp
   # make map_name
   ```

NIS maps received? (slave server)

NO

If you suspect that a slave server is not getting NIS map updates, complete the following steps on the slave server:

1. Verify that the NIS master server is running and reachable, using the `ping` command. See Section 14.1 for more information on using the `ping` command.

2. Create a `ypxfr` log file. Enter the following commands:

```
# cd /var/yp
# touch ypxfr.log
```

3. Run `ypxfr` interactively to get map updates. Enter the following command:

```
# ypxfr mapname
```

4. Check the `ypxfr.log` file. After checking the log file and resolving any problems, remove the log file. This turns logging off.

5. Verify the `ypxfr` entries in the `/var/spool/cron/crontabs/root` file. Use either the `pg` command or the `/usr/bin/crontab –l` command. The slave server entries are similar to the following:

```
# Network Information Service: SLAVE server entries
30 * * * * sh /var/yp/ypxfr_1perhour
31 1,13 * * * sh /var/yp/ypxfr_2perday
32 1 * * * sh /var/yp/ypxfr_2perday
```

6. Verify that the map has an entry in the corresponding `ypxfr` shell scripts.

7. Check the `syslogd` daemon message files for any NIS messages. See Section 14.8 for more information.

8. Verify that the slave server is in the `ypservers` map for the domain.

YES

Problem still exists? Report your problem to Compaq. See Chapter 16.

13.7 Solving NIS Client Problems

NIS configured?

NO

Check the `/etc/rc.config` file for the following entry:

```
NIS_CONF="YES"
```

If the entry does not exist, run the `nissetup` script. See Section 7.3 for more information.

YES

NIS entries in svc.conf file?

NO

Check the `svc.conf` file to be sure that it contains entries for NIS. Use the `/usr/sbin/svcsetup` script to verify this. NIS entries are indicated by the letters “yp.”

For the `passwd` and `group` databases, the Security Integration Architecture (SIA) controls whether NIS is used. However, in order to use NIS, the following characters must appear as the last line in both databases:

```
+:
```

YES

portmap daemon started?

NO

Verify that the `portmap` daemon is running. Enter the following command:

```
# ps -e | grep portmap
```

If no `portmap` daemon is running, stop and restart NIS, using the following commands:

```
# /sbin/init.d/nis stop
# /sbin/init.d/nis start
```

If the `portmap` daemon does not start, reboot the client.

YES

NIS daemons started?

NO

Verify that a `ypbind` process is running. Enter the following command:

```
# ps -e | grep yp
```

If no `ypbind` process is running, stop and start NIS, using the following commands:

```
# /sbin/init.d/nis stop
# /sbin/init.d/nis start
```

If a `ypbind` process is running, enter the `ypwhich` command:

```
# ypwhich
```

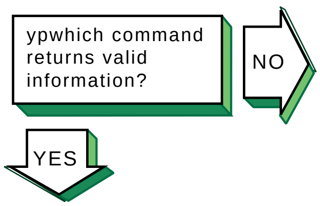
If the `ypwhich` command does not return an answer, kill the `portmap` process. Enter the following command:

```
# kill -9 portmap_PID
```

Stop and start NIS, using the following commands:

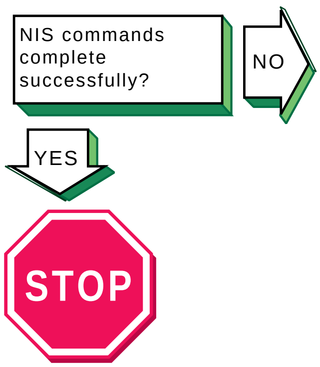
```
# /sbin/init.d/nis stop
# /sbin/init.d/nis start
```

YES



If the `ypwhich` command displays inconsistent information when invoked several times in succession, your client system is changing the server system to which it is bound. This can occur over time, especially if your system is on a busy network or if the NIS servers are busy. Once all clients get acceptable response time from the NIS servers, the system will stabilize.

If the `ypwhich` command reports that the domain is not bound, your system did not initially bind to a server system. Issue a `ypcat` command, then reissue the `ypwhich` command.



Problem still exists? Report your problem to Compaq. See Chapter 16.

If an NIS command hangs, the following message is displayed on the console:

```
yp: server not responding for domain domainname.
Still trying
```

The client cannot communicate with the server. Complete the following steps:

1. Verify that the domain name returned by the `domainname` command matches the NIS_DOMAIN variable in the server's `/etc/rc.config` file. If the domain name does not match, reconfigure the client system by using the `nissetup` script. See Section 7.3 for information on configuring NIS.
2. Verify that there is at least one NIS server for your domain running on your local subnetwork. If there is not, you must use the `–S` option to the `ypbind` command. Reconfigure the client, using the `nissetup` command, and choose this option.
3. Check with other clients on the subnetwork to determine if they are having problems with any of the NIS commands.
4. Verify that the server daemons on the remote system are running. Enter the following command:

   ```
   # rpcinfo server_name
   ```

5. Check the `syslogd` daemon message files for any NIS messages. See Section 14.8 for more information.
6. Verify that the server is running. See the solutions for solving NIS server problems in Section 13.6.

If the previous steps do not solve the problem, complete the following steps:

1. Stop and start NIS. Enter the following commands:

   ```
   # /sbin/init.d/nis stop
   # /sbin/init.d/nis start
   ```

   If this does not solve the problem, go to step 2.
2. Reboot the system.
3. Reconfigure NIS by running the `nissetup` script.

13.8 Solving NFS Server Problems

NFS subset installed?

NO

Verify whether the NFS subset is installed. Enter the following command:

```
# setld -i | grep OSFNFS
```

The following message should be displayed:

```
OSFNFSnnn  installed  NFS(tm) Utilities
  (Network-Server/Communications)
```

If the NFS subset is not installed or is corrupt, install it by using the `setld` command. See *System Administration* for more information on installing the subset.

YES

NFS configured?

NO

Check the `/etc/rc.config` file for the following entry:

```
NFSSERVING="1"
```

If the entry does not exist, run the NFS Configuration application. Be sure to specify that you will be exporting directories, even if you are going to edit the `/etc/exports` file manually. See Section 8.3 for more information.

Verify that the network software has been configured. See the solution at Network configured? in Section 13.3.

YES

portmap daemon started?

NO

Verify that the `portmap` daemon is running. Enter the following command:

```
# ps -e | grep portmap
```

If the `portmap` daemon is not running, stop and restart NFS, using the following commands:

```
# /sbin/init.d/nfs stop
# /sbin/init.d/nfs start
```

If the `portmap` daemon does not start, reboot the server.

YES

NFS daemons registered?

NO

YES

Verify that the NFS daemons are registered with the `portmap` daemon. Enter the following commands:

```
# rpcinfo -u server_name mount
# rpcinfo -u server_name nfs
```

If neither is registered, start NFS by using the following command:

```
# /sbin/init.d/nfs start
```

NFS daemons started?

NO

YES

To verify that the NFS daemons are running, complete the following steps:

1. Verify that a `mountd` process is running. Enter the following command:

   ```
   # ps -e | grep mountd
   ```

   If a `mountd` process is running, go to step 2. If no `mountd` process is running, stop and start NFS by using the following commands:

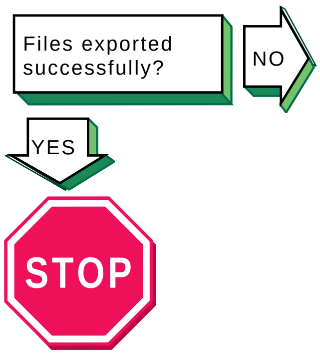
   ```
   # /sbin/init.d/nfs stop
   # /sbin/init.d/nfs start
   ```

2. Verify that an `nfsd` process is running. Enter the following command:

   ```
   # ps -e | grep nfsd
   ```

   If no `nfsd` process is running, stop and start NFS by using the following commands:

   ```
   # /sbin/init.d/nfs
   stop
   # /sbin/init.d/nfs start
   ```



Problem still exists? Report your problem to Compaq. See Chapter 16.

To verify that the files are being exported, complete the following steps:

1. Verify that file is being exported. Enter the following command:

```
# showmount −e
```

   If the file is being exported, go to step 3.

2. If the file is not being exported, check that the file has an entry in the `/etc/exports` file. If there is no entry in the `/etc/exports` file, edit the file and create an entry. Have the remote system mount the file.

3. If the file is being exported and the users cannot mount the file, check the `/etc/rc.config` file to see if they are allowed to mount the file. Enter the following command:

```
# /usr/sbin/rcmgr get NONROOTMOUNTS
```

   If the NONROOTMOUNTS parameter is 0, only users running as root can mount files from this server. To allow users not running as root to mount the files, enter the following command:

```
# /usr/sbin/rcmgr set NONROOTMOUNTS 1
```

4. Verify that the `mountd` daemon is running with Internet address checking on. Enter the following command:

```
# ps −e | grep mountd
```

   If the `−i` option is displayed, the client's name and address must be in the `/etc/hosts` file, or in the DNS or NIS `hosts` database. Only known hosts can mount the file system. If the `−d` or `−s` option is displayed, the client system must be in the same DNS domain or subdomain, respectively, as the server.

5. If the `mountd` daemon is returning stale file handles for exported files, send a hangup signal (SIGHUP) to the `mountd` daemon to force it to reread the `/etc/exports` file. Enter the following commands:

```
# ps −e | grep mountd
# kill -1 mountd_pid
```

13.9 Solving NFS Client Problems

NFS subset installed?

NO: Verify whether the NFS subset is installed. Enter the following command:

```
# setld -i | grep OSFNFS
```

The following message should be displayed:

```
OSFNFSnnn  installed  NFS(tm) Utilities
  (Network-Server/Communications)
```

If the NFS subset is not installed or is corrupt, install it by using the `setld` command. See *System Administration* for more information on installing the subset.

YES

NFS configured?

NO: Check the `/etc/rc.config` file for the following entry:

```
NFS_CONFIGURED="1"
```

If the entry does not exist, run the NFS Configuration application. See Section 8.3 for more information.

Verify that the network software has been configured. See the solution for Network configured? in Section 13.3.

YES

portmap daemon started?

NO: Verify that the `portmap` daemon is running. Enter the following command:

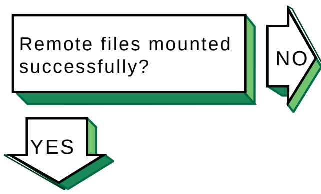
```
# ps -e | grep portmap
```

If you do not find the `portmap` daemon, stop and restart NFS, using the following commands:

```
# /sbin/init.d/nfs stop
# /sbin/init.d/nfs start
```

If the `portmap` daemon does not start, reboot the client.

YES



If the client cannot mount a remote file system or directory, complete the following steps:

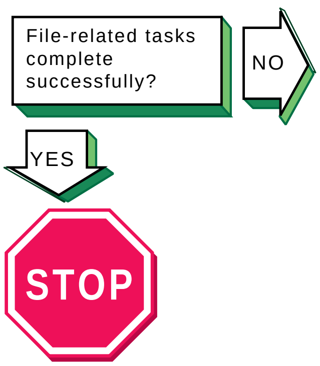
1. If an error message is displayed on the user's terminal, see Appendix E for the error message and a description.
2. Verify that the remote NFS server is on your local network and in your `hosts` database.
3. Verify that the server daemons on the remote system are running. Enter the following command:

   ```
   # rpcinfo –p server_name
   ```

4. Verify that the server is exporting the files you are expecting. Enter the following command:

   ```
   # showmount –e server_name
   ```

5. See the solutions for solving NFS server problems in Section 13.8. If the server is running and you still have problems, check the Ethernet connections and the Internet connections between the client system and the remote server.
6. Check with other clients on the network to determine if they are having problems with the remote server.
7. Verify the mount command line or the entry in the `/etc/fstab` file, and check the following:
   a. The host name matches the name of the remote NFS server.
   b. The mount point exists on your system.
8. If you get an authentication error, check the following:
   a. If you are not a superuser, the server allows nonroot mounts.
   b. Your host name is in the server's `hosts` database.
   c. If your system is not in the same domain as the server, the server performs domain checking. See `mountd`(8) for more information on server options.



Problem still exists? Report your problem to Compaq. See Chapter 16.

If application programs that perform file-related tasks do not complete their tasks or take a long time in doing so, complete the following steps:

1. If an error message is displayed on the user's terminal, see Appendix E for the error message and a description.
2. Verify that the server is running. See the solutions for solving NFS server problems in Section 13.8. If the server is running, check that the `nfsd` daemon is accumulating CPU time. If it is not, kill it and restart it. If this does not solve the problem, reboot the server. If the remote file systems or directories are mounted with the `hard` option, the program continues when the server is running once again.
3. Check with other clients on the network to determine if they are having problems with the remote server. If they are not, check the Ethernet connections and the internet connections between the client system and the remote server.
4. Check whether any `nfsiod` daemons are running. Enter the following command:

   ```
   # ps -e | grep nfsiod
   ```

   If no `nfsiod` daemons are running, start some. Enter the following command:

   ```
   # /usr/sbin/nfsiod 7
   ```

   Although the `nfsiod` daemons are not necessary for a client, they perform read-ahead and write-behind functions, which might make I/O faster.
5. If file access requests succeed but file locking requests hang indefinitely, verify that the local `rpc.statd` and `rpc.lockd` daemons are running. Enter the following commands:

   ```
   # ps -e | grep rpc.statd
   # ps -e | grep rpc.lockd
   ```

   If they are not running, start them. Enter the following commands:

   ```
   # /usr/sbin/rpc.statd
   # /usr/sbin/rpc.lockd
   ```

   Also, verify that the local `rpc.statd` and `rpc.lockd` daemons are running on the server. Enter the following commands:

   ```
   # rpcinfo -p server_name | grep status
   # rpcinfo -p server_name | grep lockmgr
   ```

   If they are not running, contact the server system administrator.

13.10 Solving UUCP Problems

uucp subset installed?

NO → Verify whether the `uucp` subset is installed. Enter the following command:

```
# setld -i | grep OSFUUCP
```

The following message should be displayed:

```
OSFUUCPnnn  installed UNIX(tm)-to-UNIX(tm)Copy
Facility (General Applications)
```

If the `uucp` subset is not installed, install it by using the `setld` command. See *System Administration* for more information on installing the subset.

YES ↓

Basic Networking Services subset installed?

NO → Verify whether the Basic Networking Services subset (containing the `tip` and `cu` utilities) is installed. Enter the following command:

```
# setld –i | grep OSFCLINET
```

The following message should be displayed:

```
OSFCLINETnnn  installed Basic Networking Services
 (Network-Server/Communications)
```

If the Basic Networking Services subset is not installed, install it by using the `setld` command. See *System Administration* for more information on installing the subset.

YES ↓

uucp utilities configured?

NO → Check for entries in the `Permissions`, `Devices`, and `Systems` files in the `/usr/lib/uucp` directory. If there are no entries, run the `uucpsetup` script. See Section 9.3 for more information.

YES ↓

Network hardware configured?

NO

YES

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:
  – Forced data set ready (DSR) is disabled.
  – Full or verbose status messages are enabled.
  – Character echo is disabled.
  – Use 8-bit characters with no parity.
  – XON/XOFF flow control is disabled.
- TCP/IP connection to remote host — Use a cable to connect your system to the network. Then, run the Network Configuration application to configure the network. See Section 2.3 for more information on setting up the network.

Dial-up to remote system successful?

NO

YES

If you cannot dial up the remote system, check the following:

1. Make sure that the setup parameters (such as speed, parity, modem control, flow control, and other terminal characteristics) on the local and remote ends are properly defined for the type of modem you have.
2. Dial the number to the remote node. If you do not get an “Attached” message or a login prompt, plug a telephone handset into the local telephone line to check for a dial tone. If you do not hear a dial tone, call you local carrier to fix this problem. If you get no message, check the cabling between the local system and the modem.
3. If you get a dial tone, check that your modem is operational and perform diagnostic tests on your modem. See the modem manual for more information.
4. From another handset, dial the local telephone line. If the local telephone rings and you can carry on a conversation, the telephone line on the local end is good. If you cannot pass voice traffic, or if there is no ring, call your local carrier to fix this problem.
5. Repeat steps 2 and 3 on the remote node to resolve problems with the remote end.
6. If the telephone line is operational, verify that the remote modem is set up to automatically answer incoming calls when the system raises the data terminal ready (DTR) signal. The system raises the DTR signal by issuing a `uugetty` or `getty` command on the port.

uucp commands complete successfully?

NO: Run the `uucp` tests to test the connection to the remote system. See Section 14.5 and Section 14.6.

YES

tip commands complete successfully?

NO: If the `tip` command does not execute successfully, complete the following steps:

1. Check that the system name, connection speed, and phone number are in the `/etc/remote` file or that the system name and connection speed are in the `/etc/remote` file and the phone number is in the `/etc/phones` file. See `remote`(4) and `phones`(4) for more information.
2. Check the `at` entry in the `/etc/remote` file. If the entry is correct, create an entry for the modem in the `/etc/acucap` file. See `acucap`(4) for more information.
3. Check that the remote system is configured to answer incoming calls.

YES

Problem still exists? Report your problem to Compaq. See Chapter 16.

13.11 Solving NTP Problems

NTP configured?

NO: Check the `/etc/rc.config` file for the following entry:

```
XNTPD_CONF="YES"
```

If the entry does not exist, run the `ntpsetup` script. See Section 10.3 for more information.

YES

NTP daemon started?

NO

Verify that an `xntpd` process is running. Enter the following command:

```
# ps -e | grep xntpd
```

If no `xntpd` process is running, start NTP by using the following command:

```
# /sbin/init.d/xntpd start
```

YES

Server found?

NO

If the `ntpq` or `xntpdc` command cannot find the server host, the following message is displayed:

```
***Can't find host hostname
```

The *`hostname`* is not in the `/etc/hosts` file, the DNS `hosts` database, or the NIS `hosts` database. Edit the `/etc/hosts` file and add an entry for the server host.

YES

Server reachable?

NO

If you run one of the monitor programs and in the output from the `peers` command the reach column contains zeros (0s), complete the following steps:

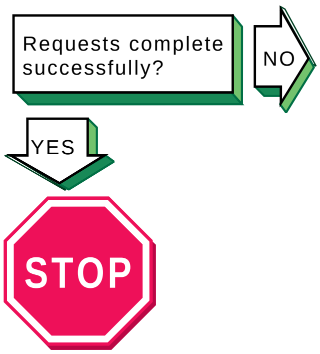
1. Contact the system administrator of the server and verify which NTP daemon the server is running. The entry for the server in the `/etc/ntp.conf` file must contain the phrase `version` *`x`* after the server name, as follows:

   ```
   server host1 version x
   ```

2. Check the `/etc/hosts` file and verify that there is an entry for each NTP server specified in the `/etc/ntp.conf` file. If you are using either DNS or NIS for host information, verify that the `hosts` database has an entry for each NTP server.

If the `xntpdc` *`hostname`* command does not display any information, check whether the *`hostname`* server is running NTP.

YES



Problem still exists? Report your problem to Compaq. See Chapter 16.

If the `ntpq` or `xntpdc` request times out, the following message is displayed:

```
hostname: timed out, nothing received
***Request timed out
```

Complete the following steps:

1. The *`hostname`* is not running the `xntpd` daemon. Check with the system administrator for that system.
2. The network connection has gone down. See the solutions for Host reachable? at the beginning of this chapter.

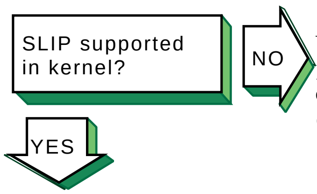
If you still cannot solve the problem, complete the following steps:

1. Check the `/etc/rc.config` file to make sure it contains an entry similar to the following:

   ```
   XNTPD_CONF="YES"
   export XNTPD_CONF
   XNTP_SERV1=server1
   export XNTP_SERV1
   XNTP_SERV2=server2
   export XNTP_SERV2
   XNTP_SERV3=server3
   export XNTP_SERV3
   XNTPD_OPTS="-g"
   export XNTPD_OPTS
   ```

   If this entry does not exist or is incorrect, run the `/usr/sbin/ntpsetup` script. See Section 10.3 for more information.
2. Check the `/etc/ntp.conf` file and make sure the information in it is accurate. It should contain entries for hosts running NTP with which you want to synchronize system time. Make sure the correct version number is specified for each server and peer. To correct any entries, run the `/usr/sbin/ntpsetup` shell script. See Section 10.3 for information on running the script.
3. Check the `daemon.log` file in the `/var/adm/syslog.dated/`*`date`* directory for information about NTP problems on the system.

13.12 Solving SLIP Problems



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, output similar to the following is displayed for each interface:

```
sl0* 296 <Link> 0 0 0 0 0
```

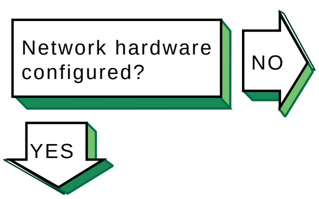
The `sl` prefix indicates that SLIP is supported on the system. In this example there is one SLIP interface.

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 megabytes of memory, SLIP is not configured into the kernel. To add SLIP into the kernel, edit the system configuration file (`/usr/sys/conf`*hostname*) and add the following entry:

```
options SL
```

See *System Administration* for more information.



Configure the network hardware as follows:

- Verify that you are using the correct hardware. See Section 4.1.2.1 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 4.1.3.1.
  - For dial-out systems, follow the guidelines in Section 4.1.3.2.

Dial in successful?
(dial-in systems)

NO

If a remote system cannot dial in to your system successfully, complete the following steps:

1. Edit the `/etc/slhosts` file and include the `debug` option in the login entry that cannot log in. See `slhosts`(4) for more information.
2. Instruct the remote user to dial in again.
3. Check the `daemon.log` file in the `/var/adm/syslog.dated/`*`date`* directory for information on SLIP problems on the dial-in system.

YES

Dial out successful?
(dial-out systems)

NO

If you cannot dial out to the remote system, complete the following steps:

1. Verify that the modem is working correctly:

   a. 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.

2. 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. Check the debug log file for information about SLIP dial-out problems.

YES

Connection
to remote system
successful?

NO

If you cannot communicate with the remote host and none of the debug messages shows an error, complete the following steps:

1. Check that the IP addresses and netmasks are correct on both ends of the connection.
2. Check 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.

YES

Connection to remote network successful?

NO

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.

YES

startslip command completes successfully?

NO

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.

YES

Problem still exists? Report your problem to Compaq. See Chapter 16.

13.13 Solving PPP Problems

PPP supported in kernel?

NO

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, output 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.

YES

PPP interface configured?

NO

Verify that the PPP interface is configured with the local and remote IP addresses and a network mask. Enter the following command:

```
# ifconfig pppx
```

The *x* variable is the interface number (0, 1, 2, ...). If the PPP interface is configured, output similar to the following is displayed:

```
ppp0: flags=cb0<POINTOPOINT,NOTRAILERS,NOARP,SIMPLEX>
```

YES

Network hardware configured?

NO

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.

YES

Connection to remote system successful?

NO

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, check the following:

- Check that the serial connection is set up successfully. Use the `chat –v` command to log the characters the `chat` program sends and receives.
- Check that `pppd` starts on the remote system. Use the `chat –v` command to log the characters the `chat` program sends and receives.
- Check 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.

YES

Network applications complete successfully?

NO

If network applications do not work successfully, this might indicate an IP-address assignment problem or a routing problem. Do the following:

1. Use the `ifconfig`, `netstat –i`, `netstat –r`, `ping`, and `traceroute` commands.
2. If you can communicate with the peer machine but cannot communicate with other other 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 see if the `ipforwarding` and `ipgateway` variables are on. If these variables are off, use the `iprsetup -s` command to turn them on.

YES

Problem still exists? Report your problem to Compaq. See Chapter 16.

13.14 Solving LAT Problems

LAT subset installed?

NO

Verify whether the Local Area Transport subset is installed. Enter the following command:

```
# setld –i | grep OSFLAT
```

The following message should be displayed:

```
OSFLATnnn installed Local Area Transport (LAT)
  (General Applications)
```

If the subset is not installed, install it by using the `setld` command. See *System Administration* for information on installing the subset.

YES

LAT configured in kernel?

NO

Verify whether the 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.

YES

LAT startup enabled?

NO

Check the `/etc/rc.config` file for the following entry:

```
LAT_SETUP="1"
```

If the entry does not exist, run the `latsetup` utility. See Section 5.3 for information.

YES

latsetup completes successfully?

NO

If `latsetup` fails while creating new LAT ttys, check 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.

YES

LAT started?

NO

Verify whether LAT has been started. Enter the following command:

```
# latcp -d
```

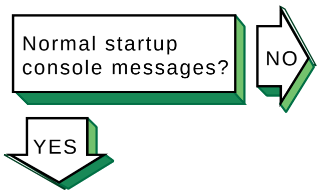
If LAT is running, the following line is displayed:

```
LAT Protocol is active
```

If LAT was not started, start it. Enter the following command:

