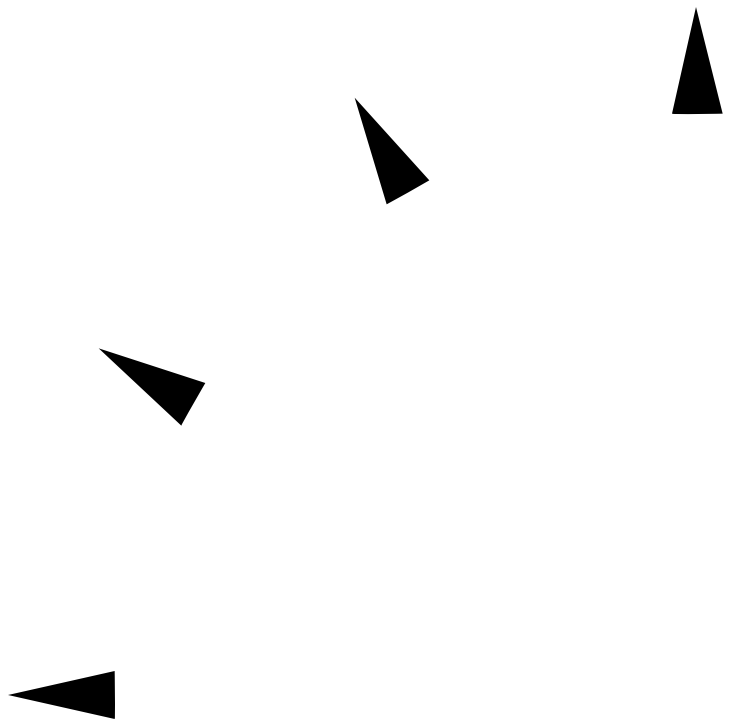


User Accounts, Printers, and Mail Administration



© 1994 Sun Microsystems, Inc.
2550 Garcia Avenue, Mountain View, California 94043-1100 U.S.A.

All rights reserved. This product and related documentation are protected by copyright and distributed under licenses restricting its use, copying, distribution, and decompilation. No part of this product or related documentation may be reproduced in any form by any means without prior written authorization of Sun and its licensors, if any.

Portions of this product may be derived from the UNIX[®] and Berkeley 4.3 BSD systems, licensed from UNIX System Laboratories, Inc., a wholly owned subsidiary of Novell, Inc., and the University of California, respectively. Third-party font software in this product is protected by copyright and licensed from Sun's font suppliers.

RESTRICTED RIGHTS LEGEND: Use, duplication, or disclosure by the United States Government is subject to the restrictions set forth in DFARS 252.227-7013 (c)(1)(ii) and FAR 52.227-19.

The product described in this manual may be protected by one or more U.S. patents, foreign patents, or pending applications.

TRADEMARKS

Sun, the Sun logo, Sun Microsystems, Sun Microsystems Computer Corporation, SunSoft, the SunSoft logo, Solaris, SunOS, OpenWindows, DeskSet, ONC, ONC+, and NFS are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and certain other countries. UNIX is a registered trademark of Novell, Inc., in the United States and other countries; X/Open Company, Ltd., is the exclusive licensor of such trademark. OPEN LOOK[®] is a registered trademark of Novell, Inc. Internet[™] is a trademark of Internet, Inc. PostScript and Display PostScript are trademarks of Adobe Systems, Inc. DEC[®], DECwriter[™], DEC LN03 and DEC LA100 are trademarks or registered trademarks of Digital Equipment Corporation. LaserWriter[®] is a registered trademark of Apple Computer Inc. Epson[®] is a registered trademark of Seiko Epson Corporation. NEC[®] is a registered trademark of Nippon Electric Corporation, Inc. All other product names mentioned herein are the trademarks of their respective owners.

All SPARC trademarks, including the SCD Compliant Logo, are trademarks or registered trademarks of SPARC International, Inc. SPARCstation, SPARCserver, SPARCengine, SPARCstorage, SPARCware, SPARCcenter, SPARCclassic, SPARCcluster, SPARCdesign, SPARC811, SPARCprinter, UltraSPARC, microSPARC, SPARCworks, and SPARCcompiler are licensed exclusively to Sun Microsystems, Inc. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc.

The OPEN LOOK and Sun[™] Graphical User Interfaces were developed by Sun Microsystems, Inc. for its users and licensees. Sun acknowledges the pioneering efforts of Xerox in researching and developing the concept of visual or graphical user interfaces for the computer industry. Sun holds a non-exclusive license from Xerox to the Xerox Graphical User Interface, which license also covers Sun's licensees who implement OPEN LOOK GUIs and otherwise comply with Sun's written license agreements.

X Window System is a product of the Massachusetts Institute of Technology.

THIS PUBLICATION IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT.

THIS PUBLICATION COULD INCLUDE TECHNICAL INACCURACIES OR TYPOGRAPHICAL ERRORS. CHANGES ARE PERIODICALLY ADDED TO THE INFORMATION HEREIN; THESE CHANGES WILL BE INCORPORATED IN NEW EDITIONS OF THE PUBLICATION. SUN MICROSYSTEMS, INC. MAY MAKE IMPROVEMENTS AND/OR CHANGES IN THE PRODUCT(S) AND/OR THE PROGRAM(S) DESCRIBED IN THIS PUBLICATION AT ANY TIME.



Contents

Preface.....	xvii
<i>Part 1—User Accounts</i>	
1. Setting Up User Accounts and Groups.....	1
About Setting Up User Accounts and Groups.....	2
Using Administration Tool.....	2
Policies for User Accounts and Groups	4
Set a Standard for User Names and User IDs	4
Passwords.....	6
Setting Up Groups.....	7
Home Directory	9
Initialization Files	11
Password, Shadow, and Group File Information Summary... ..	14
Fields in the Password File.....	14
Fields in the Shadow File	15
Fields in the Group File.....	16

Adding Users to a System and a Network	17
Instructions for Setting Up User Accounts and Groups	18
Setting Up Initialization (Skeleton) Files	19
▼ How to Set Up the System Initialization File	19
▼ How to Set Up the User Initialization File	20
Adding a Group	21
▼ How to Add a Group	22
Adding a User Account	23
▼ How to Add a User Account	23
Mounting a User's Home Directory	28
▼ How to Automount the Home Directory	29
▼ How to Manually Mount the Home Directory	29
Customizing a User's Environment	30
▼ How to Customize a User's Environment	30
User Account Reference Information	35
Collect and Record User Information	35
Shell Features	36
Environment Variables and Shell Variables	37
Default File Permissions (umask)	41
Message of the Day	41
2. Administering User Accounts and Groups	43
About Administering User Accounts and Groups	44
Using Administration Tool	44

Administering User Accounts	45
Creating and Modifying Passwords	47
Administering Groups	48
Instructions for Administering User Accounts and Groups. . .	49
Changing a User's Home Directory	49
▼ How to Create the New Home Directory.	50
▼ How to Move the Contents of the Home Directory . . .	52
Copying Initialization Files into a User's Home Directory	53
Modifying a User Account	54
▼ How to Modify a User Account.	55
Deleting a User Account.	57
▼ How to Delete a User Account	57
▼ How to Disable a User Account.	58
Modifying or Deleting Groups	58
▼ How to Modify a Group.	59
▼ How to Delete a Group.	60
▼ How to Create or Change a Password	60
<i>Part 2—Printing</i>	
3. Setting Up Printers	67
About Setting Up Printing Services.	68
Types of Print Clients and Print Servers.	70
Using Printer Manager	71
Network Prerequisites for Setting Up Printing	73
Local Printer Setup	74

Print Server Setup	74
Print Client Setup	74
Print Server Configurations	75
System Requirements for a Print Server.....	76
Printer Configuration Information	79
Printer Entries in the terminfo Database	84
Instructions for Setting Up Printing Using Printer Manager ..	86
Adding a Printer	87
▼ How to Add a Printer	87
Setting Up a Client.....	91
▼ How to Set Up a Client.....	91
Modifying or Deleting Printer Information.....	93
▼ How to Modify Printer Information.....	93
▼ How to Delete a Printer	94
Viewing Available Printers.....	94
▼ How to View Available Printers	95
Reference Information for Setting Up Printing Services.....	95
Frequently Used Printers	95
4. Routine Printer Administration.....	99
About Routine Printer Administration.....	100
Checking the Status of Printers and Print Requests	101
Managing Print Requests	101
Making Printers Available	103
Removing a Printer	104

Starting and Stopping the LP Print Service Scheduler	104
Cleaning Out Log Files	105
Instructions for Performing Routine Printer Administration. . .	105
▼ How to Check the Status of Printers and Print Requests	105
▼ How to Cancel Print Requests	111
▼ How to Accept or Reject Print Requests.	112
▼ How to Move Print Requests to Another Printer	113
▼ How to Change the Priority of a Print Request.	114
▼ How to Hold, Resume, or Print a Request Next	115
▼ How to Enable or Disable a Printer.	116
▼ How to Remove a Printer From Service	118
▼ How to Start or Stop the Print Service Scheduler	120
▼ How to Monitor and Clean Out Log Files	122
Reference Information for Routine Printer Administration . . .	123
Summary of LP Print Service Commands	124
5. Managing Character Sets, Filters, Forms, and Fonts	125
About Character Sets, Filters, Forms, and Fonts	126
Setting Up and Administering Character Sets.	126
Setting Up and Administering Print Filters	129
Setting Up and Administering Forms.	130
Installing and Maintaining PostScript Fonts	132
Managing Character Sets, Filters, Forms, and Fonts	135
▼ How to Set Up Hardware-Mounted or Selectable Character Sets	136

▼ How to Mount, Unmount, and Set Alerts for Hard Character Sets	138
▼ How to Add, Change, Remove, or Restore Print Filters	140
▼ How to View Print Filter Definitions	142
▼ How to Add, Change, or Remove a Form	143
▼ How to Mount, Unmount, and Specify Alerts for Forms	145
▼ How to Obtain Information About Forms	147
▼ How to Allow or Deny User Access to Forms	149
▼ How to Allow or Deny Use of Forms on a Printer	150
▼ How to Install Downloaded and Host-Resident PostScript Fonts	151
6. Setting Printing Policies	153
About Setting Printing Policies	154
Controlling the System Printer Destination	154
Setting Banner Page Policy	155
Permitting Users to Enable or Disable a Printer	155
Setting Printer Fault Alerts	156
Specifying Printer Fault Recovery	157
Defining Classes of Printers	158
Adding a Printer Description	159
Setting Priority Limits	159
Creating allow and deny Lists	160
Controlling User Access to Printers	161
Instructions for Setting Printing Policies	162
▼ How to Set the System Printer Destination	162

▼ How to Control Printing of Banner Pages	162
▼ How to Control Access to enable and disable Commands 163	
▼ How to Specify Alerts for Printer Faults	164
▼ How to Specify Fault Recovery	165
▼ How to Define a Class of Printers	166
▼ How to Add a Printer Description	166
▼ How to Set Default Priority and Priority Limits for Users 167	
▼ How to Allow or Deny Users Access to a Printer	168
7. Troubleshooting Printing Problems	171
About Troubleshooting Printing Problems	172
Troubleshooting No Output (Nothing Prints)	172
Troubleshooting Incorrect Output	174
Troubleshooting Hung LP Print Service Commands	176
Troubleshooting Idle (Hung) Printers	177
Troubleshooting Conflicting Status Messages	178
Instructions for Troubleshooting Printing Problems	178
▼ How to Troubleshoot No Output	179
▼ How to Troubleshoot Incorrect Output	199
▼ How to Unhang the LP Commands	204
▼ How to Troubleshoot an Idle (Hung) Printer	205
▼ How to Resolve Conflicting Status Messages	207

Part 3—Mail

8. Understanding Mail Services	211
---	------------

Components of the Mail Service	211
Mail Services Programs and Files	212
Mail Services Terminology	214
Planning Your Mail System	236
Local Mail Only	236
Local Mail and a uucp Connection	237
One Domain, Two Networks, and a Router	238
Two Domains and a Gateway	240
An Overview of the Mail Service	241
Structure of the Mail Service	241
How the Mail Service Works	244
How sendmail Works	244
Error Handling	246
How Mail Addressing Works	247
9. Setting Up and Administering	
Mail Services	251
Setting Up Mail Services	252
Setting Up Mail Security	254
▼ How to Set Up a Mail Server	254
▼ How to Set Up a Mail Client	254
▼ How to Set Up a Mail Host	255
▼ How to Set Up a Relay Host	256
Creating Mail Aliases	257
▼ How to List the Contents of an NIS+ Aliases Table	257

▼ How to Add Aliases to a NIS+ Aliases Table from the Command Line	258
▼ How to Add Entries by Editing an NIS+ Aliases Table	259
▼ How to Change Entries in an NIS+ Aliases Table	259
▼ How to Delete Entries from a NIS+ Aliases Table	260
▼ How to Set Up NIS Aliases Files	260
▼ How to Set Up Local Mail Aliases Files	261
▼ How to Set Up DNS Aliases Files	263
▼ How to Set Up the Postmaster Alias	264
Testing the Mail Configuration	265
▼ How to Test the Mail Configuration	265
Administering the Mail Configuration	266
Duties of the Postmaster	266
▼ How to Print the Queue	268
▼ How to Force the Queue	268
▼ How to Run the Old Mail Queue	269
System Log	270
Troubleshooting Tips	272
▼ How to Check Aliases	272
▼ How to Test the sendmail Program	272
▼ How to Verify Connections to Other Systems	273
Other Diagnostic Information	273
A. Setting Up Printer Services	
Using the Command-Line Interface	275

About Setting Up Printing Using the LP Print Service	
Command-Line Interface	276
Print Server Configuration	277
Printer Configuration Information for LP Print Service	
Commands	279
Setting Up Printing Services with the Command-Line Interface	280
▼ How to Set Up a SunOS 5.x Local Printer	281
▼ How to Set Up a SunOS 4.1 Local Printer	288
▼ How to Set Up a SunOS 5.x Print Server	288
▼ How to Set Up a SunOS 4.1 Print Server	292
▼ How to Set Up SunOS 5.x Print Clients	292
▼ How to Set Up SunOS 4.1 Print Clients	295
Adding a Network Printer	296
▼ How to Add a Network Printer (Using the BSD Protocol)	
296	
▼ How to Add a Network Printer (Using a Special Protocol)	
298	
B. Understanding and Customizing	
 the LP Print Service	301
An Overview of the LP Print Service	302
Functions Provided by the LP Print Service	302
The Structure of the LP Print Service	308
Understanding How the Print Service Works	319
Local Printing	319
Remote Printing	321
About Customizing the LP Print Service	327

Adjusting Printer Port Characteristics	327
Adjusting the terminfo Database for an Unsupported Printer 327	
Customizing the Printer Interface Program.....	328
Creating a New Print Filter	332
Creating a New Printer Form.....	342
Instructions for Customizing the LP Print Service	344
▼ How to Adjust the Printer Port Characteristics.....	344
▼ How to Adjust the terminfo Database for an Unsupported Printer	345
▼ How to Provide a Custom Printer Interface Program..	348
▼ How to Provide a New Print Filter	349
▼ How to Provide a New Form.....	353
C. Customizing sendmail Configuration Files.....	357
Features of the sendmail Program.....	358
Overview of sendmail Functions.....	359
Interfaces to the Outside World.....	360
How sendmail Works	360
Message Header Editing.....	362
Configuration File	362
How sendmail Is Implemented	362
Mail to Files and Programs.....	363
Message Collection	363
Message Delivery.....	364

Queued Messages	364
Configuration Overview	365
Arguments to sendmail	366
Queue Interval	366
Daemon Mode	367
Debugging	367
Trying a Different Configuration File	368
Tuning Configuration Parameters	368
Time Values	368
Delivery Mode	370
Load Limiting	370
Log Level	371
File Modes	372
sendmail Configuration File	373
Purpose of the sendmail Configuration File	373
A Sample sendmail Configuration File	375
sendmail Configuration File Syntax	384
Special Header Lines	390
R and S—Rewriting Rules	390
Address Rewriting Rules	392
Building a Configuration File	403
How sendmail Interacts With a Name Service	405
How to Set Up sendmail Requirements for Name Services	406
sendmail Command-Line Arguments	409

sendmail Configuration Options	411
Mailer Flags	414
D. Bibliography	417
Glossary	419
Index	427

Preface

User Accounts, Printers, and Mail Administration contains procedures for setting up and administering users and groups, user environments, printers, and electronic mail services. It also contains conceptual and planning information about the LP print service and electronic mail services.

This book assumes that you have already installed SunOS™ 5.x software, and have set up any networking software that you plan to use. The SunOS 5.x operating system is part of the Solaris™ 2.x product family, which also includes numerous bundled utilities and OpenWindows™ Version 3.x. The SunOS 5.x operating system is compliant with AT&T's System V, Release 4 operating system.

About Task Orientation

User Accounts, Printers, and Mail Administration is divided into three parts. Part I, "User Accounts," and Part II, "Printing," are written as task-oriented chapters. A task orientation lets you choose the best way to obtain the information you need. Each task-oriented chapter separates conceptual information from the task's instructions and contains these two main sections:

- *About section* – Provides background information related to the tasks.
- *Instructions section* – Provides step-by-step instructions that you need to perform each task, along with examples of the input and output associated with the task.

If necessary, a task-oriented chapter also includes this section:

- *Reference section* - Provides additional material that you may need to perform a task.

Depending on how much background information you may need to accomplish a task, you can either read the discussions in the About sections or go directly to the steps for the tasks in the Instructions sections. You also may be referred to additional information in the Reference sections (if any), which will help you perform the tasks.

Part III, “Mail,” is written in a more traditional, linear sequence. Descriptions of the underlying concepts of electronic mail services are described in two chapters. Understanding how these services work can help you troubleshoot problems. If you are familiar with how electronic mail services work, you can turn directly to the chapter that describes setup and administration procedures, along with related concepts.

Who Should Use This Book

This book is written for system administrators who have a basic working knowledge of the SunOS 5.x operating system, and who are familiar with windowing environments, including mouse- and menu-driven applications. It explains system administration concepts for administrators at all levels of expertise.

How This Book Is Organized

This book is organized into three parts and four appendixes.

Part 1, User Accounts

Chapter 1, “Setting Up User Accounts and Groups,” explains how to add, delete, and modify user accounts and how to assign passwords. It also explains how to create new groups and how to add, delete, and modify group accounts.

Chapter 2, “Administering User Accounts and Groups,” explains how to create home directories, set up the user’s default environment with initialization files, and set up a mail account (when mail service is already set up and running).

Part 2, Printing

Chapter 3, “Setting Up Printers,” describes the tasks for setting up basic printing services using the Printer Manager, an Administration Tool application.

Chapter 4, “Routine Printer Administration,” describes the tasks for administering printing services after printers are installed and set up.

Chapter 5, “Managing Character Sets, Filters, Forms, and Fonts,” describes additional printer-specific tasks, like setting up and administering print wheels, filters, forms, and fonts.

Chapter 6, “Setting Printing Policies,” describes the tasks for setting print policy, like whether banner pages print.

Chapter 7, “Troubleshooting Printing Problems,” describes how to troubleshoot problems that may occur when setting up or maintaining printing services.

Part 3, Mail

Chapter 8, “Understanding Mail Services,” describes what the mail service does, the programs it uses, and how it works.

Chapter 9, “Setting Up and Administering Mail Services,” describes several typical mail scenarios and details the tasks for setting up mail for each scenario. It also provides information about how to troubleshoot a mail installation and how to administer mail services.

Appendixes

Appendix A, “Setting Up Printer Services Using the Command-Line Interface,” describes the tasks for setting up basic printing services using the LP print service’s command-line interface.

Appendix B, “Understanding and Customizing the LP Print Service,” describes what the LP print service does, how it works, and the directory structure of the LP print service commands. It also explains how to adjust printer port characteristics, add entries to the `terminfo` database, write an interface program, and provide new print filters and forms.

Appendix C, “Customizing sendmail Configuration Files,” explains the structure and syntax of the `sendmail.main.cf` configuration file and how to customize it for more complex mail environments.

Appendix D, “Bibliography,” provides a list of published books on system administration that you can use to supplement the information in the system administration documentation set.

“Glossary,” lists words and phrases used in this book, and their definitions.

Related Books

If you need to install system software, or set up and configure systems on a network, refer to *SPARC: Installing Solaris Software* and *x86: Installing Solaris Software*.

If you are an experienced SunOS 4.1 system administrator, refer to the *Solaris 1.x to Solaris 2.x Transition Guide* for information about how to make the transition from administering SunOS 4.1 systems to administering SunOS 5.x systems.

For information about basic operating system commands and shells, see the *Solaris User’s Guide*.

For quick-reference information about system administration commands, see the *Solaris Advanced User’s Guide*.

User Accounts, Printers, and Mail Administration is part of the system administration documentation set. Cross-references in the text refer you to the appropriate book.

- *Peripherals Administration*
- *Name Services Administration Guide*
- *NFS Administration Guide*
- *Security, Performance, and Accounting Administration*
- *TCP/IP Network Administration Guide*

What Typographic Changes and Symbols Mean

The following table describes the typographic conventions in this book.

Table P-1 Typographic Conventions

Typeface or Symbol	Meaning	Example
AaBbCc123	The names of commands, files, and directories; on-screen computer output	Edit your <code>.login</code> file. Use <code>ls -a</code> to list all files. system% You have mail
AaBbCc123	What you type, contrasted with on-screen computer output	system% su password:
AaBbCc123	Command-line “placeholder”; replace with a real name or value	To delete a file, type <code>rm filename</code> .
AaBbCc123	Book titles, new words or terms, or words to be emphasized	Read Chapter 6 in <i>User’s Guide</i> . These are called <i>class</i> options. You <i>must</i> be root to do this.
%	UNIX C shell prompt	system%
\$	UNIX Bourne or Korn shell prompt	system\$
#	Superuser prompt, either shell	system#
SELECT	By default, the left mouse button	Click SELECT on the desired menu choice.
ADJUST	By default, the middle mouse button	Click ADJUST on each entry.
MENU	By default, the right mouse button	Press MENU on the desired menu button.

For steps, press Return only when instructed to do so, even if the text breaks at the end of the line, as shown in this example:

1. **Type** `pmadm -a -p tcp -s lpd -i root -m `nlsadmin -o /var/spool/lp/fifos/listenBSD -A `xuniversal_address` -v `nlsadmin -V`` **and press Return.**

For code examples, press Return at the end of a line. If the line ends with a backslash (\), do not press Return, because the example is continued on the line below.

```
# pmadm -a -p tcp -s lpd -i root -m `nlsadmin -o \  
/var/spool/lp/fifos/listenBSD -A \  
'\x000202038194180e0000000000000000'\` -v `nlsadmin -V`
```

When following steps or using examples, be sure to type double quotation marks ("), left single quotation marks (`), and right single quotation marks (') exactly as shown.

Part 1— User Accounts

Part 1 has two chapters:

Chapter 1, “Setting Up User Accounts and Groups,” describes how to set up user accounts and groups using Administration Tool’s User Account Manager application. You can add user accounts, in a distributed manner, to systems on the network, automatically creating home directories, and populating them with initialization files.

Chapter 2, “Administering User Accounts and Groups,” describes how to change home directories, create or change passwords for user accounts, and delete or modify group accounts.

Setting Up User Accounts and Groups

1 

This chapter describes how to set up user accounts and groups using User Account Manager, an Administration Tool application.

If you want to skip the background information that explains the concepts of setting up user accounts and groups, and *proceed directly to step-by-step instructions*, use the following table to find the page where the instructions for a specific task begin.

<i>How to Set Up the System Initialization File</i>	<i>page 19</i>
<i>How to Set Up the User Initialization File</i>	<i>page 20</i>
<i>How to Add a Group</i>	<i>page 22</i>
<i>How to Add a User Account</i>	<i>page 23</i>
<i>How to Automount the Home Directory</i>	<i>page 29</i>
<i>How to Manually Mount the Home Directory</i>	<i>page 29</i>
<i>How to Customize a User's Environment</i>	<i>page 30</i>

If you want to review background information first, read the “About Setting Up User Accounts and Groups” on page 2.