```
# latcp -s
```

YES



If LAT starts up and messages appear continually on the system console, check 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:

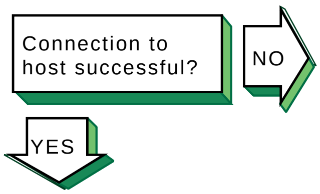
- Configure LAT into the kernel. See *System Administration* for information on configuring LAT into the kernel.
- Remove the LAT `getty` entries from the `/etc/inittab` file, either manually or by using the `latsetup` command.

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. Check if the service name is correct, using the `latcp -d` command. If the service name is incorrect, delete the service with the incorrect name. Enter the following command:

   # **latcp –D –a*service_name***

   Then, add a service with the correct name. Enter the following command:

   # **latcp –A –a*service_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 whether any group code matches a group displayed by 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 –g*list* –a*service_name***

   • Change the port characteristics at the terminal server by adding a group that matches the service.

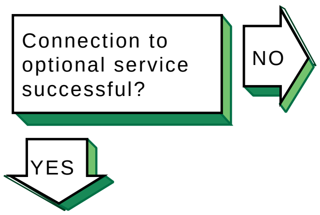
   See `latcp`(8) for more information.

3. Check whether 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. Check whether 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 –a*service_name***

   Then, add the service with the correct name and the `–o` option. Enter the following command:

   # **latcp –A –a*service_name* –o**

   See `latcp`(8) for more information.

2. Check if 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 –a*service_name***

     Then, add the service with the correct name and the `-o` option. Enter the following command:

     # **latcp –A –a*service_name* –o**

   See `latcp`(8) for more information.

Sufficient resources at host?

NO

YES

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 5.3 for information.

Host-initiated connection successful?

NO

YES

If a host-initiated connection fails, check 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 name. Enter the following command:

```
# latcp -D -pport_name
```

Then, add the application ports, using correct spelling. Enter either of the following commands:

```
# latcp -A -plocal_port -Hnode -Rrem_port
# latcp -A -plocal_port -Hnode -Vsvc_name
```

See `latcp`(8) for information.

Note

When you delete an application port for a LAT printer, any currently executing print operation continues until the printer buffer is empty. The print job might not be complete.

Printing on LAT port successful?

NO

YES

If you print a file to a printer attached to a LAT application port, the printer is online, and no printing occurs, check 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 whether 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?

NO

YES

If problems are encountered with the LAT/Telnet gateway, check the `syslogd` daemon messages file. Use the error message to eliminate the error. See Section 14.8 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 `lattelnet`, 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.

Connection stays up?

NO

If the LAT connection terminates abnormally, complete the following steps:

1. Check 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. Check the `/etc/inittab` file for duplicate LAT entries. Remove the duplicate entries, leaving the original entry.

YES

Problem still exists? Report your problem to Compaq. See Chapter 16.

13.15 Solving sendmail Problems

sendmail configured?

NO

Check the `/etc/rc.config` file for the following entry:

```
MAIL_CONF="YES"
```

If the entry does not exist, run the Mail Configuration application. See Section 11.3 for more information.

YES

sendmail daemon started?

NO

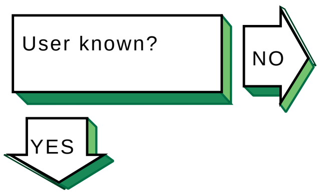
Verify whether sendmail has been started. Enter the following command:

```
# ps -e | grep sendmail
```

If `sendmail` is not running, start it, using the following command:

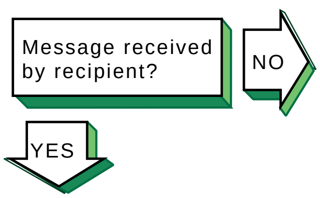
```
# sendmail
```

YES



If a user cannot send mail to another user, complete the following steps:

1. Check whether the `aliases` database was changed. If it was, update the database by using the `newaliases` command.
2. Check the `mail.log` files generated by the `syslogd` daemon for the specific mail message. If the message reached its destination, the addressee is not on the destination system. Verify that the user has the correct address. See Section 14.8 for information on viewing the `syslogd` message files.

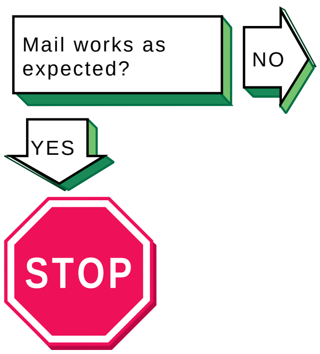


If you sent a mail message and the recipient did not receive it, complete the following steps:

1. Check whether the address is correct.
2. Check whether the remote node is reachable by using the `ping` command.
3. Look in the `mail.log` files generated by the `syslogd` daemon for the sender's user name. See Section 14.8 for information on viewing the `syslogd` message files. If you find an entry, write down the message ID. If no entry is found, send the message again.
4. Using the message ID, search through the `mail.log` files for the "from" and "to" entries. If you find a "from" entry but no "to" entry, either `sendmail` did not receive the message or the message was corrupted. Check the `/var/spool/mqueue` directory for files containing the message ID by entering the following command:

   ```
   # ls -l /var/spool/mqueue/*fAAmessage_ID
   ```

   If the `qfAA`*`message_ID`* control file is present but the (`dfAA`*`message_ID`*) data file is not present, the message was lost. If you find both a "from" entry and a "to" entry and the status is deferred, the message is in the queue. If there is no corresponding sent entry, use the `mailq` command to send the file and report the reason for the deferral. If you find both a "from" entry and a "to" entry and the status is sent, the message was delivered. If a local delivery, the message reached the destination. If a remote delivery, have the system administrator on the remote host search for the message.



If sendmail is not working correctly, complete the following steps:

1. Check the rejected message for an error message.
2. Check for error messages in the `mail.log` files generated by the `syslogd` daemon. See Section 14.8 for information on viewing the `syslogd` message file.

See Appendix G for a list of `sendmail` error messages.

Problem still exists? Report your problem to Compaq. See Chapter 16.

14

Using the Problem Solving Tools

To help you resolve problems with network hardware, the network itself, and various network services, the Tru64 UNIX system provides problem solving tools to help you do the following tasks:

- Test access to network hosts on the Internet network
- Display and modify the Internet to Ethernet translation tables
- Display a datagram's route to a network host
- Display headers of packets on the network
- Test a uucp remote connection
- Monitor a uucp file transfer
- Display the error log file
- Display the `syslogd` daemon message files

14.1 Testing Reachability of Network Hosts on the Internet Network

You test your system's ability to reach a host on the Internet network by using the `ping` command. The `ping` command has the following syntax:

/usr/sbin/ping [*options...*] *hostname*

Table 14–1 shows some of the `ping` command options.

Table 14–1: Options to the ping Command

Option	Function
`–R`	Includes the RECORD_ROUTE option in the packet and displays the route buffer on returned packets.
`–r`	Executes the `ping` command for a host directly connected to the local host. With this option, the `ping` 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.

The `ping` command sends an Internet Control Message Protocol (ICMP) echo request to the host name 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 displays the following message:

`unknown host` *`hostname`*

See `ping`(8) for more information on this command.

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 to isolate faults, you should 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
```

14.2 Displaying and Modifying the Internet to Ethernet Translation Tables

You can display and modify the Internet to Ethernet translation tables used by the Address Resolution Protocol (ARP) to help solve direct routing problems resulting from the following circumstances:

- A source host has incorrect Ethernet address information for a destination host
- Two hosts have the same host name

  Although you can modify the translation tables and change the name, you should resolve the name conflict permanently by changing one host name.

To display the entries in the Internet to Ethernet address translation tables, use the `arp` command with the following syntax to translate an Internet address to an Ethernet address:

/usr/sbin/arp *hostname*

To modify the entries in the Internet to Ethernet address translation tables, do the following:

1. Log in as root.
2. Use the `arp` command and options as follows:

   /usr/sbin/arp [*options*] *hostname*

Use the `arp` command to solve direct routing problems on an Ethernet.

See `arp`(8) for more information on this command.

The following example shows the Ethernet address for an Internet 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 trailers
```

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
```

14.3 Displaying a Datagrams'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 14–2 lists some of the traceroute command options.

Table 14–2: Options to the traceroute Command

Option	Function
`–m` *`max_ttl`*	Sets the maximum time-to-live (`ttl`) used in outgoing probe packets. The `ttl` parameter specifies the maximum number of hops a packet can take to reach its destination. The default is 30 hops.
`–n`	Displays hop addresses numerically only, rather than both numerically and symbolically.
`–p` *`port`*	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.
`–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 `traceroute` command returns an error.
`–s` *`IP_address_number`*	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 `traceroute` 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` *`type-of-service value`*	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, *Internet Protocol* for more information on types of service.
`–v`	Displays verbose output, which includes received ICMP messages other than `time exceeded` and `port unreachable`.

Table 14–2: Options to the traceroute Command (cont.)

Option	Function
–w *wait_time*	Sets the time (in seconds) to wait for a response to a probe. The default is 3 seconds.
packetsize	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. The probe packets are sent with a small `ttl` parameter, which specifies the maximum number of hops a packet can take to reach its destination.

You might see the `time exceeded` and `port unreachable` ICMP messages when using the `traceroute` command. The ICMP `time exceeded` message means that the IP router that received the probe packet cannot forward it any further due to the `ttl` value. The ICMP `port unreachable` message means that the host that received the probe packet cannot access the port intended for the probe packet.

In displaying a routing path, the `traceroute` command starts by specifying a `ttl` value of one hop, and increasing the `ttl` value by one for each probe packet it sends. The `time exceeded` message tells you which IP routers are processing the packets. The `port unreachable` message tells you that the probe packet reached its intended destination, but could not access the intended port.

The `traceroute` command sends three probe datagrams for each `ttl` setting, and displays a line showing the following:

- `ttl`
- Address of the IP router
- Round-trip time of each probe datagram

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), the command displays an asterisk (*) 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
```

14.4 Displaying Headers of Packets on the Network

You display packet headers on the network any time 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.

To display packet headers for a network interface, use the `tcpdump` command with the following syntax:

tcpdump [*options...*]

The `tcpdump` command options enable you to specify the interface on which to listen, the direction of the packet transfer, 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.

14.5 Testing a UUCP Remote Connection

You test a `uucp` remote connection to solve problems; for example, to determine why there is a backlog of transfer requests in the queue.

To test a remote connection, do the following:

1. Log in as root.
2. Change to the `/usr/lib/uucp` directory by using the `cd` command.
3. Test the remote connection by using the `uutry` command, using the following syntax:

   uutry *system_name*

   The *`system_name`* variable names the remote system to contact.
4. Examine the debugging output; the last line contains the status of the transaction. If your local system succeeds in establishing a connection to the remote system, the debugging output will contain a good deal of information. You can press Ctrl/C to stop the `uutry` shell script.

The `uutry` command has the following characteristics:

- It is a shell script stored in the /usr/lib/uucp directory.
- It contacts a remote system with debugging turned on. If you are using the uucp scheduler, uusched, to start uucico automatically at specified intervals, the uutry command overrides the retry time interval specified in the /usr/spool/uucp/.Status/*system_name* file.

  If you use the uutry command frequently, you can put the pathname to the command in the PATH entry in your .profile file.
- It directs debugging information to a file named /tmp/*system_name*, where *system_name* is the name of the local system. The uutry command then executes a tail –f command to display the file's contents.

If your local system cannot contact the remote system, do the following:

1. Check the physical connections between the local and remote systems. At both systems, check that the computer is turned on, that all the cables are properly connected, that the ports are enabled, and the modems (if being used) are working. If the remote system is not at your physical location, contact the system administrator for the remote system.
2. Check all configuration files on both systems. Verify that all entries in the Devices, Systems, and Permissions files are correct. If you are using a modem, verify all entries in the Dialers and Dialcodes files.

   If you are using a TCP/IP connection, verify that configuration files contain the correct TCP entries. Verify that the inetd daemon can start the uucpd daemon. Edit the /etc/inetd.conf file and delete the comment character (#) from the beginning of the line containing the uucp entry. Restart the inetd daemon by using the following command:

   ```
   # /sbin/init.d/inetd start
   ```

Always save the debugging output produced by the uutry command until you are certain that the problem is resolved.

The following example shows a successful test of a remote connection to system host6:

```
# /usr/lib/uucp/uutry host6
    ⋮
Conversation Complete: Status SUCCEEDED
```

The following example shows an unsuccessful test of a remote connection to system host6:

```
# /usr/lib/uucp/uutry host6
   ⋮
mchFind called (host6)
conn (host6)
getto ret -1
Call Failed: CAN’T ACCESS DEVICE
exit code 101
Conversation Complete: Status FAILED
```

14.6 Monitoring a File Transfer

Monitoring a file transfer enables you to solve other UUCP problems, especially if you can already establish a remote UUCP connection.

To monitor a file transfer, do the following:

1. Check the status of the files in the spooling directory on your local system by using the `uustat –q` command.
2. Verify that the local system can contact the remote system by using the `uutry` *system_name* command.
3. If the debugging output indicates that the connection was not successful, follow the steps described in Section 14.5.
4. Prepare a file for transfer by using the `uucp –r` command. The `–r` option instructs `uucp` to place the file in the queue without starting the `uucico` daemon.

   Start the file transfer by using the `uutry` command.

See `uutry`(1) for additional information on the `uutry` command.

The following example sends the test1 file to the system host6:

```
# uucp –r test1 host6!˜/test1
# /usr/lib/uucp/uutry host6
```

14.7 Viewing the Error Log File

You can view the binary error log file, `/var/adm/binary.errlog`, to see the contents of system events recorded there. The error log file is a data file that is read with the `uerf` command.

The events recorded in the `/var/adm/binary.errlog` file include error messages relating to the system hardware and the software kernel, as well as information about system status, startup, and diagnostics.

The `uerf` command has the following syntax:

/usr/sbin/uerf [*options...*]

The `uerf` command runs the error report formatter and displays the contents of the `/var/adm/binary.errlog` file.

You can use the `uerf` command to diagnose kernel and hardware errors.

See *System Administration* and `uerf`(8) for a complete description of this command.

14.8 Viewing the syslogd Daemon Message Files

The `syslogd` daemon records system messages into a set of files. The `syslogd` daemon starts running from the `/etc/rc.config` file when you boot the system, and whenever it receives a hangup signal. Before the `syslogd` daemon starts logging system messages, it scans the `/etc/syslog.conf` file to determine its configuration information. The configuration information determines the files into which the `syslogd` daemon logs system messages.

System messages can contain a priority code indicating the type and severity of the message. For example, system messages can indicate error conditions and warnings.

The `syslogd` daemon is available to the entire system, including binary kernel errors. See `syslogd`(8) for a complete description of the `syslogd` daemon.

To review the `syslogd` daemon log files, do the following:

1. Change your current directory to the `/etc` directory by using the `cd` command.
2. Display the contents of the `syslog.conf` file, which tells you where the `syslogd` files are kept on your system, using the `cat` command:

```
# cat syslog.conf
   #
   # syslogd config file
   #
   # facilities: kern user mail daemon auth syslog lpr binary
   # priorities: emerg alert crit err warning notice info debug
   kern.debug   /var/adm/syslog.dated/kern.log
   user.debug   /var/adm/syslog.dated/user.log
   mail.debug   /var/adm/syslog.dated/mail.log
   daemon.debug /var/adm/syslog.dated/daemon.log
   auth.debug   /var/adm/syslog.dated/auth.log
   syslog.debug /var/adm/syslog.dated/syslog.log
   lpr.debug    /var/adm/syslog.dated/lpr.log
```

```
binary.err   /var/adm/binary.errlog
msgbuf.err   /var/adm/crash/msgbuf.savecore
kern.debug   /var/adm/messages
kern.debug   /dev/console
*.emerg      *
```

3. Change your current directory to the logging directory specified in the `syslog.conf` file. In the following example, the logging directory is `/var/adm/syslog.dated/28-Oct-12:49`

   ```
   # cd /var/adm/syslog.dated/28-Oct-12:49
   ```

4. Display the list of available log files by using the `ls` command.
5. Display the contents of the log file you want to see by using the `cat` command. In the following example, the file is `daemon.log`:

   ```
   # cat daemon.log
   ```

You can use the `syslogd` daemon to help solve session layer problems such as access control problems for the Internet Protocol (IP).

15

Testing DNS Servers

In concept, testing DNS/BIND servers consists of finding out where information you are looking for is located. In practice, testing DNS servers involves tracing through a network of servers and their databases back to the server responsible for the information. This section provides the tests you use to locate the information.

Appendix I contains a worksheet for you to record information from the various tests. On a copy of the worksheet, write the current server's name, current domain name, and target domain name.

15.1 Glossary

The following terms are used in this section. Refer back to them as needed during the problem solving tests.

authoritative server

A server that stores the information that was requested locally. In contrast, a server that is not authoritative must ask other servers for information about the target host.

current server

The server you are currently logged in to and running tests from.

data types

The types of resource records in the DNS database files. See `named`(8) for a complete list and explanation.

forwarder

A server that can answer DNS queries from data in its databases and cache, whether or not it is authoritative for the information. Forwarder entries can be in the `named.boot` file.

nameserver (NS) record

Nameserver records map a domain name to a system that serves the domain, and determine whether a system is familiar with the name servers for the authoritative domain. Nameserver records have the following form:

```
domain-name          IN      NS      machine-name
```

On the left is the domain name; on the right is the name of the machine that services the domain.

primary server

A server that stores the main copy of a target domain’s databases. If the target host’s information is not in the primary server’s databases, it does not exist.

secondary server

A server that pulls a copy of the target domain’s data from another server. In most cases, the data is pulled from a primary server. However, in some cases, the data is pulled from another secondary server.

start of authority (SOA) record

Start of authority records mark the start of a zone of authority. They occur at the beginning of each master database file. SOA records have the following form:

`domain-name IN SOA machine-name`

target domain name

The portion of the target host name that begins after the first period (.).

target host

Host name you are trying to resolve. The target domain name is derived from the target host name.

15.2 Starting the DNS Server Test

To determine if the current server can resolve the target data, complete the following steps:

1. Determine whether the current server can access the target data. Use the following commands:

```
# nslookup
Default Server: host1.corp.com
Address:  127.0.0.1

> server localhost
Default Server: localhost.corp.com
Address:  127.0.0.1

> set timeout=45
> set retry=2
> target_host.target_domain.
```

If the nslookup command:	Action:
Succeeds	Go to step 3.
Fails	If the first time, go to step 2.
	If the second time, go to Section 15.3.

2. Determine whether the named daemon is running by using the following command:

```
# ps gax | grep named
```

If the named daemon is:	Action:
Running	Go to step 1.
Not running	Start the daemon by using the /sbin/init.d/named start command. If the Internet name service started message appears, go to step 1.
	If the message is not displayed, this machine is not configured as a DNS server. Decide how the machine should be configured. See Section 6.3 for more information.

3. Log in to the client system and use the nslookup command to try to access the target data.

If the nslookup command:	Action:
Succeeds	STOP. The client can resolve the target data.
Fails	The server knows the information, but is not transferring it to the client. Log out from the client; restart DNS on the server by using the /sbin/init.d/named restart command; log in to the client; and use the nslookup command. If it cannot resolve the target data, you have the wrong server or the DNS server is broken.

15.3 Determining the Server Type

To determine whether the current server is a primary server or a secondary server, complete the following steps:

1. Compare the target domain name with all domain names of the primary and secondary entries in the /etc/named.boot file. These entries have the following form:

```
primary        domain        file
secondary      domain       [host host... ]    file
```

When directed, record information in the `named.boot` file section on the worksheet.

If a `named.boot` entry:	And the first field is:	Action:
Matches the target domain name	Primary	Write the server type, domain name, and database file name on the worksheet and go to Section 15.7.
	Secondary	Write the server type, domain name, database file name, and host IP addresses on the worksheet and go to Section 15.6.
Is a subset of the target domain name	Primary	Write the server type, domain name, and database file name on the worksheet and go to step 2.
	Secondary	Write the server type, domain name, database file name, and host IP addresses on the worksheet and go to step 2.
Neither matches nor is a subset of the target domain name	Primary or secondary	Go to Section 15.4.

In the following example, the target domain name is zz.bb.cc.:

```
# cat /etc/named.boot
   .
   .
   .
primary       aa.bb.cc.                              aa.bb.cc.db    1
primary       cc.                                    cc.db        2
secondary     bb.cc.                                 bb.cc.db     3
secondary     zz.bb.cc.                              aa.bb.cc.db    4
   .
   .
   .
```

1 Not a subset of zz.bb.cc..

2 A subset of zz.bb.cc.. The server is primary for cc. domain information and stores the information in the aa.bb.cc.db. file.

3 A subset of zz.bb.cc.. The server is secondary for bb.cc. domain information and stores the information in the file bb.cc.db.

4 An exact match of zz.bb.cc.. The server is secondary for zz.bb.cc. domain information and stores the information in the zz.bb.cc.db file. Since this is an exact match, you would go to Section 15.6 and perform additional tests.

2. Compare the target domain name with all nameserver (NS) records in the database file recorded on the worksheet. When directed, record

information in the Nameservers section on the worksheet. Use the following commands to create and view a list of NS records:

```
# grep –n NS database_file > ns_list
# grep –n ORIGIN database_file >> ns_list
# sort –n ns_list > ns_list.srt
# cat ns_list.srt
```

If any NS record:	And the server is:	Action:
Contains a longer subset of the target domain name than the domain name on the worksheet	Primary or secondary	The server is neither primary or secondary for the target information. Write the names of the servers on the worksheet and go to step 3.
Does not contain a longer subset of the target domain name than the domain name on the worksheet	Primary	The database files contain the target information. Go to Section 15.7.
	Secondary	The database files contain the target information. Go to Section 15.6.

The following example shows the file created by the preceding commands. The target domain is zz.bb.cc. and the domain name from the worksheet is zz.bb.cc..

```
# cat ns_list.srt
1:$ORIGIN cc.
10:             IN         NS        server_1.cc.
17:$ORIGIN cc.
18:bb           IN         NS        server_3.bb.cc.
21:$ORIGIN cc.
22:bb           IN         NS        server_4.bb.cc.
41:$ORIGIN bb.cc.
42:zz           IN         NS        server_5.zz.bb.cc.   1
45:$ORIGIN bb.cc.
46:zz           IN         NS        server_6.bb.cc.   2
```

1 A longer subset (exact match) of the target domain. The domain name from the preceding $ORIGIN line, .bb.cc., is appended to the domain name of this line, zz, resulting in zz.bb.cc..

2 A longer subset (exact match) of the target domain. The domain name from the preceding $ORIGIN line, .bb.cc., is appended to the domain name of this line, zz, resulting in zz.bb.cc..

3. Find the IP addresses in the database file for any name servers on the worksheet. Use the following commands:

```
# grep -n ORIGIN database_file > ip_list
# grep -n server_name database_file >> ip_list
   ⋮
# sort -n ip_list > ip_list.srt
# cat ip_list.srt
```

Write the IP addresses on the worksheet next to the corresponding server name and go to Section 15.4. The following example shows the file created by the preceding commands:

```
# cat ip_list.srt
1:$ORIGIN cc.
17:$ORIGIN cc.
21:$ORIGIN cc.
41:$ORIGIN bb.cc.
42:zz          IN        NS        server_5.zz.bb.cc.
43:$ORIGIN zz.bb.cc.
44:server_5    IN        A         10.140.48.3     1
45:$ORIGIN bb.cc.
46:zz          IN        NS        server_6.bb.cc.
47:$ORIGIN bb.cc.
48:server_6    IN        A         10.12.48.3    2
```

1 The IP address for server_5.

2 The IP address for server_6.

15.4 Finding the Target Domain Information

To determine which servers the current server communicates with in order to get information for the target domain, complete the following steps:

1. Search the `named.boot` file and find any forwarder lines. Use the following command:

   ```
   # grep forwarders /etc/namedb/named.boot
   ```

   When directed, record information in the Forwarders section on the worksheet.

If your system:	Action:
Contains a forwarder line	The current server forwards requests. Write the IP addresses for any forwarders on the worksheet and go to Section 15.5.
Does not contain a forwarder line	The current server does not forward queries. Go to step 2.

2. Compare the target domain name with all nameserver (NS) records in the database file recorded on the worksheet. When directed, record information in the Nameserver section on the worksheet.

Use the following commands to create and view a list of NS records for each database file:

```
# grep -n NS database_file > ns_list
# grep -n ORIGIN database_file >> ns_list
# sort -n ns_list > ns_list.srt
# cat ns_list.srt
```

If any NS record:	And:	Action:
Contains a longer subset of the target domain name than the domain name on the worksheet	→	Write the names of the servers on the worksheet and go to step 3.
Does not contain a longer subset of the target domain name than the domain name on the worksheet	The Nameserver section on the worksheet is blank	Go to Section 15.8.

3. Find the IP addresses in the database file for any name servers on the worksheet. Use the following commands:

```
# grep -n ORIGIN database_file > ip_list
# grep -n server_name database_file >> ip_list
  ⋮
# sort -n ip_list > ip_list.srt
# cat ip_list.srt
```

   Write the IP addresses on the worksheet next to the corresponding server name and go to step 4.

4. Verify whether each server listed in the Nameserver section on the worksheet is reachable by using the `ping` command.

If a server:	And:	Action:
Responds to the `ping` command	You have root access to the server	The server is reachable and under your administrative control. Note both items on the worksheet. Go to step 5.
	You do not have root access to the server	The server is reachable, but not under your administrative control. Note both items on the worksheet. Go to step 5.

If a server:	And:	Action:
Does not respond to the `ping` command	→	Note this on the worksheet. If no servers responded to the `ping` command, STOP. The current server is isolated from its servers on the network. You cannot solve the problem; contact your enterprise network administrator.

5. Log in to each reachable server by using the `telnet` command. Each server you log in to becomes the current server. Get a new worksheet and write the current server name, current domain name, and target domain name on it. Go to Section 15.2.

15.5 Testing the Forwarders

To determine whether the forwarders prevent you from resolving the target host name, complete the following steps:

1. Determine whether each forwarder listed on the worksheet is reachable by using the `ping` command.

If a forwarder:	And:	Action:
Responds to the `ping` command	You have root access to the forwarder	The forwarder is reachable and under your administrative control. Note both items on the worksheet. Go to step 2.
	You do not have root access to the forwarder	The forwarder is reachable, but not under your administrative control. Note both items on the worksheet. Go to step 2.

If a forwarder:	And:	Action:
Does not respond to the `ping` command	→	Note this on the worksheet.
		If no forwarders responded to the `ping` command, STOP. The current server is isolated from its forwarders on the network. You cannot solve the problem; contact your enterprise network administrator.

2. Edit the `named.boot` file and eliminate any forwarders that did not respond to the `ping` command.
3. Enter the `nslookup` command again for the target host.

If the `nslookup` command:	Action:
Succeeds	Go to step 4.
Fails	Go to step 5.

4. Edit the `named.boot` file and add the forwarders removed in step 2 at the end of the forwarders line. In addition, contact the administrators of forwarders not under your administrative control and inform them that they might have a problem with their forwarder. STOP.
5. Log in to each reachable forwarder by using the `telnet` command. This forwarder is now the current server. On a new worksheet, write the current server name, current domain name, and target domain name. Go to Section 15.2.

If the forwarder or other machines:	Action:
Cannot resolve the target name	Remove the forwarder from `named.boot` file.
Can resolve the target name	STOP.

15.6 Testing Secondary Authoritative Servers

To determine whether the secondary server has the information you want, complete the following steps:

1. Find the database serial number in the start of authority record in the database file. Use the following command:

```
# head -4 database_file
```

Write the first number on the worksheet in the `named.boot` section. This is the serial number. If you have a serial number on a previous worksheet, compare the current serial number with that one. If the current number is larger, write "newer" on the worksheet. If it is smaller, write "older." In the following example, 23 is the serial number:

```
# head -4 database_file
$ORIGIN cc.
bb          IN       SOA       host1.bb.cc. postmaster.host1.bb.cc. (
            23 300 60 1209600 43200 )
            IN       MX        100 host1.bb.cc.
```

2. Determine whether the target data is contained in the database file written on the worksheet. Use the following commands to create and view a list of resource records:

```
# grep -n data_type database_file > ns_list
# grep -n ORIGIN database_file >> ns_list
# sort -n ns_list > ns_list.srt
# cat ns_list.srt
```

If the database file:	And the serial number is:	Action:
Contains the target data	Newer	The data exists in the domain. Go to step 3.
Contains the target data	Older or same	The server is broken or you made a mistake. Recheck all steps up to this point.
Does not contain the target data	→	The data does not exist in the domain. Go to step 4.

3. Determine whether the current server can access the target data. Use the following commands:

```
# nslookup
Default Server: host1.corp.com
Address:  127.0.0.1

> server localhost
Default Server: localhost.corp.com
Address:  127.0.0.1

> set timeout=45
> set retry=2
> target_host.target_domain.
```

If the `nslookup` command:	And the database serial number is:	Action:
Succeeds	→	STOP. The server is working. Either the client or server cannot communicate with this server or this server just started working.
Succeeds	Newer	Log out of the secondary server. Get the previous secondary server's worksheet and go to step 8.
Fails	→	Restart the current secondary server by using the `/sbin/init.d/named restart` command. Then reenter the `nslookup` command.

4. Verify whether each name server listed on the worksheet is reachable by using the `ping` command.

If a server:	And:	Action:
Responds to the `ping` command	You have root access to the server	The server is reachable and under your administrative control. Note both items on the worksheet.
	You do not have root access to the server	The server is reachable, but not under your administrative control. Note both items on the worksheet.
Does not respond to the `ping` command	→	Note this on the worksheet.
		If no servers responded to the `ping` command, STOP. The current server is isolated from its servers on the network. You cannot solve the problem; contact your enterprise network administrator.

Count the number of servers that responded to the `ping` command and that are under your administrative control. If the number is zero (0), go to Section 15.9.

5. Edit the `named.boot` file and find the `secondary` entry. Delete the IP address for those servers that are not reachable and are not under your administrative control. Delete those entries from the worksheet as well.

6. Log in to each reachable server by using the `telnet` command. Start a new worksheet for each server, writing the server name as the current server. Save the old worksheet.

7. Compare the target domain name with all domain names of the primary and secondary entries in the `/etc/named.boot` file. These entries have the following form:

```
primary      domain          file
secondary    domain          [host host ...]      file
```

   When directed, record information in the `named.boot` file section on the worksheet.

If a `named.boot` entry:	And the first field is:	Action:
Matches the target domain name	Primary	Write the domain name and database file name on the worksheet and go to Section 15.7.
	Secondary	Write the domain name, host IP addresses, and the database file name on the worksheet and go to step 1.
Is a subset of the target domain name	→	STOP.
Neither matches nor is a subset of the target domain name	→	STOP.

8. Restart the current secondary server by using the following command:

```
# /sbin/init.d/named restart
```

   After restarting, wait a few minutes before proceeding to the next step. This allows time for the database to be updated.

9. Determine whether the current server can access the target data. Use the following commands:

```
# nslookup
Default Server: host1.corp.com
Address:  127.0.0.1

> server localhost
Default Server: localhost.corp.com
```

```
Address:  127.0.0.1

> set timeout=45
> set retry=2
> target_host.target_domain.
```

If the nslookup command:	Action:
Succeeds	STOP. If you are in a telnet session to another secondary server, log out. Go to step 8.
Fails	If you just ended a telnet session to another server, go to step 10.
	If you did not end a telnet session, either the current server is broken and cannot read the database file or you made an error. Check all steps up to this point.

10. Compare the database serial number of the current server with the database serial number of the server from which you just logged out. Use the following command:

    ```
    # head –4 database_file
    ```

If the current database serial number is:	Action:
Older	Either the server cannot pull the database from the authoritative server or you made a mistake. Check all steps up to this point.
The same	The serial numbers cannot be equal. Check all steps up to this point.

15.7 Testing Primary Authoritative Servers

To determine whether the primary server has the information you want, complete the following steps:

1. If you are in a telnet session from a secondary server to a primary server, go to step 2. Otherwise, go to step 3.
2. Find the database serial number in the start of authority record in the database file. Use the following command:

    ```
    # head –4 database_file
    ```

    Write the first number on the worksheet in the named.boot section. This is the serial number. If you have a serial number of a previous worksheet, compare the current version number with that one. If the current number is larger, write "newer" on the worksheet. If it is smaller, write "older." If it is equal, write "same." In the following example, 23 is the serial number:

```
# head –4 database_file
$ORIGIN cc.
bb          IN       SOA       host1.bb.cc. postmaster.host1.bb.cc. (
            23 300 60 1209600 43200 )
            IN       MX        100 host1.bb.cc.
```

3. Determine whether the target data is contained in the database file written on the worksheet. Use the following commands to create and view a list of resource records:

```
# grep –n data_type database_file > ns_list
# grep –n ORIGIN database_file >> ns_list
# sort –n ns_list > ns_list.srt
# cat ns_list.srt
```

If the database file:	Action:
Contains the target data	The data exists in the domain. Go to step 4.
Does not contain the target data	The data does not exist in the domain. Go to step 5.

4. Determine whether the current server can access the target data. Use the following commands:

```
# nslookup
Default Server: host1.corp.com
Address:  127.0.0.1

> server localhost
Default Server: localhost.corp.com
Address:  127.0.0.1

> set timeout=45
> set retry=2
> target_host.target_domain.
```

If the nslookup command:	And the database serial number is:	Action:
Succeeds	→	STOP. The server is working. Either the last server cannot communicate with this server or this server just started working.
Succeeds	Older or same	STOP. The server is broken or you made a mistake. Check all steps up to this point.

If the `nslookup` command:	And the database serial number is:	Action:
Succeeds	Newer	Log out of the primary server. Get the previous secondary server's worksheet and go to Section 15.6, step 8.
Fails	→	Restart the current primary server by using the `/sbin/init.d/named restart` command. Then reenter the `nslookup` command.

5. Edit the database file and increment the database serial number by 1 to age the database. The following example shows the SOA record before and after editing. Note the serial number increase from 23 to 24.

```
# head -4 database_file
$ORIGIN cc.
bb          IN        SOA        host1.bb.cc. postmaster.host1.bb.cc. (
            23 300 60 1209600 43200 )
            IN        MX         100 host1.bb.cc.
# vi database_file
   .
   .
   .
# head -4 database_file
$ORIGIN cc.
bb          IN        SOA        host1.bb.cc. postmaster.host1.bb.cc. (
            24 300 60 1209600 43200 )
            IN        MX         100 host1.bb.cc.
```

6. Edit the database file and add new data to the database. Refer to Section 15.1 for information on valid data types. Database fields are separated by a tab character. Be sure to precede any new entry with a `$ORIGIN` entry. The following example shows a new address record for host host1.bb.cc:

```
$ORIGIN bb.cc
host1            IN        A           16.141.112.11
```

7. Restart the primary server by using the following command:

```
# /sbin/init.d/named restart
```

8. Determine whether the current server can access the target data. Use the following commands:

```
# nslookup
Default Server: host1.corp.com
Address:  127.0.0.1

> server localhost
Default Server: localhost.corp.com
Address:  127.0.0.1
```

```
> set timeout=45
> set retry=2
> target_host.target_domain.
```

If the nslookup command:	Action:
Succeeds	Log out of the primary server. Get the previous secondary server's worksheet and go to Section 15.6, step 8.
Fails	Either the server is broken or you made a mistake. Check all steps up to this point.

15.8 Tracing Information from the Root Name Server

To resolve the target name beginning with the root of the DNS namespace, complete the following steps:

1. Determine whether the current server has a cache file containing the information necessary to find a root server. Use the following command:

   ```
   # grep cache /etc/named.boot
   ```

If a cache line:	Action:
Does not exist	The current server cannot contact a root name server. Note this on the worksheet and go to step 2.
Exists	Note this on the worksheet and go to step 3.

2. Add a cache file to your server.

   Caution

   This step alters many system files. Perform the steps as shown to ensure the correct operation of your system.

   a. Create copies of specific DNS and system files. Enter the following commands:

   ```
   # cd /etc
   # cp -r namedb namedb.back
   # cp rc.config rc.config.back
   # cp hosts hosts.back
   # cp resolv.conf resolv.conf.back
   # cp svc.conf svc.conf.back
   # cd /var/adm/sendmail
   # cp sendmail.cf sendmail.cf.back
   ```

b. Display the name of the local host by using the `hostname` command. You will need to set the host name again after running the BIND Configuration application.

c. Run the BIND Configuration application. (See Section 6.3.) Modify the configuration and create a caching server. Do not start the DNS daemon automatically and do not run `svcsetup`.

d. Copy the system files back to the `/etc` directory. Use the following commands:

```
# cd /etc
# cp rc.config.back rc.config
# cp hosts.back hosts
# cp resolv.conf.back resolv.conf
# cp svc.conf.back svc.conf
```

e. Set the host name back to the original host name by using the `hostname` command.

f. Copy the `sendmail` file back to the `/var/adm/sendmail` directory and restart `sendmail`.

```
# cd /var/adm/sendmail
# cp sendmail.cf.back sendmail.cf
# /sbin/init.d/sendmail restart
```

g. Copy the DNS files back to the `/etc` directory. Use the following commands:

```
# cd /etc
# cp namedb/namedb.boot namedb.back/named.boot_new
# cp namedb/namedb.ca namedb.back
# rm -rf namedb.back namedb
# mv namedb.back namedb
# cd namedb
```

h. Edit the `named.boot` file and add the following two lines to the end of the file:

```
;
cache          .                    named.ca
```

i. Remove the `named.boot_new` file.

j. Restart the current server by using the `/sbin/init.d/named restart` command.

3. Display the `named.ca` file by using the following command:

```
# cat named.ca
```

Write the root name server names and their IP addresses in the Root nameservers section on the worksheet.

4. Verify whether each root name server listed on the worksheet is reachable by using the `ping` command.

If a root name server:	Action:
Responds to the `ping` command	Note this on the worksheet. Go to Section 15.10.
Does not respond to the `ping` command	Note this on the worksheet. If no servers responded to the `ping` command, go to step 5.

5. Do either of the following:

   - Give the current server access to the Internet. Then restart the `named` daemon by using the following command:

     ```
     # /sbin/init.d/named restart
     ```

     Keep the same current server and worksheet, and go to Section 15.2.

   - Add a forwarder entry to direct the current server to communicate with a machine with access to the Internet. Then restart the `named` daemon by using the following command:

     ```
     # /sbin/init.d/named restart
     ```

     Keep the same current server and worksheet, and go to Section 15.2.

15.9 Resolving Target Data

To resolve target data using a name server, complete the following steps:

1. Enter the `nslookup` command for the target system. Choose the first name server from either the Root nameserver section or the Nameserver section. Use the following commands:

   ```
   current_server> nslookup
   Default Server:  localhost.zk3.dec.com
   Address:  127.0.0.1

   > server IP_address
   Default Server:  [IP_address]
   Address:  128.102.16.10

   > set type data_type
   > target_name
   ```

If the `nslookup` command:	And:	Action:
Succeeds	→	STOP. The server is working. Either the last server you checked out does not talk to this one or this server just started working. Recheck all steps completed up to this point.
Fails	An error message is returned	If a “non-existent domain” message is displayed, no data exists for the *target_name*. Go to Section Section 15.10. If a “no information available” message is displayed, the *target_name* exists, but the information you want is not associated with it. If this information should exist, contact the target domain administrator. If a “timed-out” message is displayed, the server to which you sent the query cannot contact the server that knows the information. Go to step 2.
Fails	An error message is not returned	An unknown error. Contact the target domain administrator.

2. Modify the retry and timeout values and enter the `nslookup` command again. Enter the following commands:

```
current_server> nslookup
Default Server:  localhost.zk3.dec.com
Address:  127.0.0.1

> server IP_address
Default Server:  [IP_address]
Address: IP_address

> set type data_type
> target_name
```

If the `nslookup` command:	And:	Action:
Succeeds	→	STOP. The server is working, but is slow. This might prevent the query from being resolved. Check the network connection to the server. If that is correct, wait for the performance to improve. If it does not improve, contact the name server administrator.
Fails	An error message is returned	If a "non-existent domain" message is displayed, no data exists for the *target_name*. Go to Section 15.10. If a no information available message is displayed, the *target_name* exists, but the information you want is not associated with it. If this information should exist, contact the target domain administrator. If a "timed-out" message is displayed, the server to which you sent the query cannot access the server that knows the information. Select another name server from the worksheet and go to step 1.
Fails	An error message is not returned	An unknown error. Contact the target domain administrator.

15.10 Finding the First Nonexistent Domain

To find the first nonexistent domain in a target name, complete the following steps:

1. Enter the `nslookup` command, using the smallest subset of the target domain name. Enter the following commands:

```
current_server> nslookup
Default Server:  localhost.zk3.dec.com
Address:  127.0.0.1

> server IP_address
Default Server:  [IP_address]
Address: IP_address

> set type=ns
> target_name_subset
```

For example, if the target domain name is zz.bb.cc., the first attempt is to resolve the target name subset cc.. If necessary, the second attempt uses bb.cc., and the third, zz.bb.cc..

If the `nslookup` command:	And:	Action:
Succeeds	→	Go to step 3.
Fails	An error message is returned	If a “non-existent domain” message is displayed, no data exists for the *target_name*. If the information should exist, contact the domain administrator and request that the information be added to the domain. STOP.
		If a “timed-out” message is displayed, go to step 2. This should not happen because the server worked well before.

2. Modify the retry and timeout values and enter the `nslookup` command again. Enter the following commands:

```
current_server> nslookup
Default Server:  localhost.zk3.dec.com
Address:  127.0.0.1

> server IP_address
Default Server:  [IP_address]
Address: IP_address

> set retry=2
> set timeout=45
> set type=ns
> target_name_subset
```

If the `nslookup` command:	And:	Action:
Succeeds	→	Go to step 3.
Fails	An error message is returned	If a “non-existent domain” message is displayed, no data exists for the *target_name*. If the information should exist, contact the domain administrator and request that the information be added to the domain. STOP. If a “timed-out” message is displayed, select another name server from the worksheet and go to Section 15.9.

3. Add the next part of the target domain name to the target subset and go to step 1.

16

Reporting Network Problems

If you are unable to solve a critical problem with the network or network service, you should complete the following steps:

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. Check if the product is still under warranty or if 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 Compaq service representative and describe your problem.
4. If you are requested to supply any information pertaining to the problem, gather the necessary information and submit it.

16.1 Gathering Information

You might be asked to submit some of the information that is listed in the sections that follow. This information 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.

16.1.1 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.

16.1.2 Hardware Architecture

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
  ```

- 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 xdpyinfo: >> /system.information
  # xdpyinfo >> /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
  ```

16.1.3 Software Architecture

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 /sbin/rc{0,2,3} >> /system.information
  # pr /sbin/init./* >> /system.information
  ```

- A description of the layered products installed

A

Configuration Worksheet

This appendix contains the worksheet for you to photocopy and fill in before performing the configuration tasks described in this manual. Read the "Planning" section of each chapter and, where appropriate, fill in the blanks. You might want to use a worksheet for each system that you are setting up.

Figure A–1: Configuration Worksheet, Part 1A

Part 1A: Interface and Daemon Information

All Network Interfaces

Adapter name: ____________________ ____________________

Host name: ____________________ ____________________

IP address source: DHCP server ☐ User supplied ☐

Internet address: ____________________ ____________________

Network mask: ____________________ ____________________

Token Ring interface

Adapter speed: ____________________

rwhod daemon

rwhod: Yes ☐ No ☐

Flags: Broadcast only ☐ Listen only ☐ Both ☐

routed daemon

routed: Yes ☐ No ☐

Flags: ☐ Run routed on gateway host

☐ Write all packets to standard output

☐ Log additional information

RIP data: Supply ☐ Run quietly ☐

Gateways file

Destination type: Net ☐ Host ☐

Destination: ____________________

Gateway: ____________________

Hop count: ____________________

Route type: External ☐ Passive ☐ Active ☐

gated daemon

gated: Yes ☐ No ☐

Configuration file: ____________________

IP router

IP router: Yes ☐ No ☐

Figure A–2: Configuration Worksheet, Parts 1B and 2A

Part 1B: Network Files Information

Static routes file

Destination type: Default gateway ☐ Host ☐ Network ☐

Destination: ____________

Route via: Gateway ☐ Interface ☐

Gateway: ____________

Hosts file

Host name: ____________ ____________ ____________

Internet address: ____________ ____________ ____________

Alias: ____________ ____________ ____________

hosts.equiv file

Host name: ________ ________ ________ ________

User name: ________ ________ ________ ________

Networks file

Network name: ____________ ____________ ____________

Network address: ____________ ____________ ____________

Alias: ____________ ____________ ____________

Part 2A: DHCP Server/Security Parameters

BOOTP address from pool: True ☐ False ☐

BOOTP compatibility: True ☐ False ☐

Default lease time: ____________

Name service: `/etc/hosts` ☐ DNS ☐ NIS ☐

Ping timeout: ____________

Provisional time to live: ____________

Restrict to MAC address: True ☐ False ☐

IP ranges

Subnetwork address: ____________

DHCP server: ____________

IP ranges: ____________ ____________

____________ ____________

Host name lists

Domain name: ____________

DHCP server: ____________

Host name prefix: ____________

Host names: ____________ ____________

____________ ____________

Figure A–3: Configuration Worksheet, Part 2B

Part 2B: Basic DHCP Parameters

Configuration type: Node ☐ Subnet ☐ Group ☐

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 ☐ False ☐

Subnetwork mask: ____________

TFTP root directory: ____________

Broadcast address: ____________

Subnetworks are local: True ☐ False ☐

Supply masks: True ☐ False ☐

DHCP rebinding time: ____________

DHCP renewal time: ____________

Lease time: ____________

Figure A–4: Configuration Worksheet, Parts 3A and 3B

Part 3A: SLIP Setup

Type of connection: ☐ Hardwired ☐ Modem
Type of system: ☐ Dialin ☐ Dialout
Local IP address: ____________
Network mask: ____________
Destination IP address: ____________
Terminal name: ____________
Speed: ____________
SLIP login information: ____________

Dialout systems

startslip subcommands: ________________________
____
____
____

Dialin systems

slhosts file options: ________________________
Gateway: ☐ Yes ☐ No

Part 3B: PPP Setup

Local IP address: ____________
Remote IP address: ____________
Network mask: ____________
Terminal name: ____________
Speed: ____________
Level of authentication: ____________
Type of authentication: ☐ PAP ☐ CHAP
Options: ________________________

Figure A–5: Configuration Worksheet, Parts 4 and 5

Part 4: LAT Setup

Start LAT automatically at boot time: Yes ☐ No ☐

Type of tty devices: ____________

Number of LAT tty devices: ____________

Number of LAT entries (getty) in `/etc/inittab`: ____________

Part 5: DNS Setup

Local domain name: ____________

Server

Host name resolution order: ☐ `/etc/hosts` ☐ DNS

Zones

Zone domain name:	Authority:	Data file or server address:
____________	☐ Primary ☐ Secondary	____________
____________	☐ Primary ☐ Secondary	____________
____________	☐ Primary ☐ Secondary	____________
____________	☐ Primary ☐ Secondary	____________

Forwarders

Forwarder name: ____________ ____________

Client

Server name:	Internet address:
____________	____________
____________	____________
____________	____________
____________	____________

Host name resolution order: ☐ `/etc/hosts` ☐ DNS

Figure A–6: Configuration Worksheet, Part 6

Part 6: NIS Setup

Domain name: ____________________

Master Server

/etc/files for maps: __________ __________

__________ __________

__________ __________

__________ __________

/var/yp/src/mail.alias file: ☐ Yes ☐ No

/ver/yp/src/netgroup file: ☐ Yes ☐ No

Setup options: ____________________

Slave name: __________ __________

IP address: __________ __________

Slave name: __________ __________

IP address: __________ __________

Slave Server

Setup options: ____________________

Master name: __________ __________

IP address: __________ __________

Server name: __________ __________

IP address: __________ __________

Server name: __________ __________

IP address: __________ __________

Client

Setup options: ____________________

Server name: __________ __________

Server name: __________ __________

Figure A–7: Configuration Worksheet, Part 7

Part 7: NFS Setup

Server

Number of nfsd threads: TCP: ________ UDP: ________

NFS locking: ☐ Yes ☐ No

PCNFS daemon: ☐ Yes ☐ No

Allow nonroot mounts: ☐ Yes ☐ No

Path name:	Permissions:	Network group/ Node name:

Client

Number of I/O threads: ________

NFS locking: ☐ Yes ☐ No

Automount: ☐ Yes ☐ No

Remote server name:		
Directory path:		
Local mount point:		
Readonly mount:	☐ Yes ☐ No	☐ Yes ☐ No

Figure A–8: Configuration Worksheet, Parts 8A and 8B

Part 8A: UUCP Setup

Connections

Type of connection: ☐ Modem ☐ Direct link ☐ TCP/IP

Modems:

Modem type: ____________________

Baud rate: ____________________

Device name: ____________________

`/etc/inittab` entry ID: ____________________

Direct links:

Remote system name: ____________________ ☐ Direct

Baud rate: ____________________

Device name: ____________________

`/etc/inittab` entry ID: ____________________

TCP/IP:

Outgoing connections: ☐ Yes ☐ No

Incoming connections: ☐ Yes ☐ No

Part 8B: UUCP Setup

Outgoing System

Remote system name: ____

Mode of connection: ☐ Modem ☐ Direct link ☐ TCP/IP

For TCP/IP, conversation protocol: ☐ g ☐ t ☐ e ☐ f

Calling times: ____

Baud rate: ____________________ Any ☐

Phone number (for modem): ____

Login ID: ____

For modem/direct links, expect send string:

☐ Carriage returns
☐ None
☐ Prompt

Figure A–9: Configuration Worksheet, Parts 8C and 9

Part 8C: UUCP Setup

Incoming System

Remote system name: ______________________

Local system name: ______________________

Login ID: ______________________

Alternative login ID: ______________________

Options: ______________________

REQUEST option: ☐ Yes ☐ No

SENDFILES option: ☐ Yes ☐ No

Additional READ/WRITE locations: ______________________

Additional NOREAD/NOWRITE locations: ______________________

Commands: ______________________

____

____

VALIDATE option: ☐ Yes ☐ No

CALLBACK option: ☐ Yes ☐ No

Phone number (for modem): ______________________

Part 9: NTP Setup

Server

Time source: ______________________

Server Internet address:	Server name:	NTP version:

Client

Local NTP server address:	Server name:	NTP version:

Figure A–10: Configuration Worksheet, Parts 10A and 10B

Part 10A: Basic Mail Setup Information

Mail server (clients only): ____________

Top domain (servers only): ____________

Mailbox directory: ☐ Local ☐ NFS client ☐ NFS server

Locking: ☐ lockf ☐ Lock file ☐ Both

Mailbox server: ____________

Part 10B: Mail Protocol Information

Internet (SMTP)

Forward: ☐ None ☐ Internet ☐ Nonlocal ☐ Local

Relay's host name: ____________

Relay's protocol: ____________

Pseudo domain: ____________

Pseudo domain aliases: ____________

Host aliases: ____________

Others

Protocol: ☐ DECnet ☐ DECnet/OSI ☐ MTS ☐ UUCP ☐ X.25

Routing: ☐ Internet ☐ Direct ☐ Relay

Relay's host name: ____________

Relay's protocol: ____________

Node address (DECnet): ____________

DNS name space (DECnet/OSI): ____________

Pseudo domain: ____________

Pseudo domain aliases: ____________

Host aliases: ____________

B

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.

B.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
- Carries 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.

B.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.

B.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.

B.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 beaconing sent a frame to the directed beacon multicast address, indicating the suspected cause of the ring break. (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.) 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.

B.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.