About Setting Up User Accounts and Groups

This chapter describes how to set up user accounts and groups in a network environment. The same procedures apply to standalone systems. The methods you use to set up and administer users and groups on a network depend on whether the network is administered through a name service.

Administration Tool enables you to set up user accounts on a local or remote system or in a name service environment. With a name service like NIS+, you can manage network information in a centralized location so that important system information, such as system and user names, do not have to be duplicated on every system in the network.

Using Administration Tool

Use the following Administration Tool applications to manage user account information:

- User Account Manager to set up and maintain user accounts
- Database Manager to set up and maintain UNIX[®] group information

Depending on the name service used on the network, the information defining user accounts and groups is stored in either:

- NIS+ tables
- NIS maps
- Local `/etc` files

To avoid confusion, all three types of information will be referred to as the `passwd` or `group file`, rather than the `passwd` or `group file, table, or map`.

Note – You can view the information in NIS maps with Administration Tool, but you cannot change the information. Refer to the SunOS 4.1 system documentation for information about how to administer NIS.

Before Using Administration Tool

The following hardware and software requirements must be met before using Administration Tool:

- SunOS 5.x software on all systems to be administered.
- A bit-mapped display monitor – Administration Tool’s applications can be used only on a system where the console is a bit-mapped screen.
- OpenWindows software. Start OpenWindows, if necessary.

```
$ /usr/openwin/bin/openwin
```

Required Access Privileges for Setting Up User Accounts

Table 1-1 describes the required access privileges for setting up user accounts.

Table 1-1 Required Access Privileges for Setting Up User Accounts

To Set Up User Accounts Using The	The Required Access Privileges Are:
/etc files	Root access or membership in the sysadmin group (GID=14) on the local or remote system.
NIS+ tables	Membership in the sysadmin group (GID=14) in the NIS+ group table. Create and destroy permissions on the NIS+ passwd, group, auto_home, mail_aliases, and cred tables. NIS+ permissions are granted by membership in an NIS+ group.

Traditional UNIX groups and NIS+ groups are separate. See *Name Services Administration Guide* for instructions on setting up NIS+ groups.

Using Administration Tool

Start Administration Tool from an OpenWindows window as follows.

```
$ admintool &
```

Policies for User Accounts and Groups

Your responsibilities for setting up users and groups at your site depend on the size of your company and the configuration of the systems and network. Here are some general suggestions for setting policy to help you administer users and groups:

- Set up a standard way of assigning user (login) names and user ID (UID) numbers for your site. See the next section for more information.
- Develop a method for collecting information. For example, you might design a form that users can fill out when requesting a new account. See “Collect and Record User Information” on page 35.
- If you need to create a new group ID (GID) for new users, create it before you assign users to the group. See “How to Add a Group” on page 22 for instructions.

Set a Standard for User Names and User IDs

User names (login names) let users access their local own systems and remote systems with the appropriate access privileges. Associated with each user name is a user ID number (UID). The UID identifies the user name to any system on which the user attempts to log in. UID numbers are also used by systems to identify the owners of files and directories. The reason for having both names and numbers is that users can remember names better than numbers, while computers process numbers faster than names.

User (Login) Names

You choose a user (login) name for each user account you create. User names must be unique on both local and remote systems. In addition, user names should be unique within your organization, which may span multiple domains.

A particular user name should always refer to the same person. The user name should be consistent across all accounts. If you create user accounts for a single individual on a number of different systems, always use the same user name and user ID. In that way, the user can easily interchange files between systems without ownership problems.

Each new user name must be distinct from any aliases known to the system or to an NIS or NIS+ domain. Otherwise, mail may be delivered to the alias rather than to the actual user.

Note – User Account Manager makes sure names are unique within the NIS+ domain.

It is helpful to establish a standard way of forming user names, and the names should be easy for users to remember. The name can contain from two to eight letters and numerals. The first character must be a letter, and at least one character must be a lowercase letter. The user names cannot contain underscores.

A simple scheme is to use the first name initial and first seven letters of the user's last name. For example, Ziggy Ignatz becomes `zignatz`. If that results in duplicate names, you can use the first initial, middle initial, and the first six characters of the user's last name. For example, Ziggy Top Ignatz becomes `ztignatz`. If that still results in duplicate names, you can use the first initial, middle initial, first five characters of the user's last name, and the number 1, or 2, or 3, and so on, until you have a unique name.

User ID Number

The user ID number (UID) identifies the account to the operating system by number. The number must be a whole number less than or equal to 60000. UIDs are required for both regular user accounts and special system accounts.

Customarily, the numbers for regular user accounts range from 100 through 60000 and the numbers from 0 through 99 are reserved for system accounts. UID 60001 is reserved for the `nobody` account and 60002 is reserved for the `noaccess` account.

Although UID numbers 0 through 99 are reserved, you can add a user with one of these numbers. However, it is not recommended that you use them for regular user accounts. By definition, `root` always has UID 0, `daemon` has UID 1, and pseudo-user `bin` has UID 2. In addition, you should give `uucp` logins and pseudo user logins, like `who`, `tty`, and `ttytype`, low UIDs so they fall at the beginning of the `passwd` file.

The UID numbers for regular user accounts should be unique across your organization. However, User Account Manager does not prevent duplicate UIDs within an NIS+. Sometimes you may want to use duplicate UIDs. For example, it's easier to clean up after a training class when all the students have the same UID, even though they log in with different user names. Also, you may have to use duplicates when the supply of unique UIDs is exhausted, which can occur in a university setting.

Be careful when you reuse a UID number. Even when a user has left the organization and the user account has been removed, the UID number remains assigned to files and directories created by that user. Confusion could result if you reassign a UID number and then restore old files created by the original user.

As with user (login) names, you should adopt a scheme to assign unique UIDs. Some companies assign unique employee numbers, and administrators add 1000 to the employee number to create a unique UID number for each employee.

Check for Duplicate User Names (Logins) and User IDs

When you use a name service, you should check new user names and UID numbers to be sure they are unique on the network. You can use User Account Manager and Database Manager to check for duplicate entries and display other user and group information in local `/etc` files, in NIS+ tables, or in NIS maps. When creating new user accounts with User Account Manager, you can't duplicate an existing user name.

Passwords

Encrypted passwords and password aging information are stored in the `/etc/shadow` file on the local system in the Solaris 2.x environment. The `shadow` file is in ASCII format and only the superuser has read privileges. See "Adding Users to a System and a Network" on page 17 for more information about the `shadow` file.

NIS+ password information is stored in the `passwd` table, which has more stringent access restrictions than the NIS software. See the *Name Services Administration Guide* for more information on NIS+ security features.

Setting Up Groups

A *user group* is a collection of users who can share files and other system resources. For example, the set of users working on the same project could be formed into a user group. A user group is traditionally known as a UNIX group. Information about user groups is stored in the `group` file.

Each group has a GID (group identification number, analogous to the user identification number), which identifies it internally to the system. A group should have a name and a list of user names. User groups can be defined in two ways:

- **Primary group membership** – is specified by an entry in the GID field in the `passwd` file. Whenever a new GID appears in the group field of the `passwd` file, a new group is defined.
- **Secondary group membership** – is specified by an entry in the `group` file by group name, GID, and user list. This is the preferred method.

All users belong to at least one group, their primary group, which is identified by the group field of their user account. When you use User Account Manager to add user accounts, you must specify the user's primary group; otherwise, the default primary group is `nobody`.

Users can belong to up to 16 secondary groups. To belong to a secondary group, the user must be added to the groups' member list, which User Account Manager does automatically.

The `groups` command shows the groups a user belongs to. A user can have only one primary group at a time. However, the user can temporarily change the primary group (with the `newgrp` command) to any other group in which he or she is a member.

Some applications, like the file system, look only at the user's primary group. For example, ownership of files accounting data reflect the primary group, not any secondary groups. Other applications may take into account a user's secondary memberships. For example, a user has to be a member of the `sysadmin` group to use the Administration Tool, but it doesn't matter if `sysadmin` is his or her current primary group.

User groups probably are best known as the groups referred to by the read-write-execute permissions for the user, *group*, and world on files and directories. These permissions are a cornerstone of security. You can't access others' files (that do not allow world access) unless your primary or secondary

group has permission to access the files. For example, a group called `techwrite` could be created for technical writers, and a central directory of document files could be set up with write permission for the `techwrite` group. That way, only writers would be able to change the files.

User groups can be local to a system or used across a network. Across the network, user groups allow a set of users on the network to access a set of files on a without making those files available to everyone.

UNIX User Groups

By default, all SunOS 5.x systems have these groups:

```
root::0:root
other::1:
bin::2:root,bin,daemon
sys::3:root,bin,sys,adm
adm::4:root,adm,daemon
uucp::5:root,uucp
mail::6:root
tty::7:root,tty,adm
lp::8:root,lp,adm
nuucp::9:root,nuucp
staff::10:
daemon::12:root,daemon
sysadmin::14:
nobody::60001:
noaccess::60002:
```

Note – You should add authorized users to the `sysadmin` group (GID=14) in the `group` file if you want them to administer systems using Administration Tool.

Creating New Groups

Many system administrators frequently create new group accounts. In the case of secondary groups, you must create the group and assign it a GID number before you can assign users to it.

User groups can be set up and administered by the system owner who wants to maintain the system files locally if he or she has root privileges. You can set up user groups on a network by duplicating the information in the local `/etc/group` file on each system. This approach makes updates difficult. A better approach is to use a name service, like NIS+, to maintain a network-wide group table. This also requires that the `nsswitch.conf` file is set up properly on each system. Either way, you should use the Administration Tool's Database Manager to create and maintain groups.

If you add a user to a group, either manually or by using Database Manager, and the user has NIS+ credentials, you must run the `nisaddcred local` command again to update the entry in the `cred` table. This step is not necessary when you use User Account Manager to add a user to a group.

Home Directory

The home directory is the portion of a file system allocated to a user for storing private files. The amount of space you allocate for a home directory depends on the kinds of files the user creates and the type of work done. Allocate at least 15 Mbytes of disk space for each user's home directory.

A home directory can be located either on the user's local system or on a remote file server. In either case, by convention the home directory is created as `/export/home/username`. On a large server that supports many home directories, there may be many different directories under `/export`, like `home1`, `home2`, `home3`, and so on, to store users' home directories. Each of these `/export/home n` directories should use a separate file system. Regardless of where their home directory is located, users usually access their home directories through a mount point named `/home/username`.

The preferred procedures described in this chapter assume the user's system is on a network and that the Automounter makes the home directory accessible. It makes no difference whether the home directory is on the user's local system or on a remote file server. You can just think of the client and the server as being the same system when the home directory is on the user's local system.

If a user's home directory resides on a server, the file system or directory is made available to the user's system using the `share` command. User Account Manager does not perform this step for you. You normally do this once as part of setting up the home directory servers on the network.

In addition, you need to define how the home directory is mounted by either:

- Adding an entry to the `auto_home` file so that the home directory is automatically mounted. Automounting is the preferred method. The User Account Manager automatically adds the required entry in when you choose the AutoHome Setup option.

or

- Adding an entry in `/etc/vfstab` file on the user's system to manually mount the home directory.

To support automatic mounting of home directories, the SunOS 5.x system software includes the `auto_master` file, which has the following entry:

```
/home      auto_home
```

This entry tells the Automounter to mount the directories specified in the `auto_home` file on the `/home` mount point on the local system. The entries in `auto_home` have the following format:

```
username      system-name: /export/home/username
```

When a user logs in with *username*, the Automounter mounts the specified directory (`/export/home/username`) from the server (*system-name*) on the `/home` mount point on the system where the user has just logged in.

This method works even when the home directory is stored on the same system the user has logged in to. But more importantly, this method allows the user to log in to another system and have his or her home directory mounted on `/home` on that system.

To use the home directory anywhere on the network, it should always be referred to as `$HOME`, not as `/export/home/username`. The latter is machine-specific. In addition, any symbolic links created in a user's home directory should use relative paths (for example, `../x/y/x`), so the links will be valid no matter on which system the home directory is mounted.

Note – When the Automounter is used to mount home directories, you are not permitted to create any directories under the `/home` mount point on any system. The system recognizes the special status of `/home` when the Automounter is active.

Initialization Files

The characteristics of a user’s working environment are determined at login. First, the `login` program sets a number of variables, like `HOME`, `LOGNAME`, and `TZ`. Then a file called the system profile (an initialization file) is run to set system-wide defaults like `PATH`, message of the day, and `umask`. Finally, a file (or two) called the user profile (another initialization file) is run that sets variables specific to the user. For example, the user profile may modify the `PATH` to include applications that only the user runs.

In addition to the system profile in the `/etc` directory on each system, each user may have one or more initialization files in their home directory. User Account Manager gives you the option to copy the user initialization files from a specified “skeleton” directory into the user’s home directory.

The user initialization files set the search path, environment variables, windowing environment, and other characteristics of the user’s environment, which are required to get the user up and running. The content of the user’s initialization files is affected by the shell that is assigned to him or her.

You define the default shell for each user when you create each user’s account. You can easily change the default shell by changing the appropriate field of a user’s entry in the `passwd` file.

Part of setting up a user’s home directory is providing initialization files for the shell specified in the user’s `passwd` entry. Each shell has its own initialization file (or files), which are shown in Table 1-2.

Table 1-2 User Initialization Files for Bourne, C, and Korn Shells

Shell	Initialization File	Purpose
Bourne	<code>\$HOME/.profile</code>	Defines user’s environment at login
C	<code>\$HOME/.cshrc</code>	Defines user’s environment for all C shells invoked after login shell

Table 1-2 User Initialization Files for Bourne, C, and Korn Shells (Continued)

Shell	Initialization File	Purpose
Korn	<code>\$HOME/.login</code>	Defines user's environment at login
	<code>\$HOME/.profile</code>	Defines user's environment at login
	<code>\$HOME/\$ENV</code>	Defines user's environment at login in the file specified by the Korn shell's <code>ENV</code> environment variable

The SunOS 5.x system software provides default user initialization files for each shell in the `/etc/skel` directory on each system, as shown in Table 1-3. You can use these as a starting point and modify them to create a standard set of files that will provide the user environment common to all users. After adding a user account, you can customize a specific user's initialization files.

Table 1-3 Default User Initialization Files

Shell	File Name
C	<code>/etc/skel/local.login</code>
C	<code>/etc/skel/local.cshrc</code>
Bourne or Korn	<code>/etc/skel/local.profile</code>

When you add a user, the initialization files for the appropriate shell must be copied from where you locate them to the user's home directory. You may want to set up a separate "skeleton" directory for each shell so that you do not need to delete extraneous files when you set up each user. If you want User Account Manager to populate a user's home directory with the appropriate initialization files, both the home directory and the "skeleton" directory must reside on the same system. Note that this is the system where the home directory originates, rather than the system where it eventually gets mounted. Normally, you would set up a system (or a few systems) to serve the home directories on the network.

System Initialization File

When users log in with valid user names and passwords, a system initialization file is run for each user's default shell. Each shell has its own system initialization file (or files), which are shown in Table 1-4.

Table 1-4 System Initialization Files for Bourne, C, and Korn Shells

Shell	System Initialization File
Bourne	/etc/profile
C	/etc/.login
Korn	/etc/profile

The system initialization file is an executable ASCII file that typically performs these actions:

- Defines and exports some environment variables
- Displays the message of the day
- Displays a list of news items if the user is not root
- Displays a message about mail items if the user has mail
- Defines the default permission for new files (`umask 022`)

You can edit the system initialization file to change existing definitions. For example, you can change the value assigned to your system mask from `022` to `077` to make files and directories created by users more secure. You can change the default `PATH` to provide access to locally developed commands.

You can also edit the system initialization file to automate certain routines. For example, you can add shell commands to display the current time and date, and tell users the number of people logged in.

To minimize your work, set system-wide default variables in the system initialization file rather than in the user initialization file.

User's Initialization Files

After the system runs the system initialization file, it runs the user's initialization files. The user's profile is one or more executable ASCII files that executes commands and shell scripts in the same way that the system profile does. In particular, the user's initialization file sets the user's home path, the search path, and other environment variables.

SunOS 5.x software provides default initialization files for the Bourne, C, and Korn shells in the `/etc/skel` directory. Starting with these default files, you can create your own standard versions to be copied into the user's home directory.

To minimize your work, set system-wide default variables in the system profile. That way you must maintain only two files (`/etc/profile` and `/etc/.login`) per system. Maintaining customized versions of user initialization files is time consuming. You may want to make some changes to the user initialization file supplied with SunOS 5.x system software, but after that, you should convey to users that they are responsible for changes specific to their own needs. For example, a user who needs to use an application would add it to his or her `PATH` environment variable.

Do not add specific references to the local system in the user's initialization file. You want the instructions in the initialization file to be valid regardless of the system to which the user logs in. For example:

- To make a user's home directory available anywhere on the network, always refer to the home directory with the variable `$HOME`. `$HOME` works even when the user logs in to another system.
- To access files on a local disk, use global path names, like `/net/machine-name/directory-name`. When the NIS+ `auto_master` and `hosts` files are set up properly on the network, any directory referenced by `/net/machine-name` can be mounted automatically on any system on which the user logs in.

Password, Shadow, and Group File Information Summary

Fields in the Password File

The fields in the `passwd` file are separated by colons, and contain the following information:

```
user-name:password:uid:gid:comment:home directory:login-shell
```

For example:

```
kryten:x:101:100:Kryten Series 4000:/export/home/kryten:/bin/csh
```

The table below describes the `passwd` file.

Table 1-5 Password File Fields

Field Name	Description
<i>user-name</i>	Contains the user or (login) name. User names should be unique and consist of 2-8 letters (A-Z, a-z) and numerals (0-9). The first character must be a letter, and at least one character must be a lowercase letter. User names cannot contain underscores.
<i>password</i>	Contains an x, a placeholder for the encrypted password. (Password is an obsolete field.). The encrypted password is stored in the <code>shadow</code> file.
<i>uid</i>	Contains a user identification number that identifies the user to the system. UIDs for regular users should range from 100 to 60000. All UID numbers should be unique.
<i>gid</i>	Contains a group identification number (GID) that identifies the user's primary group. Each GID must be a whole number between 0 and 60002 (60001 and 60002 are assigned to <code>nobody</code> and <code>noaccess</code> , respectively).
<i>comment</i>	Usually contains the full name of the user. (This field is informational only.) It is sometimes called the GCOS field because it was originally used to hold the login information needed to submit batch jobs to a mainframe running GECOS (General Electric Computer Operating System) from UNIX systems at Bell Labs.
<i>home directory</i>	Contains user's home directory pathname.
<i>login shell</i>	Contains the user's default login shell, which can be <code>/bin/sh</code> , <code>/bin/csh</code> or <code>/bin/ksh</code> . Table 1-12 on page 36 contains a description of shell features.

Fields in the Shadow File

The fields in the `shadow` file are separated by colons, and contain the following information:

```
user-name:password:lastchg:min:max:warn:inactive:expire
```

For example:

```
rimmer:86Kg/MNT/dGu.:8882:0::5:20:8978
```

The table below describes the shadow file.

Table 1-6 Shadow File Fields

Field Name	Description
<i>user-name</i>	Contains the user or (login) name.
<i>password</i>	May contain the following entries: a 13-character encrypted user password; the string *LK*, which indicates an inaccessible account, or the string NP, which indicates no password for the account.
<i>lastchg</i>	Indicates the number of days between January 1, 1970 and the last password modification date.
<i>min</i>	Contains the minimum number of days required between password changes.
<i>max</i>	Contains the maximum number of days the password is valid before the user is prompted to specify a new password.
<i>inactive</i>	Contains the number of inactivity days allowed for the user before the user's account is locked.
<i>expire</i>	Contains the absolute date when the user account expires. If exceeded, the user cannot log in to the system.

Fields in the Group File

The group database file contains the following fields:

```
group name:group password:gid:user-list
```

For example:

```
bin::2:root,bin,daemon
```


The table below describes the `group` file.

Table 1-7 Group File Fields

Field Name	Description
<i>group name</i>	Contains the name assigned to the group. For example, members of the chemistry department in a university may be called <code>chem</code> . Group names can have a maximum of nine characters.
<i>group password</i>	The Group Password field is a relic of earlier versions of UNIX. It is usually left empty or filled with an asterisk. <i>If a group has a password, the <code>newgrp</code> command prompts users to enter it. However, there is no utility to set the password.</i>
<i>group ID (gid)</i>	Contains the group's numeric ID. It must be unique on the local system, and should be unique across the entire organization. Each GID number must be a whole number between 0 and 60002 and numbers under 100 are reserved for system default group accounts). User groups range from 100 to 60000. (60001 and 60002 are assigned to <code>nobody</code> and <code>noaccess</code> , respectively.)
<i>user-list</i>	Contains a list of groups and a comma-separated list of user names, representing the users' secondary group memberships. Each user can belong to a maximum of 16 secondary groups.

Adding Users to a System and a Network

User Account Manager is a graphical user interface that enables to you to easily add user account information using these steps:

- Identify the user to the system and network by entering user information into the `passwd` file
- Define the user's primary group and optional secondary groups
- Create a password (optional, but highly recommended) and assign the account password aging and expiration attributes
- Create the user's home directory or specify an existing directory pathname
- Create the entry in the `auto_home` file that enables the user's home directory to be automatically mounted
- Set up the user's home directory with initialization files
- Specify the location of the user's mailbox
- Set up NIS+ credentials

One of User Account Manager's key features is setting up automounting of the user's home directory. The AutoHome Setup option enables users to log in to any system and have their home directories immediately available. This feature requires the appropriate settings in the `/etc/nsswitch.conf` files on the target systems. If you do not choose to set up automounting, you must manually set up the mounting required to make a user's home directory available on the network.

In addition to setting up the user's account, you may want to customize the working environment by:

- Enabling a default printer
- Modifying initialization files
- Setting up localization requirements

When entering the user account information, you must assign a primary group or accept the default: `nobody`. The primary group should already exist (if it doesn't exist, specify the group by GID). User names are not added to the member list of the primary group. If they were, the list might become too long. You can also specify optional secondary groups, in which case the user name is added to the member list. Secondary groups must exist beforehand.

Instructions for Setting Up User Accounts and Groups

This section provides step-by-step instructions for performing tasks related to setting up user accounts and groups.

The tasks are shown in the order in which you perform them, except you may want to add groups before adding user accounts. The first task of setting up the system and initialization (skeleton) files is a one-time job. After you set up the initialization files and store them on a server, User Account Manager can copy the user initialization files into the home directories of newly created user accounts.

With User Account Manager, all the steps required to add a user account are incorporated into the Add User and Copy User options under the Edit menu. If you chose to have the home directory automounted, the new user is ready to log in and work from anywhere on the network. If you do not choose to take advantage of automounting, after adding the user account with User Account Manager, you still must set up mounting, so the user can access his or her home directory.

Setting Up Initialization (Skeleton) Files

Prerequisites

- Root access on systems where initialization files will be modified.

Information You Need

- Name of servers on which files are located
- Name of directory storing system initialization file
- Name of directories storing shell-specific user initialization files
- Parameters of environment variables to be set

▼ How to Set Up the System Initialization File

- 1. Choose the system that will keep the standard version of the system initialization file you want to use for all other systems.**

Typically this is the same system from which you export home directories.

- 2. Edit /etc/profile.**

This is the system initialization file used when the user's login shell is the Bourne or Korn shell. Change the contents of the file to include commands, shell scripts, and variable definitions.

- 3. Edit /etc/.login.**

This is the system initialization file used when the user's login shell is the C shell. Change the contents of the file to include commands, shell scripts, and variable definitions.

- 4. Copy the two system initialization files to the /etc directory on each system in the network.**

```
cp /etc/.login /net/system_name/etc/.login
```

▼ How to Set Up the User Initialization File

- 1. On the system to be used as a server for home directories, log in as, or become superuser.**

This must be the system where the home directories are created and shared under `/export/home`, because User Account Manager restricts you to using the same system as the repository of the user initialization files and the sharer of the home directories.