B.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 tailures
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
```

B.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.

B.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. The following functional addresses are defined:

- Active monitor
- Ring parameter server
- Ring error monitor
- Network manager
- Netbios
- Bridge

`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.

C

Writing automount Maps

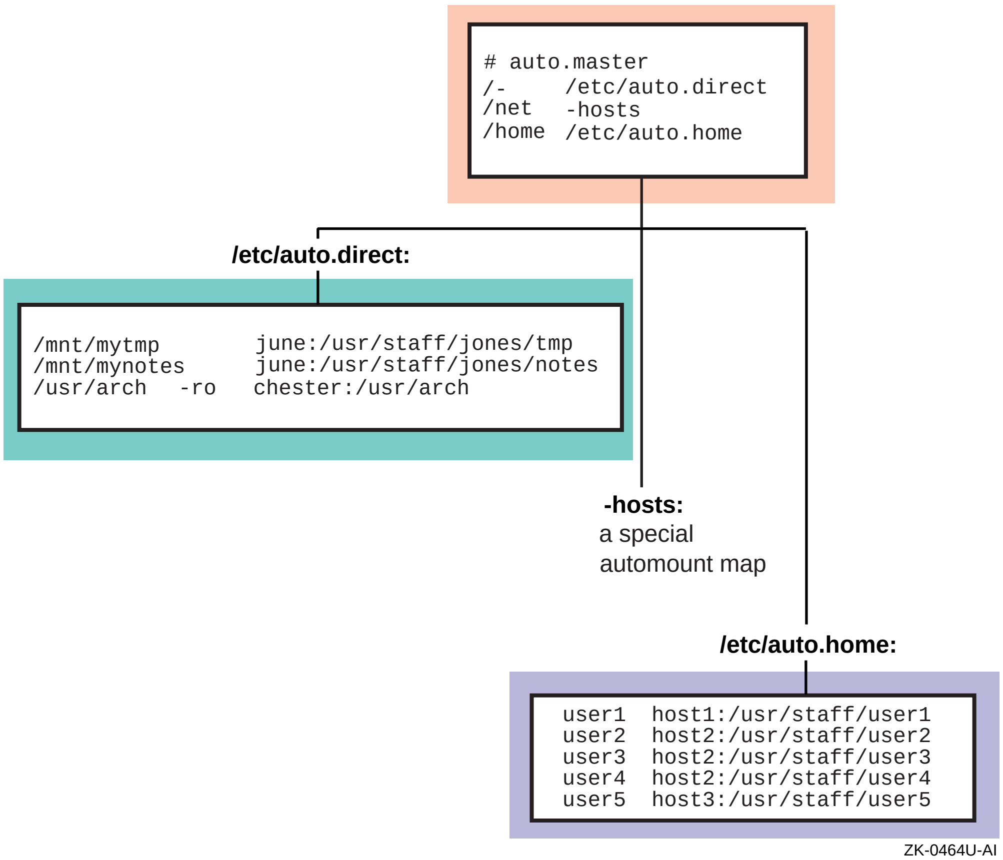
There are three types of automount maps:

- Master
- Direct
- Indirect

The automount maps can be written in a variety of ways. Maps can be direct or indirect. They can be simple or can use multiple mounts, shared mounts, replicated file systems, or any combination of the three. As discussed in Section C.1, indirect maps can be written to reduce redundancy by using substitution characters and pattern matching. The examples in this section illustrate how the same maps can be rewritten in a number of ways.

Figure C–1 illustrates an auto.master map that points to the /etc/auto.direct direct map, the built-in –hosts map, and the /etc/auto.home indirect map. Each map to which the auto.master map points is expanded to show its sample contents. Note that all of the information contained in the master map can be specified on the command line. The master map, however, simplifies organization and administration of automount.

Figure C–1: Sample automount Maps



The following examples show how the `/etc/auto.direct` map in Figure C–1 can be rewritten using multiple mounts (Example C–1); multiple mounts and shared mounts (Example C–2); and multiple mounts, shared mounts, and replicated file systems (Example C–3).

Example C–1: Multiple Mounts in a Direct Map

```
/mnt/mytmp                              june:/usr/staff/jones/tmp
/mnt/mynotes                            june:/usr/staff/jones/notes
/usr/arch       /             -ro       chester:/usr/arch \
                /bsd          -ro       chester:/usr/arch/bsd \
                /standards    -ro       chester:/usr/arch/standards \
                /dec/uws      -ro       chester:/usr/arch/dec/uws \
                /dec/ultrix   -ro       chester:/usr/arch/dec/ultrix
```

Example C–2: Multiple Mounts and Shared Mounts in a Direct Map

```
/mnt/mytmp                                 june:/usr/staff/jones:tmp
/mnt/mynotes                               june:/usr/staff/jones:notes
/usr/arch       /             –ro   chester:/usr/arch \
                /bsd          –ro   chester:/usr/arch/bsd \
                /standards    –ro   chester:/usr/arch/standards \
                /dec/uws      –ro   chester:/usr/arch/dec/uws \
                /dec/ultrix   –ro   chester:/usr/arch/dec/ultrix
```

Example C–3: Multiple Mounts, Shared Mounts, and Replicated File Systems in a Direct Map

```
/mnt/mytmp                                 june:/usr/staff/jones:tmp
/mnt/mynotes                               june:/usr/staff/jones:notes
/usr/arch       /             –ro   chester:/usr/arch \
                /bsd          –ro   chester:/usr/arch/bsd \
                                    bazel:/src/bsd \
                /standards    –ro   chester:/usr/arch/standards \
                /dec/uws      –ro   chester:/usr/arch/dec/uws \
                                    fiesta:/archive/uws\
                /dec/ultrix   –ro   chester:/usr/arch/dec/ultrix
```

The `/etc/auto.direct` maps in the preceding examples could be rewritten as indirect maps. If the `/etc/auto.direct` map is rewritten to be an indirect map, the entry pointing to it in the `auto.master` map might read:

```
/mnt     /etc/auto.indirect
```

Rewritten as a simple indirect map (`/etc/auto.indirect`), the `/etc/auto.direct` map in Figure C–1 would read as shown in Example C–4.

Example C–4: Simple Indirect Map

```
mytmp           june:/usr/staff/jones/tmp
mynotes         june:/usr/staff/jones/notes
arch    –ro     chester:/usr/arch
```

Note that the key is a simple pathname.

The following examples illustrate that indirect maps can also be rewritten using multiple mounts (Example C–5); multiple mounts and shared mounts (Example C–6); and multiple mounts, shared mounts, and replicated file systems (Example C–7).

Example C–5: Multiple Mounts in an Indirect Map

```
mytmp                                 june:/usr/staff/jones/tmp
mynotes                               june:/usr/staff/jones/notes
arch            /             −ro     chester:/usr/arch \
                /bsd          −ro     chester:/usr/arch/bsd \
                /standards    −ro     chester:/usr/arch/standards \
                /dec/uws      −ro     chester:/usr/arch/dec/uws \
                /dec/ultrix   −ro     chester:/usr/arch/dec/ultrix
```

Example C–6: Multiple Mounts and Shared Mounts in an Indirect Map

```
mytmp                                 june:/usr/staff/jones:tmp
mynotes                               june:/usr/staff/jones:notes
arch            /             −ro     chester:/usr/arch \
                /bsd          −ro     chester:/usr/arch/bsd \
                /standards    −ro     chester:/usr/arch/standards \
                /dec/uws      −ro     chester:/usr/arch/dec/uws \
                /dec/ultrix   −ro     chester:/usr/arch/dec/ultrix
```

Example C–7: Multiple Mounts, Shared Mounts, and Replicated File Systems in an Indirect Map

```
mytmp                                 june:/usr/staff/jones:tmp
mynotes                               june:/usr/staff/jones:notes
arch            /             −ro     chester:/usr/arch \
                /bsd          −ro     chester:/usr/arch/bsd \
                                      bazel:/src/bsd \
                /standards    −ro     chester:/usr/arch/standards \
                /dec/uws      −ro     chester:/usr/arch/dec/uws \
                                      fiesta:/archive/uws\
                /dec/ultrix   −ro     chester:/usr/arch/dec/ultrix
```

The −hosts map is a built-in map supplied by automount. This map allows a client to access directories that are exported from any host in its hosts database. The location of the hosts database that your system uses is determined by the services running on your system (BIND, NIS, local) and how those services are specified in the `/etc/svc.conf` file. References to a particular host name result in all of the file systems that are exported from that host being mounted on the local system. For example, the following command results in all of the file systems that are exported from `host1` being mounted on the local system:

```
# cd /net/host1
```

The `/etc/auto.home` map shown in Figure C–1 is an indirect map that allows users to remote mount their home directories. It can be rewritten using the ampersand (&) and asterisk (*) substitution characters.

The following example shows how the `/etc/auto.home` map in Figure C–1 can be rewritten using ampersands (&):

```
user1  host1:/usr/staff/&
user2  host2:/usr/staff/&
user3  host2:/usr/staff/&
user4  host2:/usr/staff/&
user5  host3:/usr/staff/&
```

C.1 Substitution and Pattern Matching

The `automount` daemon recognizes the following substitution characters, allowing you to eliminate redundancy within `automount` maps:

- The ampersand (&) can be used in both direct and indirect maps; however, it is most efficient and easily understood when used in indirect maps.
- The asterisk (*) can be used in indirect maps only.

Because the ampersand and asterisk are most easily used in indirect maps, this section discusses them in the context of indirect maps only. Recall that lines in indirect maps have the following syntax:

```
key                mount-options              location
```

Whenever the `automount` daemon encounters an ampersand (&) in a line of an indirect map, it substitutes the `key` in that line for the ampersand (&).

The following example is an indirect map that does not use ampersands:

```
#key                   mount-options      location
#
host1                  –rw,nosuid        host1:/home/host1
host2                  –rw,nosuid        host2:/home/host2
```

Using the ampersand (&) as a substitution character, the entries read as follows:

```
#key              mount-options     location
#
host1             –rw,nosuid        &:/home/&
host2             –rw,nosuid        &:/home/&
```

You can use the asterisk (*) to substitute for lines that are all formatted similarly. The `automount` daemon uses the asterisk to match any host not listed as a key in an entry before the asterisk. The following is a typical use of the asterisk (*):

```
#key              mount-options     location
#
host1             –rw,nosuid        &:/home/&
host2             –rw,nosuid        &:/home/&
*                 –rw,nosuid        &:/home/&
```

Suppose a user enters the following command:

```
% ls /home/host5
```

The `automount` daemon substitutes the host name (host5) as the `key`. After it has substituted host5 for the `key`, it then substitutes host5 for each of the ampersands in the `location` field as well. The `automount` daemon translates the preceding command into the following:

```
#key              mount-options     location
#
host5             –rw,nosuid        host5:/home/host5
```

Note

The `automount` daemon ignores any entry that follows an asterisk.

C.2 Environment Variables

You can use the value of an environment variable in a map by adding a dollar sign ($) prefix to its name. You also can use braces ({ }) to delimit the name of the variable from appended letters or digits.

Environment variables can be inherited from the environment or can be defined explicitly with the `–D` option on the command line. For example, you can invoke the `automount` daemon with the `HOST` variable by entering the following command:

```
# automount –D HOST= hostname
```

The following is an example of a direct map entry that uses the environment variable HOST to define subnetworks:

```
/mydir     -rw      server:/export/$HOST
```

C.3 Mounting File Systems

The automount daemon provides several ways to mount remote directories and file systems:

- Multiple mounts
- Shared mounts
- Replicated file systems

C.3.1 Multiple Mounts

When you write direct and indirect maps, you can specify that different directories within a file system hierarchy be mounted from different servers. For example, if you mount the /usr/local file system on your machine, you can mount the various subdirectories within /usr/local from different servers.

The following example shows an entry in a direct map in which the directories /usr/local/bin, /usr/local/src, and /usr/local/tools are mounted from the machines host1, host2, and host3, respectively:

```
/usr/local\
                /bin    -ro        host1:/usr/local/bin \
                /src    -ro        host2:/usr/local/src \
                /tools  -ro        host3:/usr/local/tools
```

This is a direct map because the key, /usr/local, is an absolute pathname. If this were an entry in an indirect map, the key would be a simple pathname, such as local. The key, /usr/local, comprises three subdirectories, each of which is a mount point for a remote directory on a different remote server. The example shows the entry split into four lines with the continuation lines indented for readability.

The preceding example shows multiple, nonhierarchical mounts under /usr/local. The following example shows a true hierarchical entry:

```
/usr/local \
             /            -ro        host0:/usr/local \
             /bin         -ro        host1:/usr/local/bin \
             /src         -ro        host2:/usr/local/src \
             /tools       -ro        host3:/usr/local/tools
```

The mount points used here for the hierarchy are /, /bin, /src, and /tools. Note that these mount points are relative to /usr/local. The mount point / mounts /usr/local from host0.

When file systems are mounted hierarchically, the entire hierarchy is treated as one object. Each file system is mounted on a subdirectory within another file system, and when a subdirectory within the hierarchy is referenced, the automount daemon mounts the entire hierarchy. The entire hierarchy is also unmounted as one object.

C.3.2 Shared Mounts

When multiple directories within the same remote directory are mounted, the location field can be specified as follows:

```
host:path:subdir
```

The host field is the remote host from which to mount the file system. The path field is the pathname of the directory to mount, and the subdir field, if specified, is the name of the subdirectory to which the symbolic link is made. This prevents duplicate mounts of the same remote file system when multiple subdirectories within it are accessed. Suppose an indirect map called /auto.myindirect has the following entries:

```
mybin           host1:/usr/staff/diane:bin
mystuff         host1:/usr/staff/diane:stuff
```

When a user accesses a file in /auto.myindirect/mybin, the automount daemon mounts host1:/usr/staff/diane, but creates a symbolic link called /auto.myindirect/mybin to the .L bin) subdirectory in the temporarily mounted file system. If a user immediately tries to access a file in /auto.myindirect/mystuff, the automount daemon needs only to create a symbolic link that points to the mystuff subdirectory because the /usr/staff/diane directory is already mounted. With the following map, the automount daemon must mount the file system twice:

```
mybin           host1:/usr/staff/diane/bin
mystuff         host1:/usr/staff/diane/stuff
```

C.3.3 Replicated File Systems

You can specify multiple locations for a single mount. If a file system is located on several servers and one of the servers is disabled, the file system can be mounted from one of the other servers. This makes sense only when mounting a read-only file system.

In the following example, the reference pages can be mounted from host1, machine2, or system3:

```
/usr/man\
                −ro,soft        host1:/usr/man \
                                machine2:/usr/man \
                                system3:/usr/man
```

The preceding example can also be expressed as a list of servers, separated by commas and followed by a colon and the pathname, for example:

```
/usr/man  −ro,soft  host1,machine2,system3:/usr/man
```

This syntax is valid only if the pathname is the same on each server.

When you access the reference pages, the `automount` daemon issues a `ping` command to each of the specified servers. The server that first responds to the `ping` command is used for the mount.

D

NIS ypservers Update Scripts

This appendix provides the following scripts for updating the `ypservers` map:

- `addypserver` — Adds a slave server
- `rmypserver` — Removes a slave server

D.1 Add Slave Server Script

Use the following procedure to create the `addypserver` script on an NIS master server:

1. Create an `addypserver` file in the `/var/yp` directory and type the following lines exactly as shown:

```
#!/bin/sh
PATH="/usr/bin:/var/yp:$PATH"
if [ $# != 1 ]; then
        echo "usage: $0 server"; exit 1
fi
DOMAIN='/usr/sbin/rcmgr get NIS_DOMAIN'
cd /var/yp
echo "
Adding $1 to ypservers map for domain DOMAIN ..."
(/var/yp/makedbm -u $DOMAIN/ypservers;\
echo $1 $1) | /var/yp/makedbm - tmpmap
mv tmpmap.dir $DOMAIN/ypservers.dir
mv tmpmap.pag $DOMAIN/ypservers.pag
yppush ypservers
```

2. Set the permissions to 700, using the `chmod` command as follows:

```
# chmod 700 /var/yp/addypserver
```

To add `host1` to the `ypservers` map, enter the following command:

```
# /var/yp/addypserver host1
```

You still need to edit the NIS master server's master `hosts` file and add an entry for the slave server, if it is not already in the `hosts` file. Then, update and distribute the map by entering the `make` command. See Section 7.4.1 for more information.

D.2 Remove Slave Server Script

Use the following procedure to create the `rmypserver` script on an NIS master server:

1. Create a `rmypserver` file in the `/var/yp` directory and type the following lines exactly as shown:

   ```
   #!/bin/sh
   PATH="/usr/bin:/var/yp:$PATH"
   if [ $# != 1 ]; then
           echo "usage: $0 server"; exit 1
   fi
   DOMAIN=‘/usr/sbin/rcmgr get NIS_DOMAIN‘
   cd /var/yp
   echo "
   Removing $1 from ypservers map for domain DOMAIN ..."
   /var/yp/makedbm -u $DOMAIN/ypservers | grep -v $1 \
   | /var/yp/makedbm - tmpmap
   mv tmpmap.dir $DOMAIN/ypservers.dir
   mv tmpmap.pag $DOMAIN/ypservers.pag
   yppush ypservers
   ```

2. Set the permissions to 700, using the `chmod` command as follows:

   ```
   # chmod 700 /var/yp/rmypserver
   ```

To remove `host1` from the `ypservers` map, enter the following command:

```
# /var/yp/rmypserver host1
```

E

NFS Error Messages

You might see the following types of NFS error messages:

- Server error messages
- Client error messages

E.1 Server Error Messages

The following error messages are issued to the screen or console or sent to the `syslogd` daemon.

`authget: unknown authflavor` *n*
`authflavor`

Explanation: Each NFS request has an authentication type. This message is displayed if the type is not AUTH_UNIX.

User Action: Have the client application use the AUTH_UNIX authentication type.

`fh3tovp: bad length:` *n*

Explanation: A client sent a bad file handle to the server.

`NFS request from unprivileged port, source IP address =` *n*

Explanation: The server, performing NFS server port monitoring, received an NFS request from a nonprivileged port (greater than or equal to 1024) on a client. This might indicate a security problem.

`NFS server: fs(`*n,n*`)not mounted; client address =` *n.n.n.n*

Explanation: The client requested a file on a file system that is not mounted or does not exist on the server. This can occur if a file system is unmounted while clients are using it or if the client passed an invalid file handle.

User Action: Make sure that the appropriate file system is mounted on the NFS server. If the file system is mounted on the same device, have the client system retry the operation. If the file system is mounted on a different device, have the client system unmount and remount the remote file system.

`NFS server: stale file handle fs(`*`n,n`*`)file `*`file`*` gen `*`n`*`,`
`client address = `*`n.n.n.n`*` errno `*`n`*

Explanation: The client accessed a file that no longer exists. The file was deleted either by the server or by another client.

`NFS server: unexported fs(`*`n,n`*`)file `*`file`*`, client address = `*`n.n.n.n`*

Explanation: A client that previously had access to a file system can no longer access the file system, either because of changes in the `/etc/exports` file or in net group mapping.

User Action: Have the client system unmount the file system.

`rfs_dispatch botch`

Explanation: The duplicate request cache routine returned an illegal value.

`rfs_dispatch: bad rfs reply `*`n`*
`ret`

Explanation: A server routine did not return a value or returned an incorrect value.

`rfs_dispatch: dispatch error, no reply`
`rfs_dispatch: sendreply failed`

Explanation: Possible reasons for this message include the following:

- The server is out of memory and cannot process or reply to a request.
- The server cannot find a route to the source.
- Some other network-related problem.

`too many nfsds`

Explanation: More `nfsd` daemons registered with NFS than were started.

E.2 Client Error Messages

This appendix provides an explanation and suggested user actions for the following classes of client error messages:

- Remote mount error messages
- automount error messages
- Console error messages

Within each section, error messages are listed alphabetically.

E.2.1 Remote Mount Error Messages

The following error messages are displayed if you are mounting directories or file systems from remote systems:

`Don’t know how to mount` *xxx*

Explanation: There is no entry in the `/etc/fstab` file for the argument you specified on the mount command line.

User Action: Edit the `/etc/fstab` file and verify that the mount point or remote file system exists.

`/etc/fstab: No such file or directory`

Explanation: The `/etc/fstab` file does not exist. The `mount` command discovered this when it tried to look up the name specified on the command line.

User Action: Create an `/etc/fstab` file and include the appropriate entries.

`nfs_mount: Permission denied for` *yyy*

Explanation: Your host name is not in the export list for the file system or directory you want to mount from the server.

User Action:

1. Get a list of your host’s exported file systems and directories, using the `showmount –e` command. For example, enter the following command if your server’s host name is host2:

   ```
   # /usr/bin/showmount –e host2
   ```

2. If the file system or directory you want to mount remotely is not in the list, or if your host or network group name is not in the user list for the file system or directory, log in to the server and check the `/etc/exports` file for the correct file system entry.

3. If the file system or directory name appears in the `/etc/exports` file, but not in the output from `showmount`, the failure is in the `mountd` daemon. The `mountd` daemon could not parse that line in the file, could not find the file system or directory, or the file system or directory name was not a locally mounted file system.

   If the file system or directory name appears in the `/etc/exports` file and Network Information Service (NIS) is running, check the server’s `ypbind` daemon; it might have stopped. See `exports`(4) for further information.

`nfs_mount: cannot mount` *xxx* `on` *yyy*: `Mount device busy`

Explanation: The file system or directory you are trying to mount is already mounted.

`nfs_mount: cannot mount` *xxx* `on` *yyy*: `No such file or directory`

Explanation: The local directory does not exist.

User Action: Check the spelling; list the files in both directories by using the `ls` command.

`nfs_mount: cannot mount` *xxx* `on` *`file`*: `Not a directory`

Explanation: Either the remote or local path is not a directory.

User Action: Check the spelling; list both directories by using the `ls` command.

`nfs_mount: cannot mount` *xxx* `on` *yyy*: `Not owner`

Explanation: You must mount the remote file system or directory as superuser (root) on your system.

`nfs_mount: illegal file system name` *xxx*; `use host:pathname`

Explanation: You did not specify the name of the server when you issued the `mount` command.

User Action: For example, to mount the file system `/usr/src` from the server host2, enter the following command:

```
# mount host2:/usr/src /host2/usr/src
```

`nfs_mount: invalid directory name` *xxx*
`directory pathname must begin with '/'.`

Explanation: The mount point on the local (client) system must be an absolute path starting at the root directory (`/`).

`nfs_mount: RPC: Authentication error;`
`why=Client credential too weak`

Explanation: The server is allowing client superuser mounts only and you are not a superuser. See `mountd`(8) for further information.

`nfs_mount: RPC: Authentication error;`
`why=Server rejected credential`

Explanation: Possible reasons for this error message include the following:

- The server is running with Internet address checking turned on and it cannot resolve your Internet address. If your system has multiple network interfaces configured, the server must be able to resolve all IP addresses, either using the local `/etc/hosts` file or the distributed `hosts` file.
- The server is running with domain or subdomain checking turned on and your system is not in the same domain or subdomain as the server.

See `mountd`(8) for further information.

`nfs_mount:` *`xxx`* `server not responding: port mapper failure rpc timed out Giving up on` *`yyy`*

Explanation: The server you are trying to mount from is down, or its port mapper is inoperative.

User Action:

1. Log in remotely to the server. If you are able to log in, the network is working.
2. Execute the `rpcinfo` command from the server. For example, for a server named host2, you would enter the following command:

   ```
   # /usr/sbin/rpcinfo -p host2
   ```

3. If the port mapper is running properly on the server, the `rpcinfo` command lists the registered program numbers. If it does not, restart the port mapper on the server. You also need a port mapper running on the client host; if it is not running there, start it.
4. After you restart the port mapper, stop the NFS daemons by entering the following command:

   ```
   # /sbin/init.d/nfs stop
   ```

   If NIS is running, stop the `ypbind` daemon on the server. Use the `kill` command and specify the process ID (PID).
5. If you stopped the `ypbind` daemon, restart it by entering the following command:

   ```
   # /usr/sbin/ypbind
   ```

   Restart the NFS daemons on the server by entering the following command:

   ```
   # /sbin/init.d/nfs start
   ```

`nfs_mount:` *`xxx`* `server not responding: rpc prog not registered`

Explanation: The `mount` command got through to the port mapper, but the NFS `mountd` daemon was not registered.

User Action:

1. Log in to the server.
2. Check that the `/usr/sbin/mountd` file exists, using the `ls` command.
3. Run the `ps` command to see if the `mountd` daemon is running. If it is not running, restart it by entering the following command:

   ```
   # /usr/sbin/mountd
   ```

`Can't get net id for host`

Explanation: There is no entry in the `/etc/hosts` file for the NFS server specified in the `mount` command line. If NIS is running, there is no entry in the `hosts` NIS map for the host name specified. If BIND is running, there is no entry in the `hosts` database for the host name specified.

E.2.2 automount Error Messages

The following error messages are issued to the screen or console or sent to the `syslogd` daemon by the `automount` program:

`bad entry in map` *`mapname`*

Explanation: The map entry in *`mapname`* is malformed and the `automount` program cannot interpret it.

User Action: Recheck the entry; you might need to include escape characters.

`Can't mount` *`mountpoint`*`:` *`reason`*

Explanation: The `automount` program cannot mount itself at *`mountpoint`*. The reason should be self-explanatory.

`couldn't create directory:` *`reason`*

Explanation: The `automount` program could not create a directory. The reason should be self-explanatory.

`dir` *`mountpoint`* `must start with '/'`

Explanation: The *`mountpoint`* must have a full pathname.

User Action: Check both the spelling and pathname of the mount point.

`hierarchical mountpoint:` *`mountpoint`*

Explanation: The `automount` program will not allow itself to be mounted within an automounted directory.

User Action: Use another strategy to mount the directory.

`host` *`hostname`* `not responding`

Explanation: The `automount` program attempted to mount from *`hostname`* but received no response or failed. These errors could indicate a server or network problem.

`hostname:filesystem` `server not responding`

Explanation: The `automount` program attempted to mount from *`hostname`* but received no response or failed. These errors could indicate a server or network problem.

`hostname``: exports:` *`rpc_err`*

Explanation: The `automount` program encountered an error while attempting to get the list of exported file systems and directories that it is allowed to mount from *`hostname`*.

This error occurs when a user attempts to access a mount point that has the `–hosts` map associated with it. This error indicates a server or network problem.

`hostname:filesystem` `already mounted on` *`mountpoint`*

Explanation: The `automount` program is attempting to mount a file system on a mount point that has already been mounted with that file system.

`map` *`mapname`*`, key` *`key`*`: bad`

Explanation: The map entry in *`mapname`* is malformed and the `automount` program cannot interpret it.

User Action: Recheck the entry; you might need to include escape characters.

`mapname``: Not found`

Explanation: The `automount` program cannot locate the map it requires. This message is returned only when you specify the `–v` option.

mapname: *yp_err*

Explanation: The automount program encountered an error when looking up a NIS map entry.

Mount of *hostname:filesystem* on *mountpoint:* reason

Explanation: The automount program attempted to mount from *hostname* but received no response or failed. These errors could indicate a server or network problem.

mountpoint: Not a directory

Explanation: The *mountpoint* exists but is not a directory.

User Action: Check both the spelling and pathname of the mount point.

mountpoint– *pathname* from *hostname*: absolute symbolic link

Explanation: The automount program detected that *mountpoint* is an absolute symbolic link (begins with /). The content of the link is *pathname*. Because this might have undesired consequences on the client, the automount program will not mount on absolute symbolic links.

no mount maps specified

Explanation: The automount program cannot find any maps to serve, nor can it find any NIS maps. This message is returned only when you specify the –v option.

WARNING: *hostname:file system* already mounted on mountpoint

Explanation: The automount program is mounting itself on top of an existing mount point. This message is a warning only.

WARNING: *mountpoint* not empty!

Explanation: The *mountpoint* directory is not empty. This message is returned only when you specify the –v option. It is warning you that the previous contents of *mountpoint* will not be accessible while the mount is in effect.

The following error messages can occur when a file system is exported from multiple servers as specified in a multiple-server map entry. They indicate possible network problems that can occur when the automount daemon requests a response from the servers.

`Cannot create socket for broadcast rpc:` *`rpc_err`*

Explanation: No server in a multiple-server map entry is responding. This indicates that the replicated file system could not be reached on any of the specified servers.

`Cannot receive reply to many_cast:` *`rpc_err`*

Explanation: No server in a multiple-server map entry is responding. This indicates that the replicated file system could not be reached on any of the specified servers.

`Cannot send broadcast packet:` *`rpc_err`*

Explanation: No server in a multiple-server map entry is responding. This indicates that the replicated file system could not be reached on any of the specified servers.

`Many_cast select problem:` *`rpc_err`*

Explanation: No server in a multiple-server map entry is responding. This indicates that the replicated file system could not be reached on any of the specified servers.

`NFS server (pid` *`n@mountpoint`*`) not responding still trying`

Explanation: An NFS request to the automount daemon with PID *n* serving mount point has timed out. The `automount` daemon might be overloaded or not running.

User Action: If the condition persists, reboot the client. You can also do the following:

1. Exit all processes that are using automounted directories.
2. Kill the current `automount` process.
3. Restart the `automount` process from the command line.

`Remount` *`hostname:filesystem`* `on` *`mountpoint`* `server not responding`

Explanation: The `automount` program was attempting to remount *`filesystem`* because it discovered that a part of the automounted hierarchy at the *`mountpoint`* was busy. The remote file system's server, *`hostname`*, did not respond to the mount request. This error indicates a server problem.

`trymany: servers not responding:` *`reason`*

Explanation: No server in a multiple-server map entry is responding. This indicates that the replicated file system could not be reached on any of the specified servers.

E.2.3 Console Error Messages

The following error messages might be displayed on the NFS client system console and in the error logger. They note an NFS file access failure.

`NFS server` *`hostname`* `not responding, still trying`

Explanation: File operations in a hard-mounted file system have suspended because communication between the client and the server has stopped.

`NFS server` *`hostname`* `ok`

Explanation: File operations have resumed.

`NFS` *`file operation`* `failed for server` *`hostname`*: *`reason`*

Explanation: If the operation is in a soft-mounted file system and the server is inoperable, the reason for the failure is that the operation timed out.

`NFS write error, server` *`hostname`*, `remote file system full`

Explanation: A write operation failed because the remote file system is full.

`NFS write error` *`errno`*, `server` *`hostname`*, `fs(`*`n,n`*`)`, `file` *`file`*

Explanation: A write operation was refused by the server. The *`fs`* and *`file`* variables are parts of the file handle (fhandle). See `errno`(2) for a description of write errors.

F

uucp Messages

This appendix provides a description and suggested user actions for the following uucp messages:

- Status and log file messages
- tip error messages

F.1 Status and Log File Messages

The messages in this section might appear in uucp status or log files. Use the uulog or uustat command to see the status messages.

ASSERT ERROR

An ASSERT error occurred, indicating a condition that only a system manager can solve. ASSERT errors are stored in the /usr/spool/uucp/.Admin/errors file and have the following form:

ASSERT ERROR (*prog*) pid: *xxxx* (*date/time*) *error error-location*

The variables have the following meaning:

prog	Name of the program generating the error.
xxxx	Process ID (PID) of the program.
date/time	Data and time when the error occurred.
error	A message describing the error. The message might include arguments. If there is a value contained in parentheses following the message, this value is often the error number (errno).
error-location	Name and version of the source file and the line in the file where the error occurred.

Table F–1 lists the ASSERT error messages.

Table F–1: ASSERT Error Messages

Error Message	Explanation and User Action
BAD LINE *line* (*num*)	The /usr/lib/uucp/Devices file has a bad line: *line* is the bad line and *num* is the number of fields found in the line. Correct the entry in the file. See Devices(4) for information on the file entries.
BAD LOGIN_UID (−1) BAD UID (−1) CAN NOT FIND UID (*num*)	The user ID used by the process is not currently logged in and is not defined in the /etc/passwd file or the networks database, if using NIS. Check your user ID, using the id command, and change the entry in the /etc/passwd file or the networks database, if using NIS.
BAD SPEED (*num*)	An unsupported baud rate (*num*) was specified. Check the command arguments or uucp configuration files. Then run uucpsetup to change the baud rate.
CAN'T CHDIR *dir* (*num*)	A command to change to directory *dir* failed with errno *num*. The uucp program required read access to the directory. Check the permissions on the directory. If the directory does not exist, check the permissions on the spool directory.
CAN'T CLOSE file (*num*)	Could not close file with errno *num*.
CAN'T CREATE file (*num*)	Could not open file with errno *num*. The uucp program needs write access to the file or directory. Check the permissions on the file and directory.
CAN'T LINK file (*num*)	Could not link a source file to the work file *file* in the uucp spool directory with errno *num*. Check the spool directory permissions.
CAN'T LOCK LCK.SQ. *sys* (0)	Could not lock the /var/spool/locks/LCK.SQ. *sys* file for system *sys*.

Table F–1: ASSERT Error Messages (cont.)

Error Message	Explanation and User Action
	Check the time and permissions on the file. If it is old, delete the file.
CAN'T OPEN file (*num*)	Could not open file with errno *num*. The uucp program needs write access to the file or directory.
	Check the permissions on the file and directory.
CAN'T STAT file (*num*)	The uucico daemon could not obtain information about the file with errno *num*.
	Check the permissions on the file.
CAN'T UNLINK file (*num*)	Could not unlink the file with errno *num*.
	Check the permissions on the file.
CAN'T WRITE file (*num*)	Could not open the file with errno *num*. The uucp program needs write access to the file or directory.
	Check the permissions on the file and directory.
FILE EXISTS file (*num*)	The file already exists and an access() call on that file returned errno *num*. The file is a uucp work file that was not cleaned up by another uucp process.
No uucp server (0)	The uucp service is not defined in the /etc/services file.
	Edit the /etc/services file and add a uucp entry.
SYSLST OVERFLOW (*num*)	There are too many jobs queued for a single system. The number of jobs is *num*.
	Use the uustat –q command and examine the queue. If the jobs are not old, try the request again. If there are old jobs in the queue, use the uucleanup command to clean out the queue. See uucleanup(8) for more information.
TOO MANY LOCKS (*num*)	The system limit on the number of lock files was exceeded while creating lock file *num*.

Table F–1: ASSERT Error Messages (cont.)

Error Message	Explanation and User Action
	Retry the request after the the current activity is completed.
XMV ERROR file (*num*)	The uuxqt daemon could not move the execute file to the .Xqtdir directory in the uucp spool area and failed with errno *num*. Use the ls –l command and verify that the .Xqtdir directory is owned by uucp and has a 775 permission.

BAD LOGIN/MACHINE COMBINATION

Explanation: There are two possible reasons for this message:

- The VALIDATE option for the local system is set in the Permissions file on the remote system and the local system's user name does not match the LOGNAME entry for the system in the remote system's Permissions file.

  Either ask the remote system's administrator to modify the LOGNAME entry for that user name, or edit the Systems file and modify the entry for the remote system to use the expected user name.

- The local system's user name has no corresponding LOGNAME entry in the remote system's Permissions file.

User Action: Either ask the remote system administrator to add a LOGNAME entry for that user name, or edit the Systems file and modify the entry for the remote system to use a known user name.

BAD SEQUENCE CHECK

Explanation: The information in /usr/lib/uucp/SQFILE file on the local and remote system is inconsistent. Possible reasons include:

- A new SQFILE has been installed on either system, possibly because a new operating system release was installed. If so, synchronize the files.
- Another system is imitating either the local or remote system. This indicates a potential security problem.

CALLBACK REQUIRED

Explanation: The local system initiated a call and informed the remote system that it has work for that system. The remote system is configured to accept work only if it initiates a call to the local system. Work is queued until the remote system calls the local system.

CALLER SCRIPT FAILED

Explanation: An error occurred while processing the chat script, defined in the Systems file.

User Action Enter the uutry *remote_system* command and observe the prompts from the remote system. Compare the prompts to the chat script. If there is a difference, run the uucpsetup script and change the chat script.

CAN’T ACCESS DEVICE

Explanation: Possible reasons include:

- The physical device could not be opened. Check the permissions on the terminal (tty) line, using the ls −l command. If neither user uucp nor group uucp has write access to the line, change the mode to 666.
- The modem type is not defined in the /usr/lib/uucp/Dialers file. Verify that the modem type has an entry in the Dialers file. If not, run the uucpsetup script and make an entry for the modem type.

CANNOT OPEN SYSTEMS FILE FOR READ

Explanation: The uucp program cannot read the /usr/lib/uucp/systems file.

User Action: Change the mode to 650, and the owner and group to uucp.

CONN FAILED (*string*)

Explanation: The connection to the remote system failed; *string* describes the reason for the failure.

CONVERSATION FAILED

Explanation: The conversation with the remote system has abnormally ended. Possible reasons are a modem error or system crash. Partially completed jobs are requeued and processed later.

DEVICE LOCKED

Explanation: Another utility (tip, cu, uugetty, or uucico) is already using the device.

User Action: Retry the request; you will continue to receive this message until the other utility has finished using the device.

`DIAL FAILED`

Explanation: The modem dialing sequence failed or timed out.

User Action: Retry the command.

`LOGIN FAILED`

Explanation: The `uucico` daemon timed out while trying to log in to the remote system.

User Action: Use the `uutry` command with your request to determine why the login is failing.

If the error occurs while processing the chat script, run the `uucpsetup` script and modify the chat script to reflect the actual messages used by the remote system. For example, if the chat script stops while waiting for a login prompt, modify the chat script to send a carriage return and delay before getting a login prompt.

If the login to the remote system is successful and then an error occurs, the `uucico` daemon on the remote system failed to start or was slow in sending the `Shere` message to the local system.

`LOST LINE (LOGIN)`

Explanation: The connection was lost during the login process.

User Action: Retry the request.

`NO DEVICES AVAILABLE`

Explanation: There are no devices available on this system of the type or speed requested.

User Action: You can install additional devices on your system, if your system allows, or modify the request to use one of the available devices in the `/usr/lib/uucp/Devices` file.

`REMOTE DOES NOT KNOW ME`

Explanation: The local system does not have an entry in the remote system's `Systems` file.

User Action: Contact the remote system's administrator to have an entry for your system put in the `Systems` file.

REMOTE HAS A LCK FILE FOR ME

Explanation: The remote system is trying to contact the local system while the local system is trying to connect to the remote system. The uucp utilities do not allow simultaneous connections between systems.

User Action: You can either retry the request later, or wait and see if the queued request is performed when the remote system connects to your system.

REMOTE REJECT AFTER LOGIN

Explanation: After successfully logging in to the remote system, the local and remote systems could not start a conversation. The remote system also returns the message BAD LOGIN/MACHINE COMBINATION.

REMOTE REJECT, UNKNOWN MESSAGE

Explanation: The remote system rejected the connection to the local system, but did not return a recognizable error message.

User Action: Retry your operation.

STARTUP FAILED

Explanation: After successfully logging in to the remote system, the local and remote systems could not start a conversation. Either the systems could not agree on a protocol or they could not start the protocol.

User Action: Verify that both the local and remote systems specify the same protocol in the /usr/lib/uucp/Systems file.

SUCCESSFUL

Explanation: The conversation completed successfully.

SYSTEM NOT IN Systems FILE

Explanation: The remote system is not in the /usr/lib/uucp/Systems file.

User Action: Use the uuname command to view a list of known uucp systems.

TALKING

Explanation: The local system is having a conversation with the remote system.

`WRONG MACHINE NAME`

Explanation: The remote system name does not match the system name entry in the `/usr/lib/uucp/Systems` file.

User Action: Verify the system name and run `uucpsetup` to make the necessary changes.

`WRONG TIME TO CALL`

Explanation: The remote system cannot be called at this time. The job is queued for completion later.

User Action: If you want to change the time, run `uucpsetup`.

F.2 tip Error Messages

The following messages might be displayed when using the `tip` utility:

`all ports busy`

Explanation: All ports are in use.

User Action: Try your request again later.

`can't open log file '/var/log/aculog' for update`
`contact your administrator`

Explanation: The `/var/log/aculog` file does not exist.

User Action: Create the file with the mode 664, and owner and group `uucp`.

`/etc/phones: can't open phone numbers file`

Explanation: The `/etc/phones` file does not exist, or the `tip` utility cannot read the `/etc/phones` file.

`link down`

Explanation: The terminal line (tty) cannot be opened.

User Action: Check that the mode of the tty device is 666.

`missing phone number`

Explanation: The remote system's phone number is not in the `/etc/phones` file.

User Action: Edit the `/etc/phones` file and add the remote system's phone number.

`system_name: missing device spec`

Explanation: The terminal line (`dv` parameter) is not defined in the `/etc/remote` file.

User Action: Edit the `/etc/remote` file and add the parameter.

`tip: unknown host` *`sysname`*

Explanation: The remote host system is not in the `/etc/remote` file.

User Action:

- Create an entry for the system in the `/etc/remote` file. See `remote`(4) for more information.
- Invoke `tip` using the remote host system's phone number instead of its name.

`tip: can't open host description file`

Explanation: The `/etc/remote` file does not exist, or the `tip` utility cannot read the `/etc/remote` file.

`tip: unknown host tip`*`speed`*

Explanation: The `tip` utility is not configured to use the *`speed`* specified on the command line.

User Action: Verify whether the hardware supports the speed. If it can, create a `tipspeed` entry for the speed in the `/etc/remote` file, using other `tipspeed` entries as a model. You should also create corresponding `UNIX-speed` and `dialspeed` entries in the file. Specify the modem type and the serial port to which it is attached, using the `at` and `dv` fields in the `dialspeed` entry.

`Unknown ACU type`

Explanation: The modem is unsupported.

User Action: Check the `at` field for the host system entry in the `/etc/remote` file. If the entry is correct, create an entry for the modem in the `/etc/acucap` file. See `acucap`(4) for more information.

`xxx``: unknown parity value`

Explanation: The parity value (`pa` parameter) in the `/etc/remote` file is invalid.

User Action: Edit the `/etc/remote` file and enter a valid value. See `remote`(4) for more information.

G

sendmail Error Messages

This appendix provides an explanation and suggested user actions for the `sendmail` error messages. These messages can occur when sending mail to another user on the same host or when sending mail using TCP/IP. If other mailers are configured on your system (for example, DECnet), see the documentation that accompanies the mailer for additional messages.

The following `sendmail` messages are found either in a rejected message or in the `syslogd` message file:

`binmail: opening /usr/spool/mail/`*`filename`*` -: Permission denied`

Explanation: The `/bin/mail` program could not deliver the mail on the destination host. Possible reasons are as follows:

- The `/usr/spool/mail` directory permissions are incorrect. The correct permissions are 1777.
- The mailbox permissions are incorrect. The correct permissions are 600.
- The mailbox owner is incorrect.

`Cannot send message for 3 days`

Explanation: The message was not delivered during the period specified by the retry parameter in the `/var/adm/sendmail/sendmail.cf` file. It is being returned to the sender. Possible reasons are as follows:

- The destination host does not exist.
- The mail was addressed to a host outside of your company and no relay host has been configured in the `/var/adm/sendmail/sendmail.cf` file.
- The host has been off line or the network connection has been unreliable for three days.

User Action:

1. Verify all address information.
2. If the mail was addressed to a host outside of your company, you might not be able to send the mail directly. Check your `sendmail` configuration by entering the following command:

grep '^define(_GateINET' /var/adm/sendmail/*hostname*.m4

If the braces are empty (that is, do not contain a host name), reconfigure `sendmail` and specify a relay host. See Section 11.3 for more information on specifying a relay name.

3. Send the message again. The message is queued and sent automatically when the host is reachable.

`Connection refused`

Explanation: The `sendmail` daemon is not running on the destination host.

User Action: Check whether `sendmail` is running on the host by using the `ps` command as follows:

```
# ps –ax | grep send
```

If it is not, ask the system administrator to start `sendmail`.

`Connection timed out during user open`

Explanation: A problem occurred during the SMTP session between 2 hosts, causing a time out.

User Action: The message will be retried later.

`Host unknown`

Explanation: Possible reasons are as follows:

- An address record for the host was not found.
- The `/var/adm/sendmail/sendmail.cf` file does not define a relay host that can handle mail addresses outside of your company.

User Action:

1. If BIND is not configured on your host, verify that the host's address is defined. Check the `/etc/hosts` file if you are resolving addresses locally or issue the `ypmatch` *hostname* `hosts` command if you are using NIS. The hosts entry in the `svc.conf` file defines the services used. If the host is not defined, ask your system administrator to correct the problem.
2. Check for MX records for the host by using the `nslookup` command as follows:

   ```
   # nslookup –q=mx hostname
   ```

   If a record exists, go to step 3.
3. Check for address records by using the `nslookup` command. If the address is not found, have the BIND domain administrator

for the destination domain add an address (A) record for the host in the destination domain.

4. If the mail was addressed to a host outside of your company, you might not be able to send the mail directly. Check your `sendmail` configuration by entering the following command:

   ```
   # grep '^define(GateINET' /var/adm/sendmail/hostname.m4
   ```

   If the braces are empty (that is, do not contain a host name), reconfigure `sendmail` and specify a relay host. Send the message again. See Section 11.3 for more information on specifying a relay name.

`I refuse to talk to myself`

Explanation: The local host was asked to connect to itself and deliver a message.

User Action: Check your `sendmail` configuration by entering the following command:

```
# grep '^define(GateINET' /var/adm/sendmail/hostname.m4
```

If the braces on any line contain your host's name, there is a configuration error. Reconfigure `sendmail`. See Section 11.3 for more information.

`Remote protocol error`

Explanation: This message is generally found in the `mail.log` file generated by the `syslogd` daemon and indicates a problem in communicating with the remote host.

User Action: The message will be retried later.

`Service unavailable`

This is a secondary error message. Some other error has occurred that caused `sendmail` to interpret an address as an action.

User Action: Look for other error messages, for example `Host unknown`, and resolve them first. Resolving other errors should resolve this error as well.

`User unknown/Addressee unknown`

Explanation: The message reached the final destination, but the user address was not found in the local `aliases` file or the local password file at the final destination.

User Action: Check whether the user address is correct or whether the user has moved.

H

Host Resources MIB Implementation

The Tru64 UNIX Simple Network Management Protocol (SNMP) agent implements the Host Resources MIB as described in RFC 1514. Although the RFC describes conceptual objects for management of host systems, it describes them in very general terms.

This appendix describes the Tru64 UNIX Host MIB implementation, including each group or table defined in RFC 1514 (with sample data). The formatting of SNMP data is specific to the implementation of an application. Compaq currently does not ship an application that presents SNMP data in this manner with Tru64 UNIX.

H.1 Tru64 UNIX Implementation Summary

The basic Tru64 UNIX implementation of RFC 1514 is as follows:

- The RFC specifies that when a product registry does not exist, all MIB variables of type `ProductID` return an object identifier of 0.0.
- The values of the `hrDeviceIndex` and `hrFSIndex` parameters remain unique between system reboots.
- Write access is not implemented for any Host MIB object.

H.2 System Group

The system group object implementation notes are as follows:

- The `hrSystemInitialLoadDevice` parameter is not implemented.
- The `hrSystemInitialLoadParameters` parameter returns the name of the booted kernel.

The following are sample data:

```
{hrSystemUptime.0                , TimeTicks, 0d 23:00:20.00}
{hrSystemDate.0                  , OCTET STRING, 1995-11-28,15:31:52.01}
{hrSystemInitialLoadParameters.0 , OCTET STRING, vmunix}
{hrSystemNumUsers.0              , Gauge, 0}
{hrSystemProcesses.0             , Gauge, 20}
{hrSystemMaxProcesses.0          , INTEGER, 1024}
```

H.3 Storage Group

Tru64 UNIX represents three types of logical storage: swap space, kernel memory, and file systems. The storage group object implementation is as follows:

- One entry in the `hrStorageTable` group is the total kernel memory being used.
- One entry is the current total swap space. (The value of the `hrStorageAllocationFailures` parameter for this entry is always 0.)
- There are several entries that each describe a specific type of kernel memory (the kernel malloc table). There is an entry for each memory type listed in the `<sys/malloc.h>` header file that is implemented on that particular host. (The value of the `hrStorageDescr` parameter is derived from the `malloc.h` file).

  Note

  These entries do not represent actual fixed-size memory pools that could be exhausted. They do, however, indicate how system memory is being utilized amongst the various subsystems.

  The value of the `hrStorageSize` parameter for the kernel memory entries is always 0, since there is no actual limit.

- There is one entry in the `hrStorageTable` group for each locally mounted file system. As specified in RFC 1514, remotely mounted file systems are not represented in the `hrStorageTable` group.
- The value of the `hrStorageDescr` parameter for file system-related entries is the same as the the `hrFSMountedPoint` parameter for the same file system in the `hrFSTable` group.
- The values of the `hrStorageIndex` parameter for file system-related entries is returned in the `hrFSStorageIndex` variable for the same file system in the `hrFSTable` group.
- The value of the `hrStorageType` parameter for file system storage entries is always`hrStorageOther`.

See Section H.5 for information on the file system implementation.

The following are sample storage group data:

```
{hrStorageIndex.1              , INTEGER, 1}
{hrStorageType.1               , OBJECT IDENTIFIER, hrStorageRam}
{hrStorageDescr.1              , OCTET STRING, Total Kernel Memory}
```

```
{hrStorageAllocationUnits.1     , INTEGER, 1024}
{hrStorageSize.1                , INTEGER, 2088960}
{hrStorageUsed.1                , INTEGER, 261112}
{hrStorageAllocationFailures.1  , Counter, 0}
{hrStorageIndex.2               , INTEGER, 2}
{hrStorageType.2                , OBJECT IDENTIFIER, hrStorageVirtualMemory}
{hrStorageDescr.2               , OCTET STRING, Total Swap Space}
{hrStorageAllocationUnits.2     , INTEGER, 1024}
{hrStorageSize.2                , INTEGER, 200704}
{hrStorageUsed.2                , INTEGER, 11920}
{hrStorageAllocationFailures.2  , Counter, 0}
{hrStorageIndex.3               , INTEGER, 3}
{hrStorageType.3                , OBJECT IDENTIFIER, hrStorageRam}
{hrStorageDescr.3               , OCTET STRING, MBUF}
{hrStorageAllocationUnits.3     , INTEGER, 1}
{hrStorageSize.3                , INTEGER, 0}
{hrStorageUsed.3                , INTEGER, 4096}
{hrStorageAllocationFailures.3  , Counter, 0}
{hrStorageIndex.4               , INTEGER, 4}
{hrStorageType.4                , OBJECT IDENTIFIER, hrStorageRam}
{hrStorageDescr.4               , OCTET STRING, MCLUSTER}
{hrStorageAllocationUnits.4     , INTEGER, 1}
{hrStorageSize.4                , INTEGER, 0}
{hrStorageUsed.4                , INTEGER, 32768}
{hrStorageAllocationFailures.4  , Counter, 0}
   :
{hrStorageIndex.99              , INTEGER, 99}
{hrStorageType.99               , OBJECT IDENTIFIER, hrStorageOther}
{hrStorageDescr.99              , OCTET STRING, /}
{hrStorageAllocationUnits.99    , INTEGER, 1024}
{hrStorageSize.99               , INTEGER, 63167}
{hrStorageUsed.99               , INTEGER, 46098}
{hrStorageAllocationFailures.99 , Counter, 0}
{hrStorageIndex.100             , INTEGER, 100}
{hrStorageType.100              , OBJECT IDENTIFIER, hrStorageOther}
{hrStorageDescr.100             , OCTET STRING, /proc}
{hrStorageAllocationUnits.100   , INTEGER, 8192}
{hrStorageSize.100              , INTEGER, 0}
{hrStorageUsed.100              , INTEGER, 0}
{hrStorageAllocationFailures.100, Counter, 0}
{hrStorageIndex.101             , INTEGER, 101}
{hrStorageType.101              , OBJECT IDENTIFIER, hrStorageOther}
{hrStorageDescr.101             , OCTET STRING, /usr}
{hrStorageAllocationUnits.101   , INTEGER, 1024}
{hrStorageSize.101              , INTEGER, 866102}
{hrStorageUsed.101              , INTEGER, 596323}
{hrStorageAllocationFailures.101, Counter, 0}
```

H.4 Device Tables

The Tru64 UNIX implementation supports CPUs, network interfaces, and disks in the device-related tables; printers are not supported. The CPU support is as follows:

- Each CPU physically attached to the system is represented in both the `hrDevice` and `hrProcessor` tables. The value of the `hrDeviceIndex` parameter for these entries is the processor number plus 1.
- The value of the `hrDeviceErrors` parameter is always 0.

- The value of the `hrDeviceStatus` parameter is either `running` or `down`.
- The value of the `hrProcessorLoad` parameter is accurately determined for each processor running on the system. Processor idle time is any time spent in the IDLE or WAIT states. Busy time is time spent in any other state.

  A background task records CPU time every 30 seconds, retaining 2 snapshots. When an SNMP request is received, CPU times are fetched immediately and the load average is calculated as the difference between this current data and the least recent snapshot. In this manner the values returned for the `hrProcessorLoad` parameter are current load averages over a period of at least 30 seconds, but not more than 1 minute. The value of the `hrProcessorLoad` parameter is calculated as follows:

  (delta *busy*/(delta *busy*+ delta *idle*)) * 100

The disk support is as follows:

- Each `re`, `ra`, and `rz` type disk whose special file is present in the `/dev` directory is represented in the `hrDeviceTable` group, the `hrdiskStorageTable` group, and the `hrPartitionTable` group.
- The value of the `hrDeviceStatus` parameter is `running` if the disk is online, or `down` if the disk is offline.
- The value of the `hrDeviceErrors` parameter is the sum of hard and soft errors reported for the disk.
- The value of the `hrDiskStorageMedia` parameter is always `unknown`.
- Data cannot be retrieved currently for offline devices (for instance, an empty CDROM drive). In these cases, the `hrDiskStorage` entry is as follows:

  ```
  media = 'unknown'
  capacity = 0
  removable = 'false'
  access = 'readWrite'
  ```

  The value of the `hrPartitionFSIndex` parameter is either zero (0) or the value of the `hrFSIndex` parameter for the `hrFSTable` entry corresponding to the offline file system.

The network device support is as follows:

- Each network interface is represented in both the `hrDeviceTable` group and `hrNetworkTable` group.
- The value of the `hrDeviceStatus` parameter is `running` if the interface is running, `down` if the interface is not up, or `unknown`.

- The value of the `hrDeviceErrors` parameter is the sum of inbound and outbound packet errors on that interface.
- The value of the `hrNetworkIfIndex` parameter is the same as the MIB-II value of the `ifIndex` parameter for that interface.

The following are sample device table data:

```
{hrDeviceIndex.1                , INTEGER, 1}
{hrDeviceType.1                 , OBJECT IDENTIFIER, hrDeviceProcessor}
{hrDeviceDescr.1                , OCTET STRING, Digital 2100 Server Model A500MP}
{hrDeviceID.1                   , OBJECT IDENTIFIER, 0.0}
{hrDeviceStatus.1               , INTEGER, running}
{hrDeviceErrors.1               , Counter, 0}
{hrDeviceIndex.2                , INTEGER, 2}
{hrDeviceType.2                 , OBJECT IDENTIFIER, hrDeviceProcessor}
{hrDeviceDescr.2                , OCTET STRING, Digital 2100 Server Model A500MP}
{hrDeviceID.2                   , OBJECT IDENTIFIER, 0.0}
{hrDeviceStatus.2               , INTEGER, running}
{hrDeviceErrors.2               , Counter, 0}
{hrDeviceIndex.3                , INTEGER, 3}
{hrDeviceType.3                 , OBJECT IDENTIFIER, hrDeviceProcessor}
{hrDeviceDescr.3                , OCTET STRING, Digital 2100 Server Model A500MP}
{hrDeviceID.3                   , OBJECT IDENTIFIER, 0.0}
{hrDeviceStatus.3               , INTEGER, running}
{hrDeviceErrors.3               , Counter, 0}
{hrDeviceIndex.4                , INTEGER, 4}
{hrDeviceType.4                 , OBJECT IDENTIFIER, hrDeviceProcessor}
{hrDeviceDescr.4                , OCTET STRING, Digital 2100 Server Model A500MP}
{hrDeviceID.4                   , OBJECT IDENTIFIER, 0.0}
{hrDeviceStatus.4               , INTEGER, running}
{hrDeviceErrors.4               , Counter, 0}
{hrDeviceIndex.5                , INTEGER, 5}
{hrDeviceType.5                 , OBJECT IDENTIFIER, hrDeviceNetwork}
{hrDeviceDescr.5                , OCTET STRING, tu0 - DEC TULIP Ethernet Interface}
{hrDeviceID.5                   , OBJECT IDENTIFIER, 0.0}
{hrDeviceStatus.5               , INTEGER, running}
{hrDeviceErrors.5               , Counter, 9}
{hrDeviceIndex.6                , INTEGER, 6}
{hrDeviceType.6                 , OBJECT IDENTIFIER, hrDeviceNetwork}
{hrDeviceDescr.6                , OCTET STRING, tra0 -
 DEC DW300 Token Ring Interface}
{hrDeviceID.6                   , OBJECT IDENTIFIER, 0.0}
{hrDeviceStatus.6               , INTEGER, down}
{hrDeviceErrors.6               , Counter, 0}
{hrDeviceIndex.7                , INTEGER, 7}
{hrDeviceType.7                 , OBJECT IDENTIFIER, hrDeviceNetwork}
{hrDeviceDescr.7                , OCTET STRING, ln0 - DEC LANCE Ethernet Interface}
{hrDeviceID.7                   , OBJECT IDENTIFIER, 0.0}
{hrDeviceStatus.7               , INTEGER, running}
{hrDeviceErrors.7               , Counter, 40}
{hrDeviceIndex.8                , INTEGER, 8}
{hrDeviceType.8                 , OBJECT IDENTIFIER, hrDeviceNetwork}
{hrDeviceDescr.8                , OCTET STRING, sl0 - Serial Line Interface}
{hrDeviceID.8                   , OBJECT IDENTIFIER, 0.0}
{hrDeviceStatus.8               , INTEGER, down}
{hrDeviceErrors.8               , Counter, 0}
{hrDeviceIndex.9                , INTEGER, 9}
{hrDeviceType.9                 , OBJECT IDENTIFIER, hrDeviceNetwork}
{hrDeviceDescr.9                , OCTET STRING, lo0 - Local Loopback Interface.}
{hrDeviceID.9                   , OBJECT IDENTIFIER, 0.0}
{hrDeviceStatus.9               , INTEGER, unknown}
{hrDeviceErrors.9               , Counter, 0}
```

```
{hrDeviceIndex.10              , INTEGER, 10}
{hrDeviceType.10               , OBJECT IDENTIFIER, hrDeviceNetwork}
{hrDeviceDescr.10              , OCTET STRING, ppp0 - 2.2}
{hrDeviceID.10                 , OBJECT IDENTIFIER, 0.0}
{hrDeviceStatus.10             , INTEGER, down}
{hrDeviceErrors.10             , Counter, 0}
{hrDeviceIndex.11              , INTEGER, 11}
{hrDeviceType.11               , OBJECT IDENTIFIER, hrDeviceDiskStorage}
{hrDeviceDescr.11              , OCTET STRING, /dev/rz0 - SCSI RZ28}
{hrDeviceID.11                 , OBJECT IDENTIFIER, 0.0}
{hrDeviceStatus.11             , INTEGER, running}
{hrDeviceErrors.11             , Counter, 0}
{hrDeviceIndex.12              , INTEGER, 12}
{hrDeviceType.12               , OBJECT IDENTIFIER, hrDeviceDiskStorage}
{hrDeviceDescr.12              , OCTET STRING, /dev/rz1 - SCSI RZ28}
{hrDeviceID.12                 , OBJECT IDENTIFIER, 0.0}
{hrDeviceStatus.12             , INTEGER, running}
{hrDeviceErrors.12             , Counter, 0}
{hrDeviceIndex.13              , INTEGER, 13}
{hrDeviceType.13               , OBJECT IDENTIFIER, hrDeviceDiskStorage}
{hrDeviceDescr.13              , OCTET STRING, /dev/rz6 - SCSI RRD43}
{hrDeviceID.13                 , OBJECT IDENTIFIER, 0.0}
{hrDeviceStatus.13             , INTEGER, down}
{hrDeviceErrors.13             , Counter, 0}
{hrProcessorFrwID.1            , OBJECT IDENTIFIER, 0.0}
{hrProcessorLoad.1             , INTEGER, 4}
{hrProcessorFrwID.2            , OBJECT IDENTIFIER, 0.0}
{hrProcessorLoad.2             , INTEGER, 0}
{hrProcessorFrwID.3            , OBJECT IDENTIFIER, 0.0}
{hrProcessorLoad.3             , INTEGER, 10}
{hrProcessorFrwID.4            , OBJECT IDENTIFIER, 0.0}
{hrProcessorLoad.4             , INTEGER, 19}
{hrDiskStorageAccess.11        , INTEGER, readWrite}
{hrDiskStorageMedia.11         , INTEGER, unknown}
{hrDiskStorageRemoveble.11     , INTEGER, false}
{hrDiskStorageCapacity.11      , INTEGER, 2055240}
{hrDiskStorageAccess.12        , INTEGER, readWrite}
{hrDiskStorageMedia.12         , INTEGER, unknown}
{hrDiskStorageRemoveble.12     , INTEGER, false}
{hrDiskStorageCapacity.12      , INTEGER, 2055240}
{hrDiskStorageAccess.13        , INTEGER, readWrite}
{hrDiskStorageMedia.13         , INTEGER, unknown}
{hrDiskStorageRemoveble.13     , INTEGER, false}
{hrDiskStorageCapacity.13      , INTEGER, 0}
{hrPartitionIndex.11.1         , INTEGER, 1}
{hrPartitionLabel.11.1         , OCTET STRING, /dev/rz0a}
{hrPartitionID.11.1            , OCTET STRING, }
{hrPartitionSize.11.1          , INTEGER, 65536}
{hrPartitionFSIndex.11.1       , INTEGER, 1}
{hrPartitionIndex.11.2         , INTEGER, 2}
{hrPartitionLabel.11.2         , OCTET STRING, /dev/rz0b}
{hrPartitionID.11.2            , OCTET STRING, }
{hrPartitionSize.11.2          , INTEGER, 200704}
{hrPartitionFSIndex.11.2       , INTEGER, 0}
{hrPartitionIndex.11.3         , INTEGER, 3}
{hrPartitionLabel.11.3         , OCTET STRING, /dev/rz0c}
{hrPartitionID.11.3            , OCTET STRING, }
{hrPartitionSize.11.3          , INTEGER, 2055240}
{hrPartitionFSIndex.11.3       , INTEGER, 0}
{hrPartitionIndex.11.4         , INTEGER, 4}
{hrPartitionLabel.11.4         , OCTET STRING, /dev/rz0d}
{hrPartitionID.11.4            , OCTET STRING, }
{hrPartitionSize.11.4          , INTEGER, 595968}
{hrPartitionFSIndex.11.4       , INTEGER, 0}
```

```
{hrPartitionIndex.11.5          , INTEGER, 5}
{hrPartitionLabel.11.5          , OCTET STRING, /dev/rz0e}
{hrPartitionID.11.5             , OCTET STRING, }
{hrPartitionSize.11.5           , INTEGER, 595968}
{hrPartitionFSIndex.11.5        , INTEGER, 0}
{hrPartitionIndex.11.6          , INTEGER, 6}
{hrPartitionLabel.11.6          , OCTET STRING, /dev/rz0f}
{hrPartitionID.11.6             , OCTET STRING, }
{hrPartitionSize.11.6           , INTEGER, 597064}
{hrPartitionFSIndex.11.6        , INTEGER, 0}
{hrPartitionIndex.11.7          , INTEGER, 7}
{hrPartitionLabel.11.7          , OCTET STRING, /dev/rz0g}
{hrPartitionID.11.7             , OCTET STRING, }
{hrPartitionSize.11.7           , INTEGER, 893952}
{hrPartitionFSIndex.11.7        , INTEGER, 3}
{hrPartitionIndex.11.8          , INTEGER, 8}
{hrPartitionLabel.11.8          , OCTET STRING, /dev/rz0h}
{hrPartitionID.11.8             , OCTET STRING, }
{hrPartitionSize.11.8           , INTEGER, 895048}
{hrPartitionFSIndex.11.8        , INTEGER, 0}
{hrPartitionIndex.12.1          , INTEGER, 1}
{hrPartitionLabel.12.1          , OCTET STRING, /dev/rz1a}
{hrPartitionID.12.1             , OCTET STRING, }
{hrPartitionSize.12.1           , INTEGER, 65536}
{hrPartitionFSIndex.12.1        , INTEGER, 0}
{hrPartitionIndex.12.2          , INTEGER, 2}
{hrPartitionLabel.12.2          , OCTET STRING, /dev/rz1b}
{hrPartitionID.12.2             , OCTET STRING, }
{hrPartitionSize.12.2           , INTEGER, 200704}
{hrPartitionFSIndex.12.2        , INTEGER, 0}
{hrPartitionIndex.12.3          , INTEGER, 3}
{hrPartitionLabel.12.3          , OCTET STRING, /dev/rz1c}
{hrPartitionID.12.3             , OCTET STRING, }
{hrPartitionSize.12.3           , INTEGER, 2055240}
{hrPartitionFSIndex.12.3        , INTEGER, 0}
{hrPartitionIndex.12.4          , INTEGER, 4}
{hrPartitionLabel.12.4          , OCTET STRING, /dev/rz1d}
{hrPartitionID.12.4             , OCTET STRING, }
{hrPartitionSize.12.4           , INTEGER, 595968}
{hrPartitionFSIndex.12.4        , INTEGER, 0}
{hrPartitionIndex.12.5          , INTEGER, 5}
{hrPartitionLabel.12.5          , OCTET STRING, /dev/rz1e}
{hrPartitionID.12.5             , OCTET STRING, }
{hrPartitionSize.12.5           , INTEGER, 595968}
{hrPartitionFSIndex.12.5        , INTEGER, 0}
{hrPartitionIndex.12.6          , INTEGER, 6}
{hrPartitionLabel.12.6          , OCTET STRING, /dev/rz1f}
{hrPartitionID.12.6             , OCTET STRING, }
{hrPartitionSize.12.6           , INTEGER, 597064}
{hrPartitionFSIndex.12.6        , INTEGER, 0}
{hrPartitionIndex.12.7          , INTEGER, 7}
{hrPartitionLabel.12.7          , OCTET STRING, /dev/rz1g}
{hrPartitionID.12.7             , OCTET STRING, }
{hrPartitionSize.12.7           , INTEGER, 893952}
{hrPartitionFSIndex.12.7        , INTEGER, 0}
{hrPartitionIndex.12.8          , INTEGER, 8}
{hrPartitionLabel.12.8          , OCTET STRING, /dev/rz1h}
{hrPartitionID.12.8             , OCTET STRING, }
{hrPartitionSize.12.8           , INTEGER, 895048}
{hrPartitionFSIndex.12.8        , INTEGER, 0}
{hrNetworkIfIndex.5             , INTEGER, 1}
{hrNetworkIfIndex.6             , INTEGER, 2}
{hrNetworkIfIndex.7             , INTEGER, 3}
{hrNetworkIfIndex.8             , INTEGER, 4}
```

```
{hrNetworkIfIndex.9            , INTEGER, 5}
{hrNetworkIfIndex.10           , INTEGER, 6}
```

H.5 File System Table

The file system table implementation is as follows:

- Each currently mounted file system is represented in the `hrFSTable` group.
- The available values for the `hrFSType` parameter do not cover all possible file system types in Tru64 UNIX. Some types (for example, `/proc`) report a value of `hrFSOther` for the `hrFSType` object.
- The `hrFSRemoteMountPoint` parameter is returned as a zero-length octet string for local file systems, as specified in RFC 1514.
- The `hrFSStorageIndex` parameter returns a zero (0) for remote file systems, in accordance with RFC 1514. For local file systems, the `hrFSStorageIndex` parameter returns the value of the `hrStorageIndex` parameter for the `hrStorageEntry` entry corresponding to that file system.

  The RFC specifies this design, presumably so that all storage-related information is available in one table. However, in order to discover file system full conditions, an SNMP application needs to do the following:

  1. Locate an entry in the the `hrFSTable` group.
  2. Retrieve that entry's value of the `hrFSStorageIndex` parameter. For example, call it *i*.
  3. If *i* is not zero (0), retrieve the values of the `hrStorageUsed.`*i* and `hrStorageSize.`*i* parameters.

- The value of the `hrFSBootable` parameter is always returned as False.
- The values of the `hrFSLastFullBackupDate` and `hrFSLastPartialBackupDate` parameters are always returned as {January 1 year 0 time 0}, in the DateAndTime format, as specified in RFC 1514, when these values are unknown.

The following are sample file system table data:

```
{hrFSIndex.1                   , INTEGER, 1}
{hrFSMountPoint.1              , OCTET STRING, /}
{hrFSRemoteMountPoint.1        , OCTET STRING, }
{hrFSType.1                    , OBJECT IDENTIFIER, hrFSBerkeleyFFS}
{hrFSAccess.1                  , INTEGER, readWrite}
{hrFSBootable.1                , INTEGER, false}
{hrFSStorageIndex.1            , INTEGER, 99}
{hrFSLastFullBackupDate.1      , OCTET STRING, 0-1-1,0:0:0.0}
{hrFSLastPartialBackupDate.1   , OCTET STRING, 0-1-1,0:0:0.0}
{hrFSIndex.2                   , INTEGER, 2}
```

```
{hrFSMountPoint.2                , OCTET STRING, /proc}
{hrFSRemoteMountPoint.2          , OCTET STRING, }
{hrFSType.2                      , OBJECT IDENTIFIER, hrFSOther}
{hrFSAccess.2                    , INTEGER, readWrite}
{hrFSBootable.2                  , INTEGER, false}
{hrFSStorageIndex.2              , INTEGER, 100}
{hrFSLastFullBackupDate.2        , OCTET STRING, 0-1-1,0:0:0.0}
{hrFSLastPartialBackupDate.2     , OCTET STRING, 0-1-1,0:0:0.0}
{hrFSIndex.3                     , INTEGER, 3}
{hrFSMountPoint.3                , OCTET STRING, /usr}
{hrFSRemoteMountPoint.3          , OCTET STRING, }
{hrFSType.3                      , OBJECT IDENTIFIER, hrFSBerkeleyFFS}
{hrFSAccess.3                    , INTEGER, readWrite}
{hrFSBootable.3                  , INTEGER, false}
{hrFSStorageIndex.3              , INTEGER, 101}
{hrFSLastFullBackupDate.3        , OCTET STRING, 0-1-1,0:0:0.0}
{hrFSLastPartialBackupDate.3     , OCTET STRING, 0-1-1,0:0:0.0}
{hrFSIndex.4                     , INTEGER, 4}
{hrFSMountPoint.4                , OCTET STRING, /tools}
{hrFSRemoteMountPoint.4          , OCTET STRING, /tools@tools}
{hrFSType.4                      , OBJECT IDENTIFIER, hrFSNFS}
{hrFSAccess.4                    , INTEGER, readWrite}
{hrFSBootable.4                  , INTEGER, false}
{hrFSStorageIndex.4              , INTEGER, 0}
{hrFSLastFullBackupDate.4        , OCTET STRING, 0-1-1,0:0:0.0}
{hrFSLastPartialBackupDate.4     , OCTET STRING, 0-1-1,0:0:0.0}
```

H.6 The Running Software Tables

The running software table implementation is as follows:

- The `hrSWOSIndex` parameter is always returned as zero (0), the kernel idle process. There is no one process that represents the primary operating system running on this host for Tru64 UNIX.
- Each process is represented as an entry in both the `hrSWRunTable` group and the `hrSWRunPerfTable` group. The value of the `hrSWRunIndex` parameter (used to index both tables) is the pid of that process. This means there is an entry whose `hrSWRunIndex` parameter value is 0 (zero), which is not typical of SNMP tables.
- The `hrSWRunName` parameter is always returned as a zero-length octet string.
- The `hrSWRunType` parameter is always returned as `unknown`.
- The `hrSWRunStatus` parameter is returned as either `running` (processes that are capable of being run or are waiting for CPU), or `notrunnable` (stopped or waiting for non-CPU resources).
- The `hrSWRunPath` parameter and the `hrSWRunParameters` parameter return the command and parameters, respectively, that were used to start this process. This is similar, but not identical, to the output of the `ps` command.
- The `hrSWRunPerfCPU` parameter returns the sum of accumulated system and user time for all threads running in a process. This value is

equivalent to the value returned by the `ps` cputime specifier (adjusted to units of centiseconds).

- The `hrSWRunPerfMem` parameter returns the current resident set size of the process. This value is equivalent to the value returned by the `ps rssize` specifier, adjusted to units of 1024 bytes (a "Kbyte" as defined in RFC 1514).

The following are sample running software table data:

```
{hrSWRunIndex.0                , INTEGER, 0}
{hrSWRunName.0                 , OCTET STRING, }
{hrSWRunID.0                   , OBJECT IDENTIFIER, 0.0}
{hrSWRunPath.0                 , OCTET STRING, }
{hrSWRunParameters.0           , OCTET STRING, }
{hrSWRunType.0                 , INTEGER, unknown}
{hrSWRunStatus.0               , INTEGER, running}
{hrSWRunIndex.1                , INTEGER, 1}
{hrSWRunName.1                 , OCTET STRING, }
{hrSWRunID.1                   , OBJECT IDENTIFIER, 0.0}
{hrSWRunPath.1                 , OCTET STRING, /sbin/init}
{hrSWRunParameters.1           , OCTET STRING,  -a}
{hrSWRunType.1                 , INTEGER, unknown}
{hrSWRunStatus.1               , INTEGER, notRunnable}
{hrSWRunIndex.3                , INTEGER, 3}
{hrSWRunName.3                 , OCTET STRING, }
{hrSWRunID.3                   , OBJECT IDENTIFIER, 0.0}
{hrSWRunPath.3                 , OCTET STRING, /sbin/kloadsrv}
{hrSWRunParameters.3           , OCTET STRING, }
{hrSWRunType.3                 , INTEGER, unknown}
{hrSWRunStatus.3               , INTEGER, notRunnable}
{hrSWRunIndex.16               , INTEGER, 16}
{hrSWRunName.16                , OCTET STRING, }
{hrSWRunID.16                  , OBJECT IDENTIFIER, 0.0}
{hrSWRunPath.16                , OCTET STRING, /sbin/update}
{hrSWRunParameters.16          , OCTET STRING, }
{hrSWRunType.16                , INTEGER, unknown}
{hrSWRunStatus.16              , INTEGER, notRunnable}
   :
   :
{hrSWRunIndex.142              , INTEGER, 142}
{hrSWRunName.142               , OCTET STRING, }
{hrSWRunID.142                 , OBJECT IDENTIFIER, 0.0}
{hrSWRunPath.142               , OCTET STRING, /usr/sbin/routed}
{hrSWRunParameters.142         , OCTET STRING,  -q}
{hrSWRunType.142               , INTEGER, unknown}
{hrSWRunStatus.142             , INTEGER, notRunnable}
{hrSWRunIndex.228              , INTEGER, 228}
{hrSWRunName.228               , OCTET STRING, }
{hrSWRunID.228                 , OBJECT IDENTIFIER, 0.0}
{hrSWRunPath.228               , OCTET STRING, /usr/sbin/nfsiod}
{hrSWRunParameters.228         , OCTET STRING,  7}
{hrSWRunType.228               , INTEGER, unknown}
{hrSWRunStatus.228             , INTEGER, notRunnable}
{hrSWRunIndex.394              , INTEGER, 394}
{hrSWRunName.394               , OCTET STRING, }
{hrSWRunID.394                 , OBJECT IDENTIFIER, 0.0}
{hrSWRunPath.394               , OCTET STRING, /usr/dt/bin/dtlogin}
{hrSWRunParameters.394         , OCTET STRING,  -daemon}
{hrSWRunType.394               , INTEGER, unknown}
{hrSWRunStatus.394             , INTEGER, notRunnable}
{hrSWRunIndex.395              , INTEGER, 395}
```