- 2. Copy the default files from `/etc/skel/local.*` to separate shell-specific directories, as follows:**

These shell-specific directory names are what you enter in the Skeleton Path field in the Add User or Copy User window under User Account Manager.

- a. Create the three shell-specific directories.**

```
# mkdir /etc/skel/C; mkdir /etc/skel/B; mkdir /etc/skel/K
```

- b. Copy the C shell script into the `/etc/skel/C` directory.**

```
# cp /etc/skel/local.cshrc /etc/skel/C/.cshrc
```

- c. Copy the login script into the `/etc/skel/C` directory.**

```
# cp /etc/skel/local.login /etc/skel/C/.login
```

- d. Copy the Bourne shell script into the `/etc/skel/B` directory.**

```
# cp /etc/skel/local.profile /etc/skel/B/.profile
```

- e. Copy the Bourne shell script into the `/etc/skel/K` directory.**

```
# cp /etc/skel/local.profile /etc/skel/K/.profile
```

3. **Edit *cshell_directory/.cshrc*.**
This is the C shell initialization file that is run every time the user invokes a C shell after initial login. Change the contents of the file to include commands, shell scripts, and variable definitions.
4. **Edit *cshell_directory/.login*.**
This is the C shell login file that is run once when the user logs in. Change the contents of the file to include commands, shell scripts, and variable definitions.
5. **Edit *bourshell_directory/.profile*.**
This is the Bourne shell initialization file that is run once when the user logs in. Change the contents of the file to include commands, shell scripts, and variable definitions.
6. **Edit *kornshell_directory/.profile*.**
This is the Korn shell initialization file that is run once when the user logs in. Change the contents of the file to include commands, shell scripts, and variable definitions.
7. **In each shell-specific directory, type `chmod 744 .*` and press Return.**
Permissions are set for the initialization files.

Note – Although it is not strictly part of setting up the initialization files, while you are working on the home directory server, you should make sure the server is sharing the home directories. For instructions, turn to “Changing a User’s Home Directory” on page 49.

Adding a Group

Prerequisites

- Start OpenWindows, if necessary
- Start Administration Tool, if necessary
- Verify required access privileges

See “Before Using Administration Tool” on page 3 for more information.

Information You Need

- Group name to give to the new group
- Group ID (GID) to use for the new group
- User names and associated group names

▼ How to Add a Group

- 1. Start Database Manager, select the group file, then select the name service in effect on your network, and click on Load.**

The Database Manager main window displays.

- 2. (Optional) Scroll through the list or use the View (Sort and Search) options to view group entries.**

This lets you see which groups are defined before you add a group. At first, the group entries are sorted by Group Name.

Note – The `nogroup` entry serves a purpose in heterogenous networks running both NIS and NIS+. It is used by the NFS software. If you have a network running only NIS+, this entry is not needed.

- 3. Choose the Add Entry option from the Edit menu.**

The Add Entry window appears.

- 4. Type the Group Name and the Group ID in the appropriate text fields.**

If you make a mistake or change your mind, click on the Reset button and retype the information.

- 5. Click on Add.**

The group is added to the group file. Repeat steps 1 through 5 to add more groups. In addition, if you are not using a name service and want to add the groups on other systems, you also must repeat step 1, and specify a different system (host) name in the `Use /etc files on host:` field in the Load Database window.

If possible, create secondary groups before you add user accounts that refer to those groups. When you add a user account with User Account Manager and assign the user to secondary groups, the user's NIS+ credentials are updated automatically.

Adding a User Account

Prerequisites

- Start OpenWindows, if necessary
- Start Administration Tool, if necessary
- Verify required access privileges

See “Before Using Administration Tool” on page 3 for more information.

Information You Need

- User (login) name
- User ID (UID)
- Primary group ID (GID)
- Secondary group memberships, if any
- Identifying information (name, office, extension, home phone)
- Home directory name
- Login shell program name

▼ **How to Add a User Account**

1. If not done already, set up the system and user initialization files.

The User Account Manager can then automatically copy these files into a user’s home directory.

2. Click on the User Account Manager icon.

User Account Manager allows you to set up and administer user accounts. The window for indicating which type of name service you have appears.

3. Choose the name service in use on your network.

4. Click on Apply.

The User Account Manager main window appears. The following example reflects a choice of NIS+ as the name service and the default of showing all user entries.

5. Select Add User from the Edit menu.

After you have added a prototype user, you may find it preferable to add new users with the Copy User option under Edit, instead of the Add User option. The input windows are the same. All of the fields in the Copy User window, except User Name, User ID, and Password, are automatically filled in from the existing user account.

The Add User window appears.

6. In the User Name field, type the user's login name.

Choose a name unique to your organization. The name can contain from two to eight letters and numerals. The first character must be a letter, and at least one character must be a lowercase letter. The name cannot contain underscores.

7. In the User ID field, type a decimal number to identify the user.

Choose a number between 100 and 60000 that is unique within your organization.

8. In the Primary Group field, type the group name or a decimal number identifying the user's primary group.

If the group does not exist, you must enter a number, rather than a name.

9. (Optional) In the Secondary Group field, type the names or numbers of additional groups to which the user will belong, separated by spaces.

The referenced group(s) must exist beforehand.

10. In the Comment field, type useful information about the user.

You can indicate the user's full name, phone number, department, and so on, for informational purposes only.

11. On the Login Shell button, make your choice from the pull-down menu.

The default login shell is the Bourne shell unless you specify a different shell. If the shell program you want to select is not listed or does not have the exact path name shown, select Other and enter the path name in the adjacent text field.

12. On the Password button, specify your password status from the pull-down menu.

The choices for password status are described in Table 1-8. By default the password status is set to `Cleared until first login`. If you choose `Normal password`, a pop-up window appears in which you type an asterisk-echoed password twice. With NIS+, a colon (`:`) is not allowed in the password.

Table 1-8 Password Status Choices in User Account Manager

Password Status	Description
Cleared until first login	(Default) Account does not have a password;user is prompted for password on first login, unless <code>passreq=no</code> is set in <code>/etc/default/login</code> .
Account is locked	Account is disabled with an invalid password and can be unlocked by assigning a new password. This type of account allows a user to own files but not to log in.
No password -- setuid only	Account cannot be logged in to directly. This allows programs like <code>lp</code> or <code>uucp</code> to run under an account, without allowing a user to log in.
Normal password...	Account will have a password, which you set in the pop-up window that appears.

13. (Optional) In the remaining Account Security fields, enter values to set the account security.

Use the descriptions of the Account Security fields in Table 1-9 to decide which values to set. In the case of Expiration Date, there are separate pull-down menus for day, month, and year.

Table 1-9 Account Security Fields in the Add User Window

Field	Description
Min Change:	The minimum number of days allowed between password changes, which is intended to prevent a user from changing his or her password and then changing it back a few seconds later. Default is 0.
Max Change:	The maximum number of days the password is valid before it must be changed; otherwise, the account is locked. Blank means the password never has to be changed.
Max Inactive:	The maximum number of days an account may go without being accessed before it is automatically locked. Blank means the account remains active no matter how long it goes unused.
Expiration Date:	Date the user account expires. None means no expiration.
Warning:	Number of days to begin warning the user before the password expires. Blank means no warning is given.

14. (Optional) Click on the Create Home Dir check box to have the user's home directory automatically created.

Use the Path and Server fields to point to an existing directory, or to specify a new directory to create. In the latter case, the Skeleton Path and Permissions become active. User Account Manager must be installed on the system where the home directory is to be created.

15. In the Path field, type the path of an existing home directory or one to be created.

The path name refers to the directory on the server, not a remotely-mounted directory. In the standalone case, the server and the user's system are the same. By convention, home directories should be named `/export/homen/username`. Add User creates the directory on the system specified in the next field. The path name of the home directory is stored in

the `passwd` file exactly as specified, unless AutoHome Setup is checked. If you choose to take advantage of automounting, the mount point, `/home/username`, is stored.

16. In the Server field, type the server name where the home directory will reside.

The server must have User Account Manager installed on it.

17. (Optional) In the Skeleton Path field, type the path to the directory that stores the user initialization (skeleton) files that will be copied into the user's home directory.

If you fill in this field, the shell-specific initialization files are copied from the specified skeleton path directory on the designated server to the specified home directory on the server. The initialization files and home directories must be located on the same (server) system.

18. (Optional) Click on the AutoHome Setup check box if you want automounting to be set up.

Automounting enables the user's home directory to be automatically mounted under `/home/username` on any system in the network. The information required for such mounting is stored in the `auto_home` file. The AutoHome Setup field does not run the `share` command on the server to export the `/export/home` directory under which the home directories for multiple users may be kept. Normally, you would do that in setting up the home directory server on the network.

19. Click on the Permissions check boxes to set the Read, Write, and Execute permissions for the user's home directory.

Permissions can be set only when you are creating a home directory.

20. (Optional) In the Mail Server field, type the host name of the system where the user's mailbox is to reside.

This creates an entry (`username@mail-server-name`) in either the NIS+ `mail_aliases` table or the local `/etc/aliases` file that identifies where the user's mail will be directed—typically his or her desktop system.

21. (Optional) Click on the Cred. Table Setup check box click if you want the user to be added to the NIS+ cred table.

This field appears only if you have selected NIS+ as the name service. As part of NIS+ security, users should be added to the `cred` table so their identities can be authenticated.

22. Click on Add.

The user information is added to the `passwd`, `group`, `auto_home`, `mail_aliases`, and `cred` (tables or local `/etc` files), accordingly. If you get a message that contains “cannot execute method, access denied,” edit the NIS+ `group` table or the `/etc/group` file and add your user name to the `sysadmin` group, whose GID is 14.

23. If you are not using a name service, repeat this procedure for each system that will have the user account.***Completing the User Setup:***

To customize the user environment, set environment variables in the user’s initialization files, provide access to printing services, and mount the user’s mailbox from a remote mail server.

Mounting a User’s Home Directory

Prerequisites

- Start OpenWindows, if necessary
- Start Administration Tool, if necessary
- Verify required access privileges

See “Before Using Administration Tool” on page 3 for more information.

Information You Need

- User’s login name
- Name of the home directory to be mounted

Either set up automounting for the home directory, or mount the home directory manually, as described below.

Note – Using the Automounter with the NIS+ `auto_home` table is an efficient way to make users’ home directories available anywhere on the network.

▼ How to Automount the Home Directory

Use User Account Manager to set up automounting of home directories when you add or modify a user account.

Note – The following steps apply whether the home directory is created on the local system or on a remote file server.

- 1. Click on the Database Manager icon.**
The Load Database window is displayed.
- 2. If necessary, select Auto_home by clicking on the Auto_home entry.**
The Auto_home entry is highlighted.
- 3. Choose the name service used to administer the network by clicking on the appropriate check box.**
- 4. Click on Load.**
The Auto_home Database window is displayed.
- 5. Choose the Add Entry option from the Edit menu.**
The Add Entry window is displayed.
- 6. Type *username* in the User Name text field.**
- 7. Type *system-name:/export/homen/username* in the Path text field.**
When the home directory is located on the user's system, the *system-name* is the user's system. More often, *system-name* will be the name of a file server.
- 8. Click on Add.**
The information is added to the `auto_home` file. The first time the user logs in, his or her home directory is automatically mounted under `/home/username`.
- 9. If you are not using a name service, repeat this procedure to update the `/etc/auto_home` file on each system that will have the user account.**

▼ How to Manually Mount the Home Directory

If the directory for a user's home directory is located on another system and the Automounter is not being used to make that available, perform the following steps:

1. Log in as root on the user's system.

2. Edit the `/etc/vfstab` file, create an entry for the user's home directory, and save the file.

For example, to create an entry for user `ignatz`, with a home directory on server `venus`, you add the following line:

```
venus:/export/home1/ignatz - /home/ignatz nfs - yes rw,intr
```

3. Type `mkdir /home/username` and press Return.

This creates the mount point on the user's system where the remote home directory is mounted. Note that the home directory does not have the same name on the user's system as it does on the server. For example, `/export/home/ignatz` on the server would be mounted as `/home/ignatz` on the user's system. This step is not necessary when the home directory is mounted using the Automounter.