```
{hrSWRunName.395               , OCTET STRING, }
{hrSWRunID.395                 , OBJECT IDENTIFIER, 0.0}
{hrSWRunPath.395               , OCTET STRING, /usr/sbin/getty}
{hrSWRunParameters.395         , OCTET STRING,  console console vt100}
{hrSWRunType.395               , INTEGER, unknown}
{hrSWRunStatus.395             , INTEGER, notRunnable}
{hrSWRunIndex.396              , INTEGER, 396}
{hrSWRunName.396               , OCTET STRING, }
{hrSWRunID.396                 , OBJECT IDENTIFIER, 0.0}
{hrSWRunPath.396               , OCTET STRING, /usr/bin/X11/X}
{hrSWRunParameters.396         , OCTET STRING,  :0 -auth /var/dt/A:0-aaamka}
{hrSWRunType.396               , INTEGER, unknown}
{hrSWRunStatus.396             , INTEGER, notRunnable}
{hrSWRunIndex.397              , INTEGER, 397}
{hrSWRunName.397               , OCTET STRING, }
{hrSWRunID.397                 , OBJECT IDENTIFIER, 0.0}
{hrSWRunPath.397               , OCTET STRING, dtlogin}
{hrSWRunParameters.397         , OCTET STRING,  <:0> -daemon}
{hrSWRunType.397               , INTEGER, unknown}
{hrSWRunStatus.397             , INTEGER, notRunnable}
   :
   :
{hrSWRunPerfCPU.0              , INTEGER, 9288}
{hrSWRunPerfMem.0              , INTEGER, 10024}
{hrSWRunPerfCPU.1              , INTEGER, 34}
{hrSWRunPerfMem.1              , INTEGER, 64}
{hrSWRunPerfCPU.3              , INTEGER, 17}
{hrSWRunPerfMem.3              , INTEGER, 2000}
{hrSWRunPerfCPU.16             , INTEGER, 4476}
{hrSWRunPerfMem.16             , INTEGER, 88}
   :
   :
{hrSWRunPerfCPU.142            , INTEGER, 891}
{hrSWRunPerfMem.142            , INTEGER, 112}
{hrSWRunPerfCPU.228            , INTEGER, 0}
{hrSWRunPerfMem.228            , INTEGER, 56}
{hrSWRunPerfCPU.394            , INTEGER, 51}
{hrSWRunPerfMem.394            , INTEGER, 264}
{hrSWRunPerfCPU.395            , INTEGER, 7}
{hrSWRunPerfMem.395            , INTEGER, 80}
{hrSWRunPerfCPU.396            , INTEGER, 4329}
{hrSWRunPerfMem.396            , INTEGER, 2648}
{hrSWRunPerfCPU.397            , INTEGER, 8}
{hrSWRunPerfMem.397            , INTEGER, 232}
   :
   :
```

I

DNS Server Worksheet

This appendix provides a worksheet for you to photocopy. Use this worksheet to record information as you solve problems with your BIND server.

WORKSHEET FOR BIND TESTS

Sheet ___ of ___

Current server: ______________________

Server type: ______________________

Current domain name: ______________________

Target domain name: ______________________

named.boot file

Domain name: ______________________

Database file name: ______________________

Serial number: ______________________

Server IP address	Reachable
	Yes ☐ No ☐
	Yes ☐ No ☐
	Yes ☐ No ☐
	Yes ☐ No ☐

Nameservers

Nameserver name	IP address	Administrative Control	Reachable
		Yes ☐ No ☐	Yes ☐ No ☐
		Yes ☐ No ☐	Yes ☐ No ☐
		Yes ☐ No ☐	Yes ☐ No ☐
		Yes ☐ No ☐	Yes ☐ No ☐

Forwarders

Forwarder IP address	Administrative Control	Reachable
	Yes ☐ No ☐	Yes ☐ No ☐
	Yes ☐ No ☐	Yes ☐ No ☐
	Yes ☐ No ☐	Yes ☐ No ☐
	Yes ☐ No ☐	Yes ☐ No ☐

Root nameservers

Cache file: Yes ☐ No ☐

Nameserver name	Server IP address	Server IP address	Reachable
			Yes ☐ No ☐
			Yes ☐ No ☐
			Yes ☐ No ☐
			Yes ☐ No ☐

J

Format of DNS Data File Entries

The DNS configuration file, by default called `/etc/namedb/named.boot`, specifies the names of the DNS data files. These data files consist of entries, also known as Resource Records (RR), that follow the formats described in this chapter.

J.1 Format of DNS Resource Records

Here is the general format of a DNS Resource Record:

name *ttl* *addr-class* *entry-type* *entry-specific-data*

The fields are defined as follows:

Field	Description
name	This is the name of the domain, for example `cities.dec.com`. The domain name must begin in the first column. For some data file entries the name field is left blank. In that case, the domain name is assumed to be the same as the previous entry. A free standing period (.) refers to the current domain. A free standing at sign (`@`) denotes the current origin, thus allowing you to specify more than one domain. Two free standing periods (`..`) represent the null domain name of the root.
ttl	This is the time-to-live field, and specifies how long, in seconds, the data will be stored in the database. If this field is left blank, the value defaults to the ttl specified in the SOA (start of authority) entry. The maximum time-to-live is 99999999 seconds, or 3 years.

Field	Description
addr-class	This field is the address class. There are three classes: IN — Internet addresses, TXT — naming service data, ANY — all other types of network addresses. The address class of all data file entries of a given entry-type in a particular zone must be the same. Therefore, only the first entry in a zone need specify the *addr-class* field.
entry-type	This field states the resource record type, for example SOA or A.
entry-specific-data	All fields after the entry-type field vary for each type of date file entry (resource record).

The case is preserved in name and data fields when loaded into the DNS server. Comparisons and lookups using the DNS are case insensitive.

The following characters have special meanings in DNS data file entries:

Character	Meaning
\x	A backslash (\) escapes the next nondigit (*x*) character so that the character's special meaning does not apply. For example, you could use a period (.) to place a period character in a label.
\nnn	A backslash denotes the octet corresponding to the decimal number represented by nnn. The resulting octet is assumed to be text and is not checked for special meaning.
()	Parentheses group data that cross a line. In effect, line terminations are not recognized within parentheses.
;	A semicolon starts a comment, causing the rest of the line to be ignored.
*	An asterisk signifies a wildcard.

Most DNS data file entries have the current domain appended to their names if they are not terminated by a period (.). This is useful for appending the current domain name to the data, such as system names, but could cause problems when you do not want this to happen. Consequently, if the name is not in the domain for which you are creating the data file, end the name with a period.

Data files (resource records) can have the following types of entries:

- $include

- $origin
- A — address
- CNAME — canonical name
- HINFO — host information
- MB — mail box
- MG — mail group
- MINFO — mailbox information
- MR — mail rename
- MX — mail exchanger
- NS — name server
- PTR — domain name pointer
- SOA — start of authority
- WKS — well known services

J.2 Description of Data File Entries

The following sections describe each data file entry and its format.

J.2.1 Include Entry

An include entry is similar to a header file in the C programming language. This feature is particularly useful for separating different types of data into multiple files. An include entry begins with `$include` in the first column, and is followed by the name of the file to be included. For example:

```
$include /etc/namedb/mailboxes
```

This entry requests the DNS to load the data file `/etc/namedb/mailboxes`.

The `$include` entry loads data files into the local zone and acts as a data file organizer. For example, you can use `$include` entries to separate mail from host information.

J.2.2 Origin Entry

An origin entry changes the origin in a data file. This feature is particularly useful for putting more than one domain in a data file. An origin entry begins with `$origin` in the first column, followed by a domain origin. For example:

```
$origin state.dec.com.
```

This entry includes the domain `state.dec.com` in the data file. As a result, the DNS can provide information about the `state.dec.com` domain in addition to the local domain, provided your server is authoritative for the zone.

The `$origin` and `$include` entries can work together. They can also save typing and help keep the files organized. For example, assume that the following entries are in the `hosts.rev` file:

```
$origin ll.128.in-addr.arpa.
$include cities.dec.com.rev
```

The period after `arpa` signifies the complete domain name. Assume that the `cities.dec.com.rev` file consists of entries similar to the following:

```
33.22 IN PTR chicago.cities.dec.com.
```

In this situation, the complete reverse name for the host chicago is translated to be as follows:

```
33.22.11.128. in-addr.arpa. IN PTR chicago.cities.dec.com.
```

J.2.3 Address Entry

The address (A) data file entry lists the address for a specific system. An A entry has the following format:

name ttl addr-class entry-type address

The fields in the A entry have the values described in Section J.1, with the exception of the *address* field. This field specifies the IP address for each system. There should be only one A entry for each address of a given system.

The following is an example of two A entries:

```
;name                  ttl    addr-class    entry-type    address
miaml.cities.dec.com.         IN            A             128.11.22.44
                              IN            A             128.11.22.33
```

In this example, note that in the first entry the *ttl* field is blank, thus using the default ttl specified in the SOA entry. In the second entry, the first and second fields are blank, thus using the default name specified in the previous entry and the default ttl specified in the SOA entry. In this example, the host miami.cities.dec.com has two IP addresses.

J.2.4 Canonical Name Entry

The canonical name (CNAME) entry specifies an alias for a canonical name. For example, if the canonical name, (also known as the full DNS name or the fully qualified name) is miami.cities.dec.com, a reasonable alias might be miami or mi.

An alias must be unique, and all other entries should be associated with the canonical name and not with the alias. Do not create an alias and then use it in other entries. A CNAME entry has the following format:

aliases   ttl   addr-class   entry-type   can-name

The fields in the CNAME entry have the values described in Section J.1, with the following exceptions:

Field	Description
alias	This field specifies the nickname (alias) of the canonical name of the host.
can-name	This is the canonical name of the host. If the canonical name is a part of the current domain, you need to specify only the host name, for example, miami. If the canonical name is for a host in another domain, you must specify the fully qualified DNS name, followed by a period (.). For example: ohio.state.dec.com.

The following example shows two CNAME entries. The first entry is for a CNAME in the current domain, cities.dec.com; the second entry is for a CNAME in another domain:

```
;aliases    ttl    addr-class    entry-type    can-name
to                 IN            CNAME         toledo
oh                 IN            CNAME         ohio.state.dec.com.
```

J.2.5 Host Information Entry

The host information (HINFO) data file entry is for host specific information. This entry lists the hardware and operating system that are running at the specified host system. Only a single space separates the name of the hardware from the operating system information. Thus, if you need to use spaces as part of a host or operating system name, you must place the name in quotation marks. In addition, there can be no more than one HINFO entry for each host on the domain. The following is the HINFO entry format:

host   ttl   addr-class   entry-type   hardware   opsys

The fields in the HINFO entry have the values described in Section J.1, with the following exceptions:

Field	Description
host	This field specifies the host name. If the host is in the current domain, you need to specify only the host, say chicago, for example. If the host is in a different domain, you must specify the full DNS name, for example, utah.state.dec.com.. Be sure to include the period (.) at the end of the host name. This indicates the fully qualified DNS name.
hardware	This field specifies the type of CPU, for example, an AlphaServer 8400.
opsys	This field specifies the type of operating system running on the specified host and should be Tru64 UNIX for the Tru64 UNIX operating system.

The following is an example of a HINFO entry:

```
;name               ttl   addr-class   entry-type   hardware            opsys
ohio.state.dec.com.       IN           HINFO        "AlphaServer 8400"   "Tru64 UNIX"
```

In this example, note that the second field specifying the ttl is blank, thus using the default ttl specified in the SOA entry.

J.2.6 Mailbox Entry

The mailbox (MB) entry lists the system where a user wants to receive mail. The following is the format of an MB entry:

login ttl addr-class entry-type system

The fields in the MB entry have the values described in Section J.1, with the following exceptions:

Field	Description
login	This field is the login name for a user. Login names must be unique for the domain.
system	This field specifies the name system where the user wants to receive mail.

The following is an example of an MB entry:

```
;login   ttl   addr-class   entry-type   system
fred           IN           MB           potsdam.cities.dec.com.
```

In this example, note that the second field is blank, thus using the ttl specified in the SOA entry. Consequently, the user fred will have mail delivered to the host named potsdam in the domain cities.dec.com.

J.2.7 Mail Group Entry

The mail group entry specifies the members of a mail group. The MG entry is usually used with a MINFO entry. The following is the format of an MG entry:

group ttl addr-class entry-type member

The fields in the MG entry have the values described in Section J.1, with the following exceptions:

Field	Description
group	This field specifies the name of the mail group, for example, users or marketing.
member	This field specifies the login name and the domain of the user to be included in the mail group.

The following is an example of a MINFO entry and several MG entries:

```
;group    ttl    addr-class    entry-type    requests        member
fun              IN            MINFO         BIND-REQUEST    fred@miami.cities.dec.com.
                 IN            MG                            john@miami.cities.dec.com.
                               MG                            amy@miami.cities.dec.com.
```

In this example, note that the second field for all three entries is blank, thus using the ttl specified in the SOA entry. In addition, Fred, John, and Amy will receive any mail sent to the mail group fun.

J.2.8 Mailbox Information Entry

The mailbox information (MINFO) entry creates a mail group for a mailing list. The MINFO entry is usually associated with a mail group (MG) entry, but can also be used with a mailbox (MB) entry. The following is the format of a MINFO entry:

mailbox ttl addr-class entry-type requests maintainer

The fields in the MINFO entry have the values described in Section J.1, with the following exceptions:

Field	Description
mailbox	This field specifies the name of the mailbox, and is usually BIND.
requests	This field specifies the name where users should send mail relating to the DNS or mail. For example, a user might want to send a mail message requesting that an alias be set up.
maintainer	This field contains the login name of the person who should receive mail error messages. This is particularly useful when an error in member's names should be reported to a person other than the sender.

The following is an example of a MINFO entry:

```
mailbox    ttl    addr-class    entry-type    requests       maintainer
BIND              IN            MINFO         BIND-REQUEST   fred@miami.cities.dec.com.
```

In this example, note that the second field is blank, thus using the ttl specified in the SOA entry.

J.2.9 Mail Rename Entry

The mail rename (MR) entry lists aliases for a specific user. The following is the format of an MR entry:

alias   ttl   addr-class   entry-type   login

The fields in the MR entry have the values described in Section J.1, with the following exceptions:

Field	Description
alias	This field lists the nicknames for the specified user. The alias must be unique to the domain.
login	This field is the login name for the user whose alias is being established. There should also be a corresponding MB entry for the specified login name. Login names must be unique for the domain.

The following is an example of an MR entry:

```
;alias    ttl    addr-class    entry-type    login
lady             IN            MR            diana
princess         IN            MR            diana
```

This example shows how to set up the aliases lady and princess for a user whose login name is diana. Note that the second field is left blank, thus using the ttl specified in the SOA entry.

J.2.10 Mail Exchanger Entry

The mail exchanger (MX) entry specifies a system in the local domain (called a gateway) that knows how to deliver mail to a system that may not be directly connected to the local network. Consequently, the MX entry is useful for systems outside your local network that want to send mail to a user on one of your network's hosts.

You can also use the MX entry to list some of the hosts in the `/etc/hosts` file so that they do not appear to other systems using the DNS service.

The following is the format of an MX entry:

system ttl addr-class entry-type pref-value gateway

The fields in the MX entry have the values described in Section J.1, with the following exceptions:

Field	Description
system	This field specifies the name of the system where mail is to be sent.
pref-value	This field specifies the order a mailer should follow when there is more than one way to deliver mail to a given system.
gateway	This field contains the name of the gateway system, that is, the system that can deliver mail to the destination system on another network.

The following is an example of two MX entries:

```
;system                ttl    addr-class    entry-type    pref-value    gateway
tampa.cities.dec.com          IN            MX            0             seismo.cs.au.
*.folks.dec.com               IN            MX            0             relay.cs.net.
```

In this example, all mail destined for the domain folks.dec.com, regardless of the host name, is sent by route of the relay.cs.net host. In addition, note that the second field in both entries is blank, thus using the ttl specified in the SOA entry. The second entry uses an asterisk, which is a wildcard.

J.2.11 Name Server Entry

The name server (NS) entry specifies that a system is a name server for the specified domain. The following is the format of the NS entry:

name ttl addr-class entry-type server

The fields in the NS entry have the values described in Section J.1, with the exception of the server field. This field specifies the name of the primary master server for the domain specified in the first field.

The following is an example of an NS entry:

```
;name    ttl    addr-class    entry-type    server
                IN            NS            utah.states.dec.com.
```

J.2.12 Domain Name Pointer Entry

The domain name pointer (PTR) entry allows special names to point to some other location in the domain. PTR names should be unique to the zone. These entries are located on a primary server in the file `/etc/namedb/hosts.rev`. The following is the format of a PTR entry:

rev-addr   ttl   addr-class   entry-type   hostname

The fields in the PTR entry have the values described in Section J.1, with the following exceptions:

Field	Description
rev-addr	This field specifies the reverse IP address of the host. For example, if the host's address is `128.11.22.33`, the reverse address is `33.22.11.128`.
hostname	This is the fully qualified DNS name of the host, for example, `miami.cities.dec.com`. Be sure to include the period (.) at the end of the host name if the host is not in the current domain.

The following is an example of two PTR entries:

```
;rev-addr                     ttl    addr-class     entry-type    hostname
33.22                                IN             PTR           chicago
66.55.44.121.in-addr.arpa.           IN             PTR           mail.peace.org.
```

In this example, the first entry is for a host whose IP host address is `22.33` in the current domain. The specified rev.addr (`33.22`) is meaningful assuming that a `$origin` entry exists. See Section J.2.2 for a description of the `$origin` entry. If there is not an `$origin` entry, then the entire IP address, in reverse, must be specified.

The second entry is for a host in different domain (mail.peace.org.). As a rule, you should not do this because you are putting data in your server's cache for which your server is not authoritative. PTR entries and other resource records should be for hosts in your domain only. The PTR entry sets up a reverse pointer for the host `mail.peace.org`.

J.2.13 Start of Authority Entry

The start of authority (SOA) entry designates the beginning of a zone. There should be no more than one SOA entry per zone. The following is the format of an SOA entry:

name ttl addr-class entry-type origin person serial# refresh retry expire min

The fields in the SOA entry have the values described in Section J.1, with the following exceptions:

Field	Description
origin	This field is the name of the host on which the data file resides. This is usually a master server.
person	This field defines the login name and mailing address of the person responsible for the DNS running on the local domain.
serial#	This field specifies the version number of the data file. The person editing the master files for the zone should increment the value in this field each time a change is made to the data within the file. The serial number being changed informs the secondary servers that there is new data to be obtained from the master server. The maximum number is $2^{32}–1$ after the decimal point. The serial number field allows the DNS to determine which of two copies of data files in a zone are more recent. Typically, the serial number field begins at one (1) and is incremented by one each time the original data file is modified. It is best to use whole integers.
refresh	This field specifies how often, in seconds, a secondary DNS server is to check with the master server to see if it needs to update its data files. If the data files are out of date (as indicated by a mismatch of serial number fields), they are updated with the contents of the master server's files. The minimum refresh period is 30 seconds. If the refresh field is blank, however, the data file is not dynamically updated.
retry	This field specifies how often, in seconds, a secondary DNS server will try to refresh its data files after a refresh failure has occurred while making the check. If a DNS server attempts to refresh the files and fails, it tries to refresh them again every so many seconds, as specified in the retry field.

Field	Description
expire	This field specifies the upper limit, in seconds, that a secondary DNS server can use the data files in its cache before the data expires for lack of being updated, or before the DNS server checks to see if its cache needs to be updated.
min	This field specifies the default time to live, in seconds, that a data entry can exist in the event that the ttl entry is left blank.

The following is an example of an SOA entry. The first line is a comment that shows the fields:

```
;name   ttl   addr-class   entry-type   origin                             person
@             IN           SOA          utah.states.dec.com. hes.utah.states.dec.com. (
                                            1       ; serial
                                            3600    ; refresh every hr.
                                            300     ; retry every 5 min.
                                            3600000 ; expire in 1000 hrs.
                                            86400 ) ; min. life is 24 hrs.
```

In this example note that the parentheses indicate to the DNS that this is a single entry. The ttl field is blank, indicating that the default time to live specified in the min field (86400 seconds) is being used.

The semicolons allow comments for readability. In the example, the serial field is 1, the refresh field is 3600 seconds (once per hour), the retry field is 300 seconds (once per 5 minutes), the expire field is 3,600,000 seconds (1000 hours), and the min field is 86400 seconds (24 hours).

J.2.14 Well Known Services Entry

The well known services (WKS) entry describes well known services supported by a particular protocol at a specified address. The services and port numbers are obtained from the list of services specified in the `/etc/services` file. The following is the format of a WKS entry:

name   ttl   addr-class   entry-type   address   protocol   services

The fields in the WKS entry have the values described in Section J.1, with the following exceptions:

Field	Description
address	This field specifies the IP address for each system. There should be only one WKS entry for each protocol at each address.
protocol	This field specifies the protocol to be used, for example TCP or UDP.

Here is an example of two WKS entries:

```
;name    ttl    addr-class    entry-type    address       protocol    services
                IN            WKS           128.32.0.4    UDP         who route
                IN            WKS           128.32.0.78   TCP         (echo talk
                                                              discard sunrpc sftp
                                                              uucp-path netstat host
                                                              systat daytime link
                                                              auth time ftp
                                                              nntp whois pop
                                                              finger smtp supdup
                                                              domain nameserver
                                                              chargen)
```

Note that the first and second fields of both entries in this example are blank, which indicates that they are using the domain name specified in a previous entry and the default ttl specified in the SOA entry. The services listed in the second entry are contained within parentheses and are thus interpreted as being one entry, even though they appear to be on several lines.

Index

A

access problems
communicating with remote host (PPP), 13–32
communicating with remote host (SLIP), 13–30
communicating with remote network (SLIP), 13–30
dialing up remote system, 13–25
files not exported, 13–19
mounting remote file system, 13–21
network commands do not complete, 13–9
NTP server not reachable, 13–27
permission denied, 13–6
programs cannot perform tasks, 13–22
acucap file, 4–26
Address Resolution Protocol
(*See ARP*)
alias database
administering and distributing alias information, 11–18
aliases, 11–18, 11–19
changing, 11–18
aliases file
distributing for mail, 11–5
entries, 8–14
anonymous users, 8–13
application programs
cannot access files, 13–22
arp command, 14–2
authentication
and NTP, 10–6
and PPP, 4–14
authoritative server, 15–1
auto.master map
modifying, 7–25
removing an NFS map, 7–26
automount command
and string substitutions, C–5
pattern matching, C–5
using the ampersand, C–5
using the asterisk, C–6
automount daemon
and NIS, 8–2
defined, 8–1
error messages, E–6
invoking, 8–21
maps, 8–2
mounting a remote file system, 8–19
starting, 8–8
automount maps, 8–2, C–1, C–7, C–8
administering locally, 8–2
administering with NIS, 8–2
and environment variables, C–6
and the /var/yp/Makefile file, 7–10
creating, C–1
direct, C–2
distributing with NIS, 7–10
examples, C–1
indirect, C–3
modifying the master map, 7–25, 7–26
replicated file systems, C–8
specifying multiple mounts, C–7
specifying shared mounts, C–8

B

Berkeley Internet Name Domain
(*See DNS*)
BIND

(*See DNS*)
BIND Configuration application, 6–7
bindconfig command, 6–7
binmail utility, 11–16
BOOTP
  and DHCP, 3–19
  configuring a client, 3–19
bound interactive service, 5–17
BSD devices
  LAT, 5–6

C

cable
  guidelines for use with modems, 4–24
  null modem, 4–3
calls
  initiating to remote hosts, 9–23
Challenge Authentication Protocol
  (*See CHAP*)
CHAP
  chap-secrets file, 4–14
  use with PPP, 4–14
chat script, 4–19
client
  and NTP, 10–2
  automount error messages, E–6
  configuring for DNS, 6–10
  configuring for mail, 11–13
  configuring for NFS, 8–8
  configuring for NIS, 7–16
  delivering mail to, 11–5
  mail, 11–1
  monitoring DHCP, 3–17
  mounting a remote file system, 8–16
  NFS error messages, E–2
  NFS management tasks, 8–16
  NIS, 7–1
  NIS management tasks, 7–29
command files (uucp), 9–20
configuration worksheet, A–1
connection timed out message, 13–6
connections
  configuring system for dial-in access, 4–24
  configuring system for dial-out access, 4–26
  LAT host-initiated, 5–13
  outgoing (LAT), 5–14
  terminating SLIP, 4–10
counters
  Ethernet, B–1
  FDDI, B–6
  token ring, B–21
cron daemon
  log file, 9–21
  running the uudemon.admin script, 9–16
  running the uudemon.cleanu script, 9–18
  running the uudemon.hour script, 9–23
  scheduling uucp jobs, 9–22

D

daemon
  looking up process ID, 8–10
  ypbind not running, 13–15
data files
  DNS, 6–12
  uucp, 9–19
databases
  distributed by NIS, 7–2
datagram
  displaying route through a network, 14–3
dedicated service, 5–17
dedicated tty device on a terminal, 5–18
devices
  LAT, 5–6
DHCP, 3–1
  and security, 3–3, 3–18
  configuration worksheet, A–2, A–4
  configuring a client, 3–15

configuring a server, 3–12
disabling address assignment, 3–19
identifying server during configuration, 2–4
information required for configuration, 3–3
joind daemon, 3–16
mapping hardware addresses, 3–18
monitoring clients, 3–17
planning for configuration, 3–2
specifying during network configuration, 2–4
starting clients, 3–17
starting servers, 3–16
troubleshooting, 3–20
xjoin, 3–12
dial-in connections, 4–24
PPP, 4–22
SLIP, 4–8
dial-out connections, 4–26
PPP, 4–18
SLIP, 4–9
Dialcodes file, 9–8
direct maps
multiple mounts, C–7
directories
uucp spooling, 9–17
directory
(*See file system*)
DNS
client, 6–2, 6–10
configuration worksheet, A–6
configuring, 6–7
configuring a client, 6–10
configuring a server, 6–8
data file, J–1
determining the server type, 15–3
finding domain information, 15–6
glossary of terms, 15–1
information required for configuration, 6–4
make command, 6–13
master file data types, 15–1
MX data file entry, 6–14
named.boot, J–1
nslookup command, 15–18
resolving target data, 15–18
resource records, 6–12, J–1
sample configuration, 6–1
server testing worksheet, I–1
servers, 6–1
starting server testing, 15–2
testing forwarders, 15–8
testing primary servers, 15–13
testing secondary servers, 15–9
testing servers, 15–1
tracing from the root name server, 15–16
troubleshooting, 13–7, 13–9
updating server data files, 6–12
using domain-addresses for mail, 11–4
using mail exchanger entries, 11–4
domain
adding an NIS map, 7–25
adding users to NIS, 7–22
finding DNS information, 15–6
removing NIS map from, 7–26
domain name
NIS, 7–3
dropped packets, 2–26
dtmail utility, 11–16
Dynamic Host Configuration Protocol
(*See DHCP*)

E

environment variables, C–6
error log file
viewing, 14–8
error messages, 13–3
console, 13–34
errors
(*See network problems*)

/etc files
(*See files*)
Ethernet
configuration worksheet, A–2
configuring, 2–12
counters, B–1
information required for configuration, 2–2
monitoring, B–1
execute files (uucp), 9–19
exporting file systems, 8–9
exports file
NFS access and, 8–12
options, 8–12
sample entries, 8–10
security and, 8–11

F

FDDI
configuration worksheet, A–2
configuring, 2–12
displaying information about, 2–22
displaying parameters, 2–22
information required for configuration, 2–2
interface characteristics, B–18
interface counters, B–6
interface status, B–9
modifying parameters, 2–22
monitoring, B–4
using netstat to monitor, 2–21
fddi_config command, 2–22
Fiber Distributed Data Interface
(*See FDDI*)
file handle
stale, 13–19
file system
(*See also remote file system*)
exporting, 8–9
halting export of, 8–12
file transfer
monitoring, 14–8
files
acucap, 4–26
aliases, 8–14, 11–5
chap-secrets, 4–14
command (uucp), 9–20
configuration worksheet, A–2
data (uucp), 9–19
execute (uucp), 9–19
exporting, 8–9
exports, 8–10, 8–11, 8–12
fstab, 8–17, 8–18
gated.conf, 2–8
gateways, 2–7
halting export of, 8–12
hosts, 2–16
hosts.equiv, 2–17
ifaccess.conf, 2–20
inet.local, 2–19
inittab, 5–9
latstartup.conf, 5–7
log (uucp), 9–19
Maxuuscheds, 9–22
monitoring the transfer queue, 9–15
mount fails, 13–21
networks, 2–17
not exported, 13–19
pap-secrets, 4–14
permission denied, 13–6
Poll, 9–24
protecting from remote superusers, 8–11
rc.config, 13–7, 13–9, 13–34, 13–40
remote, 4–27
removing from the uucp queue, 9–19
routes, 2–16
slhosts, 4–7
svc.conf, 6–8, 6–11, 7–18, 13–14
FileShare application, 8–9
firewall, 11–3
forwarder, 15–1
fstab file, 8–17
and mounting remote files, 8–17, 8–18

bg option and, 8–18, 8–19
entry, 8–17, 8–18

G

gated daemon, 2–8
configuring, 2–15
gated.conf file, 2–8
gateway
and PPP, 4–22
and SLIP, 4–9
gateways file, 2–7
getty error messages, 13–34

H

hardware
not configured (PPP), 13–32
not configured (SLIP), 13–29
not configured (UUCP), 13–24
required for UUCP, 9–2
host
adding to the mail environment, 11–19
creating lists for DHCP, 3–14
LAT connection problems, 13–38
obtaining IP information, 6–13
testing Internet access, 14–1
host name
and configuring network interface, 2–4
obtaining with DNS, 6–13
Host Resources MIB, H–1
hosts database
distributing, 8–1
hosts file
configuring, 2–16
defined, 2–9
hosts.equiv file
configuring, 2–17
defined, 2–10

I

ICMP messages, 14–5
ifaccess.conf file, 2–20
ifconfig command
configuring NetRAIN with, 2–19
enabling access filtering with, 2–20
inet.local file
NetRAIN, 2–19
initialization problems
system hangs during boot, 13–3
inittab file
customizing for LAT, 5–9
insufficient node resources message, 13–37
Internet
monitoring server ports, 8–14
selecting NTP servers, 10–4
Internet Protocol address
(*See IP address*)
Internet to Ethernet translation tables, 14–2
InterNIC
and IP address, 2–4
IP address
and InterNIC, 2–4
and network numbers, 2–6
configuring for DHCP, 3–13
obtaining, 2–4
obtaining using DNS, 6–13
IP MTU size, 2–26
IP router
configuring system as, 2–15
defined, 2–8
requirements, 2–15

J

jobs
cleaning up undelivered, 9–19
monitoring status, 9–15
scheduling UUCP, 9–22
joind daemon, 3–16

L

LAT
- adding devices, 5–6
- automatic startup and shutdown, 5–5, 5–7
- configuration worksheet, A–6
- configuring kernel for, 5–4
- configuring multiple network adapters, 5–9
- configuring with latsetup, 5–6
- connections, 5–2
- controlling access, 5–3
- creating a startup file, 5–7
- creating your own service, 5–17
- customizing the inittab file, 5–9
- dedicated tty device on a terminal, 5–18
- defined, 5–1
- defining LAT/Telnet service, 5–15
- defining port names, 5–12
- defining server names, 5–12
- devices, 5–6
- gateway service, 5–15
- host-initiated connection, 5–13
- information required for configuration, 5–5
- LAN service, 5–17
- latsetup command, 5–6
- line disciplines, 5–14
- load balancing, 5–4
- outgoing connections, 5–14
- password protection, 5–4
- planning for configuration, 5–4
- printer hardware characteristics, 5–11
- printer setup, 5–10
- program interface, 5–14
- sample configurations, 5–1
- server node, 5–1
- service node, 5–1
- service node groups, 5–3
- services, 5–17
- setup, 5–6, 5–17
- starting and stopping, 5–7
- terminal server port settings, 5–11
- testing printer settings, 5–11
- testing printer setup, 5–13
- troubleshooting, 13–33
- user-created LAN service, 5–17
- verifying DLB support, 5–4

LAT printer
- deleting the application port, 13–38

LAT/Telnet gateway, 5–15
- setup, 5–15
- startup, 5–16

latsetup command, 5–6
- fails, 13–34

latstartup.conf file, 5–7

layered products
- mounting on a client, 8–18, 8–19

line disciplines, 5–14

linking script
- and mounting layered product, 8–18, 8–19

load balancing
- LAT, 5–4

Local Area Transport
- (*See LAT*)

local host
- obtaining NTP status from, 10–8, 10–9

local reference clock
- defined, 10–4

log files
- UUCP, 9–19, 9–20, 9–21

M

mail
- (*See also sendmail*)
- adding a host, 11–19
- administering aliases, 11–18
- aliasing root, 8–13
- and DECnet, 11–5
- and firewalls, 11–3
- archiving the mail queue, 11–17
- binmail, 11–16

changing aliases database, 11–18
configuration worksheet, A–11
configuring a mail client, 11–13
configuring a mail server, 11–14
configuring a standalone system, 11–13
delivering to clients, 11–5
displaying statistics, 11–19
distributing the aliases file, 11–5
distributing the passwd file, 11–5
domain-based addresses, 11–4
dtmail utility, 11–16
error messages, G–1
gateway, 11–3
information required for configuration, 11–8
mailq command, 11–17
mailstats command, 11–19
mailx utility, 11–16
message handler (mh) utility, 11–16
monitoring the queue, 11–17
planning, 11–7
required protocols, 11–7
sample configurations, 11–1
sending to remote superusers, 8–13
sendmail utility, 11–16
statistics file, 11–19
system roles, 11–1
troubleshooting, 13–40
using DNS MX records, 11–4
utilities, 11–16
mail aliases
distributing, 11–18
Mail Configuration application, 11–12
mail host, 11–19
mail utility, 11–16
(*See also dtmail utility*)
mailconfig application, 11–12
mailq command, 11–17
mailstats command, 11–19
mailx utility, 11–16
make command
and DNS, 6–13
updating NIS map, 7–23
makedbm command
building a new NIS map, 7–19, 7–21
showing contents of NIS map, 7–19, 7–21
Makefile
editing for NIS, 7–24
modifying for NIS, 7–26
map
(*See automount maps*)
(*See NIS map*)
Maxuuscheds file, 9–22
message handler (mh), 11–16
messages
automount, E–6
console, 13–34, E–10
LAT, 13–34
mail, G–1
NFS client, E–2
NFS server, E–1
tip, F–8
UUCP, F–1
MIB
Host Resources, H–1
Microsoft Challenge Authentication Protocol
(*See MS-CHAP*)
modem
and acucap file, 4–26
cable guidelines, 4–24
commands for dial-in access, 4–25
commands for dial-out access, 4–9
guidelines, 4–15, 4–23
using with SLIP, 4–3
using with UUCP, 9–3, 9–12
mount command, 8–17, 8–19
options, 8–19
remote files, 8–17
mountd daemon
not registered, 13–18
not running, 13–19

options, 8–11
MS-CHAP
use with PPP, 4–14
multicast routing, 2–12
multiple mount, C–7
MX records, 11–4

N

nameserver record, 15–1
netconfig application, 2–11
NetRAIN
configuration worksheet, A–2
configuring an interface, 2–19
hardware restrictions, 2–18
monitoring the interface, 2–20
set members, 2–6
netstat command, 2–21, B–1
network commands fail, 13–9
Network Configuration application, 2–11
network data structures
displaying with the netstat command, 2–21
Network File System
(*See NFS*)
network groups
and NFS, 8–5
Network Information Center
(*See InterNIC*)
Network Information Service
(*See NIS*)
network interfaces
and LAT configuration, 5–9
configuring, 2–11, 2–12
controlling access, 2–20
displaying packet headers on, 14–6
information required for configuration, 2–2
managing, 2–18
monitoring, 2–20, 2–21, B–1
NetRAIN, 2–6, 2–18
token ring, 2–6
types of, 2–3
network mask
(*See subnet mask*)
network problems
connection timed out, 13–6
gathering information, 16–1
host is unreachable, 13–6t
network daemon not running, 13–4
network is unreachable, 13–4
network software not configured, 13–4t
remote host is unknown, 13–4
reporting, 16–1
solving, 13–3
system not powered on, 13–3
tools for solving, 14–1
Network Time Protocol
(*See NTP*)
networks file
configuring, 2–17
defined, 2–10
NFS
allowing client superuser access, 8–12
and BIND, 8–1
and NIS, 8–1
and superuser mail, 8–13
and the hosts database, 8–1
and UIDs on remotely mounted file systems, 8–6
automount, 8–1
client, 8–1
client error messages, E–2
client management tasks, 8–16
configuration worksheet, A–8
configuring clients, 8–8
configuring servers, 8–7
console error messages, E–10
error messages, E–1
improving file security, 8–11
information required for configuration, 8–3
monitoring server ports, 8–14
monitoring system load, 8–15
mount command, 8–17, 8–19

mountd daemon, 8–11
mounting a remote file system, 8–16
nfsd daemons, 8–4
nfsiod daemon, 8–5
nfsstat command, 8–15
sample exports file entries, 8–10
server, 8–1
server daemons, 8–4
server error messages, E–1
server management tasks, 8–9
showmount, 8–12
starting manually, 8–11
troubleshooting, 13–18, 13–21
unmounting a remote file system, 8–22
nfsconfig application, 8–6
nfsd daemon
not registered, 13–18
not running, 13–19
stopping, 8–12
nfsstat command, 8–15
NIC whois service
(*See whois service*)
NIS
adding a slave server, 7–19, D–1
adding users to a domain, 7–22
administering automount maps, 8–2
aliases database, 11–5
and automount, 8–2
and sendmail, 11–18
changing a password, 7–29
client, 7–1
commands hang, 13–16
configuration, 7–8
configuration worksheet, A–7
configuring a client, 7–16
configuring a master server, 7–9
configuring a slave server, 7–13
databases distributed by, 7–2
distributing automount maps, 7–10
information required for configuration, 7–3
managing a client, 7–29
managing a server, 7–19
modifying, 7–18
modifying svc.conf, 7–18
modifying the Makefile, 7–26
nissetup command, 7–11
obtaining map information, 7–29
removing, 7–18
removing a slave server, 7–21, D–2
sample configuration, 7–1
security, 7–28
server types, 7–1
server update scripts, D–1
troubleshooting, 13–10, 13–14
updating maps, 7–23
Yellow Pages, 7–1
NIS map
adding to a domain, 7–25
distributing, 7–24
modifying the automount master, 7–25, 7–26
obtaining information from, 7–29
removing from a domain, 7–26
updating, 7–23
nissetup command, 7–8, 7–11
nr interface, 2–18
NS record, 15–1
nslookup command, 6–13
obtaining host information, 6–13
obtaining IP information, 6–13
solving problems using, 15–18
NTP
and system security, 10–7
client, 10–2
configuration worksheet, A–10
configuring, 10–5
displaying ntpd status, 10–10
displaying status, 10–9
displaying xntpd status, 10–8
information required for configuration, 10–3
Internet time servers, 10–4
local reference clock, 10–4
ntpdc command, 10–10

ntpq command, 10–8
sample configurations, 10–2
server, 10–2
troubleshooting, 13–26
xntpdc command, 10–9
ntp command, 10–10
ntpd daemon
monitoring hosts, 10–10
ntpdate command, 10–10
ntpdc command, 10–10
ntpq command, 10–8
ntpsetup command, 10–5
null modem cable, 4–3

O

optional service, 5–17
not available, 13–36
outgoing connections, 5–14

P

packets
dropped by bridges, 2–26
rejected by remote hosts, 2–26
PAP
use with PPP, 4–14
pap-secrets file, 4–14
passwd file
and NIS, 7–22
distributing for mail, 11–5
editing master server's, 7–22
user ID and, 7–22
passwd map, 7–22
password
changing for root, 7–29
changing in NIS, 7–29
Password Authentication Protocol
(*See PAP*)
password protection
LAT, 5–4
pattern matching
substitutions, C–5
PC-NFS daemon, 8–7
permission denied message, 13–6
ping command, 13–30, 13–32, 14–1
Point-to-Point Protocol
(*See PPP*)
Poll file, 9–24
configuration, 9–15
POP, 11–16
port monitoring, 8–14
portmap daemon
not running, 13–11, 13–15, 13–18, 13–21
Post Office Protocol
(*See POP*)
PPP
(*See also SLIP*)
and gateways, 4–22
configuration worksheet, A–5
configuring, 4–18
connection guidelines, 4–23
establishing dial-in connection, 4–22
establishing dial-out connection, 4–18
guidelines for running pppd, 4–18
information required for configuration, 4–15
sample configurations, 4–11
security, 4–14
terminating a connection, 4–23
troubleshooting, 13–31
pppd daemon
guidelines for running, 4–18
options, 4–13, 4–17
printer
using with LAT, 5–10
printer setup
LAT, 5–10
testing, 5–13
problem solving
diagnostic map, 13–1
tools, 14–1
process ID
looking up, 8–10
processes
limiting number of (UUCP), 9–21

protocols
  required for mail, 11–7

Q

queue
  checking UUCP, 9–16
  mail, 11–17
  transfer, 9–15

R

rc.config file
  DNS/BIND not configured,
    13–7, 13–9
  LAT not configured, 13–34
  sendmail not configured, 13–40
rejected packets, 2–26
remote command execution
  UUCP, 9–1
remote file, 4–24, 4–27
remote file system
  (*See also file system*)
  intr option, 8–19
  mounting automatically, 8–17,
    8–18
  mounting manually, 8–17
  mounting using automount, 8–19
  mounting with hard option, 8–19
  mounting with soft option, 8–19
  unmounting, 8–22
remote host
  cannot connect to using LAT,
    13–35
  cannot dial up, 13–25
  connection timed out, 13–6
  initiating calls to, 9–23
  insufficient LAT resources, 13–37
  network is unreachable, 13–4
  not reachable, 13–6t
  obtaining NTP status from, 10–9
  polling, 9–24
  unknown, 13–4
remote mount error messages, E–3
replicated file system, C–8
replicated file systems, C–9
resource record
  data file, 6–12
root
  remote superuser, 8–13
root name server
  using to trace DNS information,
    15–16
root password
  changing, 7–29
routed daemon, 2–6
  configuring, 2–14
  defined, 2–6
routes
  static, 2–16
routes file
  configuring, 2–16
  defined, 2–9
routing, 2–9
  dynamic, 2–6, 2–8
  multicast, 2–12
  source, 2–24
  static, 2–9, 2–16
rwhod daemon, 2–6
  configuring, 2–13
  defined, 2–6

S

script
  addypserver, D–1
  rmypserver, D–2
  uudemon.admin, 9–16
  uudemon.cleanu, 9–18, 9–20
  uudemon.hour, 9–22, 9–23
  uudemon.poll, 9–24
security
  and DHCP, 3–3, 3–18
  and NTP, 10–6
  and PPP, 4–14
  and xntpd, 10–7
  controlling access to interfaces,
    2–20
  exports file and, 8–11

firewall, 11–3
NIS and, 7–28
preventing access to files, 8–12
viewing exported files, 8–12
sendmail
troubleshooting, 13–40
sendmail aliases file, 11–18
sendmail utility, 11–16
Serial Line Internet Protocol
(*See SLIP*)
server
(*See also slave server*)
and NTP, 10–2
authoritative, 15–1
configuring a NIS master, 7–9
configuring a NIS slave, 7–13
configuring for DHCP, 3–12
configuring for mail, 11–14
configuring for NFS, 8–7
DNS testing, 15–1
mail, 11–1
NFS daemons, 8–4
NFS error messages, E–1
NFS management tasks, 8–9
NIS, 7–1
NIS management tasks, 7–19
not found (NTP), 13–27
primary (DNS), 15–2
querying time, 10–10
secondary (DNS), 15–2
updating files on (DNS), 6–12
shared mounts, C–8
showmount command, 8–12
Simple Network Management Protocol
(*See SNMP*)
slave server
adding to a domain, 7–19
not receiving updates, 13–13
removing from a domain, 7–21
script for adding to domain, D–1
script for removing from domain, D–2
slhosts file, 4–7
SLIP
(*See also PPP*)
and gateways, 4–9
configuration worksheet, A–5
configuring dial-in system, 4–8
configuring dial-out system, 4–9
defined, 4–1
information required for configuration, 4–4
IP address, 4–5
modem guidelines, 4–3
planning for configuration, 4–2
sample configurations, 4–1
startslip command, 4–5
terminating a connection, 4–10
troubleshooting, 13–29
SNMP
configuring, 12–1
described, 12–1
Host Resources MIB, H–1
SOA record, 15–2
source routing, 2–24
(*See also token ring*)
spooling directories, 9–17
cleaning up manually, 9–17
scheduling work in, 9–22
UUCP, 9–17
srconfig command, 2–24
stale file handle , 13–19
start of authority record, 15–2
startppp command
use for dial-in connections, 4–22
startslip command, 4–5, 13–31
invoking subcommands from script file, 4–10
subcommands, 4–5
use for dial-in connections, 4–8
static routes, 2–9, 2–16
status
FDDI, B–9
string substitutions
automount command, C–5
subnet mask, 2–5
and network class, 2–6
defined, 2–5
subnetwork

and DHCP, 3–14
sulog file
and UUCP, 9–21
superuser
access privileges, 8–12
allowing NFS access, 8–11, 8–12
and mail, 8–13
log of command usage, 9–21
remote superuser, 8–13
svc.conf file, 6–8, 6–11
modifying with svcsetup command, 6–11, 7–18
NIS entries, 13–14
svcsetup command, 6–11, 7–18
SVR4 devices
LAT, 5–6
symbolic links
and layered products, 8–18, 8–19
syslogd daemon
viewing the message files, 14–9
system load
NFS and, 8–15
system log files
(*See log files*)
system security
(*See security*)

T

TCP server daemon, 8–4
tcpdump command, 14–6
terminal devices
cannot be opened, 13–34
time
querying, 10–10
time servers
Internet network, 10–4
tip command, 4–18, 4–25
does not complete, 13–26
error messages, F–8
token ring
adapter speed, 2–6
configuration worksheet, A–2
configuring, 2–12
counters, B–21
displaying IP MTU size, 2–26
host information, B–24
information required for configuration, 2–2
IP MTU size, 2–26
modifying IP MTU size, 2–26
monitoring, B–20
source routing, 2–24
using netstat to monitor, 2–21
traceroute command, 14–3
transfer queue
guidelines for checking, 9–16
monitoring automatically, 9–16
monitoring manually, 9–15
troubleshooting, 13–3
DHCP, 3–20
DNS client, 13–9
DNS server, 13–7
LAT, 13–33
mail, 13–40
NFS client, 13–21
NFS server, 13–18
NIS client, 13–14
NIS server, 13–10
NTP, 13–26
PPP, 13–31
SLIP, 13–29
UUCP, 13–24

U

UDP server daemon, 8–4
uerf command, 14–8
unbound interactive service, 5–17
UNIX-to-UNIX Copy Program
(*See UUCP*)
unknown host message, 13–4
user environment
setting up, 7–23
users
adding to a NIS domain, 7–22
uucleanup command, 9–17
and uudemon.cleanu script, 9–19
UUCP

cleaning spooling directories, 9–17
cleaning up log files, 9–19
cleaning up undelivered jobs, 9–19
configuration worksheet, A–8, A–10
configuring, 9–11
configuring hardwired connections, 9–3, 9–12
configuring incoming systems, 9–13
configuring modems, 9–3, 9–12
configuring outgoing systems, 9–13
configuring TCP/IP, 9–3, 9–12
error messages, F–1
hardwired connections, 9–3, 9–12
HoneyDanBer version, 9–1
incoming systems, 9–9
information required for configuration, 9–3
information required for connections, 9–3
initiating calls to remote hosts, 9–23
limiting remote executions, 9–21
log files, 9–20
modems, 9–3, 9–12
monitoring a file transfer, 14–8
monitoring the transfer queue, 9–15, 9–16
outgoing systems, 9–6
Poll file configuration, 9–15
polling remote hosts, 9–24
required hardware, 9–2
sample configurations, 9–1
scheduling jobs, 9–22
TCP/IP connections, 9–3, 9–12
testing a remote connection, 14–6
troubleshooting, 13–24
uucpsetup command, 9–11
uusched command, 9–22
uutry command, 14–6
viewing errors, 14–8
uucpsetup command, 9–11
uudemon.admin script, 9–16
monitoring uucp status, 9–16
running, 9–16
uudemon.cleanu script, 9–18
and log files, 9–20
and uucleanup command, 9–19
running, 9–18
uudemon.hour script, 9–22, 9–23
and uudemon.poll script, 9–24
running, 9–23
uudemon.poll script, 9–24
and uudemon.hour script, 9–24
uulog command, 9–20
uusched command, 9–22
uustat command, 9–15
uutry command, 14–6
uuxqt
limiting number of processes, 9–21

V

variables
environment, C–6

W

whois command, 6–14
whois service
using, 6–14

X

xjoin command, 3–12
xntpd daemon
(*See also NTP*)
and system security, 10–7
monitoring hosts, 10–8
not running, 13–26
xntpdc command, 10–9

Y

Yellow Pages
  (*See NIS*)
ypbind daemon
  not running, 13–15
ypcat command, 7–30
ypmatch command, 7–30
yppasswd command, 7–29
ypserv daemon
  not running, 13–11
ypservers map
  showing contents of, 7–19, 7–21
ypwhich command, 7–30
  displays inconsistent
    information, 13–15

How to Order Tru64 UNIX Documentation

You can order documentation for the Tru64 UNIX operating system and related products at the following Web site:

http://www.businesslink.digital.com

If you need help deciding which documentation best meets your needs, see the Tru64 UNIX *Documentation Overview* or call **800-344-4825** in the United States and Canada. In Puerto Rico, call **787-781-0505**. In other countries, contact your local Compaq subsidiary.

To place an internal order, go to the following Web site:

http://asmorder.nqo.dec.com

The following table provides the order numbers for the Tru64 UNIX operating system documentation kits. For additional information about ordering this and related documentation, see the *Documentation Overview* or contact Compaq.

Name	Order Number
Tru64 UNIX Documentation CD-ROM	QA-MT4AA-G8
Tru64 UNIX Documentation Kit	QA-MT4AA-GZ
End User Documentation Kit	QA-MT4AB-GZ
Startup Documentation Kit	QA-MT4AC-GZ
General User Documentation Kit	QA-MT4AD-GZ
System and Network Management Documentation Kit	QA-MT4AE-GZ
Developer's Documentation Kit	QA-MT5AA-GZ
General Programming Documentation Kit	QA-MT5AB-GZ
Windows Programming Documentation Kit	QA-MT5AC-GZ
Reference Pages Documentation Kit	QA-MT4AG-GZ
Device Driver Kit	QA-MT4AV-G8

Reader's Comments

Tru64 UNIX
Network Administration
AA-PS2SD-TE

Compaq welcomes your comments and suggestions on this manual. Your input will help us to write documentation that meets your needs. Please send your suggestions using one of the following methods:

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

If you are not using this form, please be sure you include the name of the document, the page number, and the product name and version.

Please rate this manual:

	Excellent	Good	Fair	Poor
Accuracy (software works as manual says)	☐	☐	☐	☐
Clarity (easy to understand)	☐	☐	☐	☐
Organization (structure of subject matter)	☐	☐	☐	☐
Figures (useful)	☐	☐	☐	☐
Examples (useful)	☐	☐	☐	☐
Index (ability to find topic)	☐	☐	☐	☐
Usability (ability to access information quickly)	☐	☐	☐	☐

Please list errors you have found in this manual:

Page Description

________ ____
________ ____
________ ____
________ ____

Additional comments or suggestions to improve this manual:

____
____
____
____

What version of the software described by this manual are you using? ____________

Name, title, department ____________________
Mailing address ____________________
Electronic mail ____________________
Telephone ____________________
Date ____________________

------- Do not cut or tear - fold here and tape -------

COMPAQ

NO POSTAGE
NECESSARY IF
MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL
FIRST CLASS MAIL PERMIT NO. 33 MAYNARD MA

POSTAGE WILL BE PAID BY ADDRESSEE

COMPAQ COMPUTER CORPORATION
UBPG PUBLICATIONS MANAGER
ZKO3 3/Y32
110 SPIT BROOK RD
NASHUA NH 03062 9987

------- Do not cut or tear - fold here and tape -------

Cut on this line