4. Type `mountall` and press Return.

All entries in the current `vfstab` file (whose `automnt` fields are set to `yes`) are mounted.

5. To verify that all entries are mounted, type `mount` and press Return.

The file systems that are mounted are displayed.

Customizing a User's Environment

Information You Need

- User's login name and password

▼ **How to Customize a User's Environment**

1. On the user's system, log in as the user.

This step allows you to see the user's environment as he or she will see it.

2. Edit the user's initialization files.

The following steps suggest some changes and show the shell-specific syntax to use. See step 3 for changes related to localization (for different locales or countries). However, you should minimize changes to the user's initialization file by putting as many of these kinds of changes as you can in the system initialization file on the systems in the network.

a. Set the user's default path to include the home directory and directories or mount points for the user's windowing environment and applications.

Do not use path names that are specific to a local system. Use global `/net/machine-name/directory-name` path names, so files can be mounted automatically on any system to which the user logs in. This capability requires that the `auto_master` file is set up properly on the network.

To change the path setting, add or modify the line for `PATH` as follows:

◆ For the Bourne or Korn shell, type

```
PATH=.: /dirname1: /dirname2: /dirname3...; export PATH
```

For example, enter a line like the following in the user's

`$HOME/.profile` file:

```
PATH=.: /usr/bin: /$HOME/bin: /net/glrr/files1/bin
export PATH
```

The “dot” (.) at the beginning of the list means “current directory” and can be omitted if security is a concern.

◆ For the C shell, type

```
set path = ( . /dirname1 /dirname2 /dirname3... )
```

For example, enter a line like the following in the user's

`$HOME/.cshrc` file:

```
set path=(. /usr/bin $HOME/bin /net/glrr/files1/bin)
```

The “dot” (.) at the beginning of the list means “current directory” and can be omitted if security is a concern.

- b. Check that the environment variables are set to the correct directories or mount points for the user's windowing environments and third-party applications. Type `env` and press Return.**

```
$ env
HOME=/home/ignatz
HZ=100
LOGNAME=ignatz
MAIL=/var/mail/ignatz
MANSECTS=\1:1m:1c:1f:1s:1b:2:\3:3c:3i:3n:3m:3k:3g:3e:3x11:3
xt:3w:3b:9:4:5:7:8
PATH=/usr/bin
SHELL=/bin/sh
TERM=sun
TZ=EST5EDT
$
```

- c. Add or change the settings of environment variables by entering either of the following lines:**

- ♦ **For the Bourne or Korn shell, type**

`VARIABLE=value;export VARIABLE`

The following line sets the user's default mail directory:

```
MAIL=/var/mail/ignatz;export MAIL
```

- ♦ **For the C shell, type `setenv VARIABLE value`**

The following line sets the history recorded to the last 100 commands:

```
setenv HISTORY 100
```

Note – For a user's home directory to be available anywhere on the network, refer to it with the variable `$HOME`. For example, use `$HOME/bin`; do not use `/export/home/username/bin`. `$HOME` works when the user logs in to another system, when home directories are automounted.

- d. Check the `umask` setting to see if it is appropriate. If necessary, change it by typing `umask nnn` and press Return.

You can either include or omit leading zeros.

For example, to set file permissions to 644, type:

```
$ umask 022
```

Table 1-10 shows the file permissions that are created for each of the octal values of `umask`.

Table 1-10 Permissions for `umask` Values

<code>umask</code> Octal Value	File Permissions
0	<code>rwX</code>
1	<code>rw-</code>
2	<code>r-X</code>
3	<code>r--</code>
4	<code>-wX</code>
5	<code>-w-</code>
6	<code>--X</code>
7	<code>---</code> (none)

See “Default File Permissions (`umask`)” on page 41 for information about the `umask` command.

3. If necessary, configure the `LANG` variable and `stty` settings for different locales, as listed in Table 1-11.

The `LANG` and `LC` environment variables determine which locale-specific conversions and conventions the shell will apply, like time zones, collation orders, and formats of dates, time, currency, and numbers. In addition, an `stty` setting determines whether the system will support multibyte characters.

a. Set the LANG and LC variables for the locale in the user's shell initialization file.

The default is the "C" locale. For the Bourne shell use the format: `VARIABLE=value; export VARIABLE`.

For the C shell use the format: `setenv variable value`.

Table 1-11 Values for LANG and LC Variables

Value	Locale
de	German
fr	French
iso_8895_1	English and European
it	Italian
japanese	Japanese
korean	Korean
sv	Swedish
tchinese	Taiwanese

LANG sets all possible conversions and conventions for the given locale. If you have special needs, you can set various aspects of localization separately through these LC variables: LC_COLLATE, LC_CTYPE, LC_MESSAGES, LC_NUMERIC, LC_MONETARY, and LC_TIME.

b. If the system needs to support multibyte characters (for example, Japanese characters), add the following command to the system initialization file (/etc/profile or /etc/.login):

```
stty cs8 defeucw
```

To enable access to printers:

After adding users to a system, you should make sure they have access to a printer. To set up a printer, see Chapter 3, "Setting Up Printers," for instructions. When printers are set up and running on the system or network, but one is not available, see "Making Printers Available" on page 103. If your site policy is to create an "allow" list for a printer so only users explicitly included in the list can use the printer, add the new user to the list as follows:

♦ **Log in as root, type `lpadmin -p printername -u allow:username` and press Return.**

To set up mail accounts:

When the mail service is already running on the network, you can perform two tasks to add a new mail client to the network:

- Set up the mounting of a mailbox for the user (mail client) that will be mounted from a mail server.
- Add the user to existing mail aliases.

User Account Reference Information**Collect and Record User Information**

You may find it useful to create a form like the one below to gather information about users. The items above the double line reflect the information specified when adding a user account with User Account Manager.

Item	Description
User Name:	_____
UID:	_____
Primary Group:	_____
Secondary Groups:	_____
Comment:	_____
Default Shell:	_____
Password Status and Aging:	_____
Home Directory Server Name:	_____
Home Directory Path Name:	_____
Mounting Method:	_____
Permissions on Home Directory:	_____
Mail Server:	=====
Department Name:	_____
Department Administrator:	_____
Manager:	_____
Employee Name:	_____

Employee Title:	_____
Employee Status:	Contract or Permanent
Employee Number:	_____
Start Date:	_____
Add to These Mail Aliases:	_____
Desktop System Name:	_____

Shell Features

Table 1-12 lists basic shell features, and shows which shells provide each feature.

Table 1-12 Basic Features of Bourne, C, and Korn Shells

Feature	Bourne	C	Korn
Syntax compatible with <code>sh</code>	Yes	No	Yes
Job control	Yes	Yes	Yes
History list	No	Yes	Yes
Command-line editing	No	Yes	Yes
Aliases	No	Yes	Yes
Single-character abbreviation for login directory	No	Yes	Yes
Protect files from overwriting (<code>noclobber</code>)	No	Yes	Yes
Ignore CTRL-D (<code>ignoreeof</code>)	No	Yes	Yes
Enhanced <code>cd</code>	No	Yes	Yes
Initialization file separate from <code>.profile</code>	No	Yes	Yes
Logout file	No	Yes	No

Environment Variables and Shell Variables

The shell maintains an *environment* that includes a set of variables defined by the `login` program, the system initialization file, and the user's initialization files. In addition, some variables are defined by default. By definition, *environment variables* are variables exported to all processes spawned by the shell. Their settings can be seen with the `env` command. A subset of environment variables, like `PATH`, affects the behavior of the shell itself.

In the standard (Bourne or Korn) shell, environment variables are set with an assignment of `NAME=value` followed by an `export NAME` statement. In the C shell, the `setenv NAME value` statement both assigns the value and exports it to generated processes.

In addition to environment variables, the shell also maintains local or “shell variables” that affect only the current shell. In the C shell, a set of these shell variables have a special relationship to a corresponding set of environment variables. These shell variables are `user`, `term`, `home`, and `path`. The value of the environment variable counterpart is used initially to set the shell variable; when the shell variable is reset (with the `set name = value` statement), the environment variable counterpart is updated. In the C shell, you use the lowercase names to set these variables. In the Bourne and Korn shells, you use the uppercase names to set these variables and environment variables, in general. For all shells you refer to these variables and environment variables, in general, by the uppercase names.

You can customize the shell environment by changing the values of the predefined environment variables and by specifying additional environment variables. Environment variables can be stored in the system initialization file (`/etc/profile` and `/etc/.login`) or the user's `$HOME/.profile`, `$HOME/.login`, or `$HOME/.cshrc` file.

The following table describes some environment and shell variables you may want to customize either in the system initialization file or in a specific user's initialization files. For more information about variables used by the different shells, see `sh(1)`, `ksh(1)`, or `csh(1)`.

Table 1-13 Shell Variable Descriptions (1 of 3)

Shell Variable	Description
ARCH	Sets the user's system architecture (for example, <code>sun4, i386</code>). Set this variable with <code>ARCH = `uname -p`</code> (in Bourne or Korn shells) or <code>setenv ARCH `uname -p`</code> (in C shell). There is no built-in behavior of the shell that depends on this variable. It's just a useful variable for branching within shell scripts.
CALENDAR	Sets the path to the Calendar executables.
CDPATH (<code>cdpath</code> in the C shell)	Sets a variable used by the <code>cd</code> command. If the target directory of the <code>cd</code> command is specified as a relative path name, the <code>cd</code> command will first look for the target directory in the current directory (“.”). If the target is not found, the path names listed in the <code>CDPATH</code> variable are searched consecutively until the target directory is found and the directory change is completed. If the target directory is not found, the current working directory is left unmodified. For example, the <code>CDPATH</code> variable is set to <code>/home/jean</code> , and two directories exist under <code>/home/jean</code> : <code>bin</code> and <code>rje</code> . If you are in the <code>/home/jean/bin</code> directory and type <code>cd rje</code> , you change directories to <code>/home/jean/rje</code> , even though you do not specify a full path
DESKSET	Sets the path to the DeskSet executables.
history	Sets history for the C shell.
HOME	(or <code>home</code> in the C shell)
LANG	Sets the locale.
LOGNAME	Defines the name of the user currently logged in. The default value of <code>LOGNAME</code> is automatically set by the login program to the user name specified in the <code>passwd</code> file. You should only need to reference (not reset) this variable.
LPDEST	Sets the user's default printer.
MAIL	Sets the path to the user's mailbox.
MANPATH	Sets the hierarchies of <code>man</code> pages available.

Table 1-13 Shell Variable Descriptions (2 of 3)

Shell Variable	Description
MANSECTS	Sets the hierarchies of man pages available.
OPENWINHOME	Sets the path to the OpenWindows subsystem.
PATH (or path in the C shell)	Lists, in order, the directories that the shell searches to find the program to run when the user types a command. If the directory is not in the search path, users must type the complete path name of a command. The default PATH is automatically defined and set as specified in .profile (Bourne or Korn shell) or .cshrc (C shell) as part of the login process. The order of the search path is important. When identical commands exist in different locations, the first command found with that name is used. For example, suppose that PATH is defined (in Bourne and Korn shell syntax) as PATH=/bin:/usr/bin:/usr/sbin:\$HOME/bin and a file named sample resides in both /usr/bin and /home/jean/bin. If the user types the command sample without specifying its full path name, the version found in /usr/bin is used.
prompt	Defines the shell prompt for the C shell.
PS1	Defines the shell prompt for the Bourne or Korn shell.
SHELL (or shell in the C shell)	Sets the default shell used by make, vi, and other tools.
TERMINFO	Specifies the path name for an unsupported terminal that has been added to the terminfo database. Use the TERMINFO variable in /etc/profile or /etc/.login. When the TERMINFO environment variable is set, the system first checks the TERMINFO path defined by the user. If it does not find a definition for a terminal in the TERMINFO directory defined by the user, it searches the default directory, /usr/share/lib/terminfo, for a definition. If it does not find a definition in either location, the terminal is identified as “dumb.”
TERM (or term in the C shell)	Defines the terminal. This variable should be reset in /etc/profile or /etc/.login. When the user invokes an editor, the system looks for a file with the same name as the definition of this environment variable. The system searches the directory referenced by TERMINFO to determine the terminal characteristics.

Table 1-13 Shell Variable Descriptions (3 of 3)

Shell Variable	Description
TZ	Sets the time zone, which is used to display dates, for example, in the <code>ls -l</code> command. If TZ is not set in the user's environment, the system setting is used; otherwise, Greenwich Mean Time is used.

Using the PATH Variable

When the user executes a command using the full path name, the shell finds the command using that path name. However, when users specify only a command name, the shell searches the directories for the command in the order specified by the PATH variable. If the command is found in one of the directories, the shell executes it.

A default PATH is set by the system, but most users modify it to add other command directories. Many user problems related to setting up the environment and accessing the right version of a command or a tool can be traced to incorrectly defined paths.

Here are some guidelines for setting up efficient PATH variables:

- If security is not a concern, put the current working directory (.) first in the path: `PATH=.: /usr/bin` for the Bourne and Korn shells; `set path = (. /usr/bin)` for the C shell. Including the current working directory in the path poses a security risk that you may want to avoid, especially for root. See *Security, Performance, and Accounting Administration*. The C shell updates the environment variable PATH when the shell variable path is reset.
- Keep the search path as short as possible. The shell searches each directory in the path. If a command is not found, long searches can slow down system performance.
- \$PATH is read from left to right, so the most likely place to find a command should be at the beginning of the path.
- Make sure directories are not duplicated in the path.
- Avoid searching large directories, if possible. Put large directories at the end of the path.
- Put local directories before NFS-mounted directories to lessen the chance of “hanging” when the NFS server does not respond.

Default File Permissions (umask)

The default file permissions assigned when a file or directory is created are controlled by the *user mask*. The user mask is set by `umask` in either the `.cshrc` or the `.profile` initialization file (depending on the user's default shell). You can display the current value of the user mask by typing `umask` and pressing Return.

`umask` is set with a three-digit octal value. The first digit sets permissions for the user; the second sets permissions for group; the third sets permissions for other (also referred to as world). Note that if the first digit is zero, it is not displayed. For example, if `umask` is set to `022`, `22` is displayed.

If `umask` is set to `000`, each file created has permissions `rw-rw-rw-` (mode `666`). Each directory created has default permissions `rxwxrwxrwx` (mode `777`). However, you can change `umask` value.

To determine the value you want to set, subtract the value of the permissions you want from `666` or `777`. The remainder is the value to use for `umask`.

For example, suppose you want to change the default mode for files from `666` (`rw-rw-rw-`) to `644` (`rw-r--r--`). The difference, `022`, is the value you use as an argument to `umask` command.

Message of the Day

The message of the day (MOTD) facility lets you send announcements or inquiries to all users of a system by adding the text of your message to the `/etc/motd` file. When users log in, the messages are displayed as part of the login process. Here is an example of a message of the day:

```
The system will be down from 1700 to 2300 hours on Friday,  
September 30, for upgrades and preventive maintenance.
```

Edit the `/etc/motd` file regularly to remove obsolete notices. Use the message of the day sparingly; if you inundate users with trivial messages, they will learn to ignore all your messages.

Administering User Accounts and Groups



This chapter describes tasks for administering user accounts and groups.

If you want to skip the background information that explains the concepts of administering user accounts and groups, and proceed directly to step-by-step instructions, use the following table to find the page where the instructions for a specific task begin.

<i>Changing a User's Home Directory</i>	<i>page 49</i>
<i>How to Modify a User Account</i>	<i>page 55</i>
<i>How to Delete a User Account</i>	<i>page 57</i>
<i>How to Disable a User Account</i>	<i>page 58</i>
<i>Modifying or Deleting Groups</i>	<i>page 58</i>
<i>How to Create or Change a Password</i>	<i>page 60</i>

If you want to review background information first, read “Using Administration Tool” on page 44.

If you want background information about setting up user accounts and groups, see Chapter 1, “Setting Up User Accounts and Groups,” on page 1.

About Administering User Accounts and Groups

This chapter describes how to administer user accounts and groups in a network environment. The same procedures apply to standalone systems. The method you use to administer users and groups on a network depend on whether the network is administered through a name service.

Administration Tool enables you to administer user accounts on a local or remote system or in a name service environment. With a name service like NIS+, you can manage network information in a centralized location so that important system information, such as system and user names, do not have to be duplicated on every system in the network.

Using Administration Tool

Use the following Administration Tool applications to manage user account information:

- User Account Manager to set up and maintain user accounts
- Database Manager to set up and maintain UNIX[®] group information

Depending on the name service used on the network, the information defining user accounts and groups is stored in either:

- NIS+ tables
- NIS maps
- Local */etc* files

To avoid confusion, all three types of information will be referred to as the *passwd* or *group file*, rather than the *passwd* or *group file, table, or map*.

Note – You can view the information in NIS maps with Administration Tool, but you cannot change the information. Refer to the SunOS 4.1 system documentation for information about how to administer NIS.

Before Using Administration Tool

The following hardware and software requirements must be met before using Administration Tool:

- SunOS 5.x software on all systems to be administered.
- A bit-mapped display monitor – Administration Tool’s applications can be used only on a system where the console is a bit-mapped screen.
- OpenWindows software. Start OpenWindows, if necessary.

```
$ /usr/openwin/bin/openwin
```

Required Access Privileges for Setting Up User Accounts

Table 2-1 describes the required access privileges for setting up user accounts.

Table 2-1 Required Access Privileges for Setting Up User Accounts

To Set Up User Accounts Using the	The Required Access Privileges Are:
/etc files	Root access or membership in the sysadmin group (GID=14) on the local or remote system.
NIS+ tables	Membership in the sysadmin group (GID=14) in the NIS+ group table. Create and destroy permissions on the NIS+ passwd, group, auto_home, mail_aliases, and cred tables. NIS+ permissions are granted by membership in an NIS+ group.

Traditional UNIX groups and NIS+ groups are separate. See *Name Services Administration Guide* for instructions on setting up NIS+ groups.

Using Administration Tool

Start Administration Tool from an OpenWindows window as follows.

```
$ admintool &
```

Administering User Accounts

Administering user accounts includes modifying, removing, and disabling user accounts.

Modifying User Accounts

Unless you define a user (login) name or numeric user ID (UID) that conflicts with an existing one, you should never need to modify a user account's login name or UID.

Users' group memberships will change. User Account Manager's Modify User option lets you add or delete a user's secondary groups. Alternatively, you can use Database Manager to directly modify a group's member list.

In most companies, users transfer departments, receive promotions, change projects, and so on. You may need to change the status of these users when such an event occurs. This information is contained in the Comment field. You can modify the Comment information in the NIS+ `passwd` table or local `/etc/passwd` file for user accounts using Administration Tool's User Account Manager application.

In a network environment, you probably will move users from one system to another, and from one server to another as they are assigned new equipment or as they move from one location to another. User Account Manager lets you change the identity of a user's home directory, but it does not create the new directory, or move the contents of an existing home directory. Create the new directory, move the files to it, and then use User Account Manager to specify the new location and have the new home directory automatically mounted. In addition, if you do not use User Account Manager to set up the user's home directory, you may have to change the `auto_home` file that defines how to automount the user's home directory.

You can use User Account Manager to change a user's password attributes. Alternatively, you can use one of the following commands, depending on which name service (or no name service) case applies: `passwd` (no name service), `yppasswd` (NIS), or `nispasswd` (NIS+).

Removing User Accounts

Before removing a user account, you might consider whether you just want to lock the account to disable it. See the next section, "Disabling User Accounts."

To remove an account, you have to reverse the tasks performed to set up the account.

When you delete a user account with User Account Manager, the entries in all the files that were made when adding the account are automatically removed. All the user's entries are removed from the following:

- `passwd` file
- `group` file
- `auto_home` file
- `mail_alias` or `aliases` files
- NIS+ `cred` table (if applicable)

In addition, you have the option to delete the files in the user's home directory and delete the contents of the user's mailbox.

If you want to delete entries from mail aliases other than the one set up to direct mail to your mailbox, you would have to delete them by hand.

Never reuse the UIDs from deleted accounts. This leaves holes in the `passwd` file, but minimizes security risks. However, if you have a high turnover of users (as in a university setting) you may have to reuse UIDs. In that case, "wipe the slate clean" so the new user is not affected by attributes set for a former user. For example, a former user may have been denied access to a printer—by being included in a printer deny list—but that attribute may not be appropriate for the new user.

Disabling User Accounts

Occasionally, you may need to temporarily or permanently disable a login account. The easiest way to disable a login account is to use User Account Manager to lock the password for an account. The content of the password field is changed to `*LK*`.

Or, on a local system, you can control access to a user's account by requiring password aging, by setting an expiration date for the login account, or by requiring that a user access the account at regular intervals. Another way that you can disable a login is to change the password.

Creating and Modifying Passwords

Passwords are an important part of system security. Each user account should be assigned a password of six to eight characters and containing a combination of letters and numbers. See *Security, Performance, and Accounting Administration* for detailed information about passwords and password aging. See also

`passwd(1)`, `yppasswd(1)`, or `nispasswd(1)` for information about changing passwords and password attributes (for example, maximum number of days a password is valid).

In Solaris, user information is stored in the NIS+ `passwd` table (or local `/etc/passwd` file). The encrypted password is stored in the `Passwd` field of the NIS+ `passwd` table or in the local `/etc/shadow` files, neither of which have general Read permissions.

You can use User Account Manager for some password administration, including clearing the password field until the user logs in for the first time, locking the account, and specifying time-outs and aging information. You can set the Password Status to prompt users to create their own password during their first login. Alternatively, you can create the password for the user and tell the user what it is before the first login.

To create passwords or modify passwords, you can also use one of these commands:

- `/usr/bin/passwd` (for no name service)
- `/usr/bin/nispasswd` (for the NIS+ name service)
- `/usr/bin/yppasswd` (for the NIS name service)

If your network is running the NIS+ name service, you should use `nispasswd`, because it automatically updates the user's credentials (LOCAL and DES entries in the `cred` table).

Administering Groups

Administering groups means you can modify or delete groups.

Modifying or Deleting Groups

Use Administration Tool's Database Manager to add or remove users from a group. This distributed application allows you to change to the NIS+ `group` table or `/etc/group` files on the network. However, if you use Database Manager to change a user's group memberships in a network running NIS+, you have to update the user's NIS+ credentials afterward (`nisaddcred local`).

When you add a user account, User Account Manager lets you define the user's primary and secondary groups. You can also use the application to modify a user's account to change a specific user's membership in secondary groups. In this case, if the network is running NIS+, the user's NIS+ credentials are automatically updated to reflect his or her group memberships.

Database Manager enables you to delete groups. When projects finish, groups that are set up for those projects may no longer be needed, and you may want to delete these groups. Be careful to avoid conflicts if you reuse the GIDs from deleted groups.

Instructions for Administering User Accounts and Groups

This section provides step-by-step instructions for performing tasks related to administering user accounts and groups. For many tasks, you will find an example of user input and system output after the instructions.

This section describes how to:

- Change a user's home directory
- Modify a user account
- Delete a user account
- Disable a user account
- Modify or delete groups
- Create or change a password

Changing a User's Home Directory

Prerequisite

- User's account exists
- Root access on the systems containing the home directories

Information You Need

- User's login name and user ID (UID)

A user's home directory can be automatically created by User Account Manager when you add a user. If you need to move a user's home directory, User Account Manager will not create the new directory nor move the contents of the old directory. You have to do both manually.

The following procedures tell you how to:

- Create a new home directory
- Copy the contents of an old home directory to a new home directory
- Copy the initialization (skeleton) files into the directory and customize the user's environment as required (this procedure is not needed when copying all the files from an existing home directory to a new one)
- Set up the mounting required to make the home directory available (this procedure is not needed when you use the AutoHome Setup option in User Account Manager)

Perform these procedures before using User Account Manager to modify the other attributes of the user account.

▼ How to Create the New Home Directory

Note – All the following steps apply whether the home directory is created on the local system or on a remote file server.

1. **Decide on which system to create the home directory for a user.**

If the home directory is accessed over the network, the system that provides the home directory should be on the same network segment as the user's local system. Check that there is enough space on the possible servers by using the `df` command.

2. **Log in to the system where you want to create the home directory.**

Usually, the system is a file server, but it can be the user's own local system.

3. **Type `cd /export/home-dir` and press Return.**

If you use a different home directory naming scheme, change to a directory where you assign users' home directories.

```
# cd /export/home1
```

4. Type `mkdir login-name` and press Return.

Create a directory whose name matches the login name of the user.

```
# mkdir ignatz
```

5. Type `chown login-name login-name` and press Return.

The user now owns the home directory.

```
# chown ignatz ignatz
```

6. Type `chgrp primary-GID login-name` and press Return.

Assign the home directory to the primary group you specified in the `passwd` file for the user account, for example, the `docia` group.

```
# chgrp docia ignatz
```

7. Type `chmod 755 /export/home-dir/login-name` and press Return.

Set the user's home directory permissions to `rx` for owner, `r-x` for group, and `r-x` for other.

```
# chmod 755 /export/home1/ignatz
```

8. Allow the home directory to be shared by other systems.

The following steps are performed once for each `/export/home-dir` directory reserved for users' home directories. By convention, these are named `/export/home`, `/export/home1`, `/export/home2`, and so on.

a. Type `share` and press Return.

If `/export/home-dir` is listed, it is already being shared, so skip to step 9.

b. Edit the file `/etc/dfs/dfstab` and add the following line:

```
share -F nfs /export/home-dir
```

Whenever the system reboots, the `share` command will run.

c. Type `shareall -F nfs` and press Return.

This command executes all the `share` commands in the `/etc/dfs/dfstab` file, so you do not have to wait for a reboot.

d. Type `ps -ef | grep mountd` and press Return.

If the daemon `mountd` is running, skip to step 9.

e. Type `/etc/init.d/nfs.server start` and press Return.

Start the daemons required for sharing file directories.

Note – If your network is not running a network name service, like NIS+, you need to add the home directory server’s Internet Protocol (IP) address and system name to the `/etc/hosts` file on the user’s system. You can use Database Manager to edit the local `/etc/hosts` file.

9. (Optional) Set up a disk quota for the user in the file system containing his or her home directory.

This step may be warranted in a setting where disk space is limited.

▼ How to Move the Contents of the Home Directory

1. Log in or use `rlogin` to log in remotely to the system where the old home directory resides, and become `root`.

Typically the system is a server remote from the local workstation on which the directory is mounted.

2. Type `cd parent-directory` and press Return.

You want to go to the directory under which the user’s home directory resides. By convention, the parent directory should be `/export/home` or `/export/home1`, and so on.

3. To see if you have permission to access the system (if remote) where the new home directory resides, type `rsh new-system date` and press Return.

If the date is displayed (rather than a message saying permission was denied) go to step 5.

4. Use the `rlogin` command to log in remotely to the new system, add the old system’s name to the `/.rhosts` file, then exit back to the old system.

**5. Type `tar cvf - old-directory | rsh new-system cd parent-directory \;`
`tar xvpf -` and press Return**

All the files in the old directory are copied to the new directory. According to convention, the name of both the old and new directory is the same as the user name. If the new home directory is on the same system as the old home directory, just omit the `rsh system` portion of the command.

6. Type `rm -r old-directory` and press Return.

All the files in the old directory are deleted, and the home directory is deleted.

The directory file for the old home directory is deleted.

```
$ rlogin anthologia
$ su
$ password
# cd /export/home
# rsh infomaniac date
Thu Sep 3 08:49:15 EDT 1992
# tar cvf - tamiro | rsh infomaniac cd /export/home \; tar xvpf -
# rm -r tamiro
```

Copying Initialization Files into a User's Home Directory

Prerequisite

- User's home directory exists
- Root access on the system containing the home directory

Information You Need

- Shell type (C, Bourne, or Korn) assigned to the user

Use this procedure if you do not use User Account Manager to create the home directory for a user account. If you want to move an existing user's home directory, just copy all the files (including the user initialization "dot" files) from the old to the new directory, as described in "How to Move the Contents of the Home Directory" on page 52, rather than copying the initialization files as shown in the following steps. How to Copy Initialization Files into a User's Home Directory

1. Type `cd /export/home-dir/username` and press Return.

You are in the user's home directory.

```
# cd /export/home1/ignatz
```

2. Type `cp shell-init-file-directory/. * .` and press Return.

The user initialization files are copied from the shell-specific directory to the user's home directory. This requires the user initialization files to follow the conventions described in "Setting Up Initialization (Skeleton) Files" on page 19. You will see error messages about the "." and ".." files not being copied, which you can ignore.

```
# cp /etc/skel/C/. * .
```

3. Type `chmod 744 . * ; chown username . *` and press Return.

Permissions are set and the user now owns the initialization files.

```
# chmod 744 . * ; chown ignatz . *
```

4. Type `chgrp primary-GID . *` and press Return.

The files are assigned to the primary group (for example, sysadmin) you specified in the passwd file for the user account.

```
# chgrp sysadmin . *
```

To set up mounting for the home directory:

♦ See "Mounting a User's Home Directory" on page 28.

To customize the user environment:

♦ See "How to Customize a User's Environment" on page 30.

Modifying a User Account

Prerequisites

- Start OpenWindows, if necessary
- Start Administration Tool, if necessary
- Verify required access privileges

See "Required Access Privileges for Setting Up User Accounts" on page 45 for more information.

If you want to change a user's account to reference a different home directory, you should first create the new directory.

Information You Need

- User's login name and user ID (UID)

▼ **How to Modify a User Account**

- 1. Click on the User Account Manager icon.**
The Naming Service window appears.
- 2. Select the name service being used to administer the network.**
- 3. Click on Load.**
The User Account Manager main window appears.
- 4. Select the user entry to be modified.**
- 5. Choose Modify/View User from the Edit menu.**
The Modify User window appears.

6. Change the information in the appropriate fields.

Field	Description of Possible Changes
<i>User Name and User ID</i>	Change either, or both. However, the home directory's ownership is not changed (if the home directory exists). An error message is displayed if you try to change permissions on the home directory. Manually change the ownership of all files and directories, including the mailbox, that have the old UID (on any systems to which they have migrated).
Primary Group	Enter a new name or number. A number is required when the group does not exist yet. The NIS+ cred table entries (if any) are updated.
Secondary Groups	You can add or delete group names from the list. The group file is updated, accordingly.
Comment	You can change the text.
Login Shell	You can change the login shell program.
Account Security	You can change any of the account security fields. If the current password status is Normal Password, change the password by choosing Normal Password from the Password menu.
Home Directory	You can change the home directory name by changing the Path and Server entries. Modify User will not create a new directory or move the contents of an existing home directory. Only the information in the passwd and auto_home files are changed. If the directory indicated by the Path and Server fields exists, the Permissions buttons are activated and you can change the permissions on the directory.
AutoHome Setup	A check mark shows automounting is in effect. By toggling the check mark you can either undo or set up automounting. In the latter case, fill in the Path and Server fields.
Cred. Table Setup	Shown only if NIS+ applies. A blank indicates credentials have not been set up. Check to add them, and, if the account has a normal password, re-enter the password to generate the DES entry.
Mail Server	You can change the Mail Server, but the user's mailbox contents are not automatically moved by User Account Manager.

To retract your changes, click on Reset.

- 7. Click on Apply.**
The user account is modified.

Deleting a User Account

Prerequisites

- Start OpenWindows, if necessary
- Start Administration Tool, if necessary
- Verify required access privileges

See “Required Access Privileges for Setting Up User Accounts” on page 45 for more information.

Information You Need

- User’s login name and user ID (UID)

▼ **How to Delete a User Account**

- 1. Click on the User Account Manager icon.**
The Naming Service window appears.
- 2. Select the name service being used to administer the network.**
- 3. Click on Load.**
The User Account Manager main window appears.
- 4. Click on the entry for the user account you want to delete.**
- 5. Choose Delete User from the Edit menu.**
The Delete User window appears.
- 6. (Optional) Click on the check box to delete the user’s home directory and its contents.**
User Account Manager must be installed on the system where the home directory resides.
- 7. (Optional) Click on the check box to delete the user’s mailbox and its contents.**
User Account Manager must be installed on the system where the mailbox resides.

8. Click on Delete.

The user's entries are removed from the `passwd`, `group`, `aliases`, `cred`, and `auto_home` files. Only the single mail alias that directs mail to the user's mail box is removed; the user name is not deleted from any other mail aliases.

Repeat steps 2 through 7 to delete entries for other users. In addition, if you are not using a name service and want to remove the user account from the other systems, you also have to repeat step 3, and specify a different system (host) name in the `Use /etc files on host` field in the name service window.

▼ How to Disable a User Account

The steps for disabling a user account overlap the steps for modifying a user account.

1. Perform the first three steps in the task “How to Modify a User Account” on page 55.**2. Choose Account Is Locked from the Password menu.**

This selects the locked password status, which you use to disable the user account.

3. Click on Apply.

The account is locked. Actually, an invalid password, `*LK*`, is assigned to the account. This prevents future logins. In addition, if NIS+ is the selected name service and entries have been added to the `cred` table, the DES entry is removed.

You can re-enable logins to the account by changing the password status to `Normal Password` or `Cleared until first login`.

Modifying or Deleting Groups

Prerequisites

- Start OpenWindows, if necessary
- Start Administration Tool, if necessary
- Verify required access privileges

See “Required Access Privileges for Setting Up User Accounts” on page 45 for more information.

▼ How to Modify a Group

- 1. Start Database Manager, select the group file, then select name service, and load the group file.**

Note – If you have a network running only the NIS+ service, the `nogroup` entry is not needed.

- 2. Click on the entry to be modified.**
The entry is highlighted.
- 3. Choose Modify Entry from the Edit menu.**
The Modify Entry window appears, showing the current information in the group file fields for the group.
- 4. Add user names to, or remove user names from, the Members List text field.**
If you make a mistake or change your mind, click on Reset and retype the information.
- 5. Click on Modify.**
The `group` file is modified. Repeat steps 1 through 5 to modify additional groups. If you are not using a name service and want to modify the groups on other systems, you also have to repeat step 1, and specify a different system (host) name in the Use `/etc files on host` field in the Load Database window.
- 6. (Optional) If the network is running the NIS+ service and credential entries have been added to the cred table, type**
`nisaddcred -P username.domainname local` **and press Return.**
This updates the NIS+ `local` credentials for a user. Repeat this step for each user name added to, or removed from, any group’s member list, so his or her credentials will reflect the new group memberships.

▼ How to Delete a Group

The steps for deleting a group are almost the same as the steps for modifying a group.

1. Perform the first four steps from the task for modifying a group (see page 58).

2. Click on the entry to be deleted.
The entry is highlighted.

3. Choose Delete Entry from the Edit menu.

4. Click on Delete.

The group is deleted from the `group` file. Repeat steps 2 through 4 in this task to delete other groups. If you are not using a name service and want to delete the groups on other systems, you also have to repeat step 1 in the section “How to Modify a Group” on page 59, and specify a different system (host) name in the `Use /etc files on host:` field in the Load Database window.

5. (Optional) If the network is running NIS+ and credential entries have been added to the `cred` table,

type `nisaddcred -P username.domainname local` and press Return.

This updates the NIS+ `local` credentials for a user. Repeat this step for each user name that was included in the group’s member list, so his or her credentials will reflect the changed group memberships.

▼ How to Create or Change a Password

Users can create or change their own password at any time. You must be `root` to create the initial password or change the password for any other user. In addition, to create an NIS+ password you must have the appropriate NIS+ privileges and you must have established the necessary network-wide credentials (see `nispasswd(1)`).

You can also use `passwd` to define, change, and view password attributes, like password aging. See `passwd(1)` for more information.

Note – Do not include a colon (`:`) in new passwords.

To create an NIS+ password:

1. **Become root on the NIS+ server.**
2. **Type `nispasswd username` and press Return.**
The prompt `New password:` is displayed.
3. **Type the new password and press Return.**
The prompt `Retype new password:` is displayed.
4. **Retype the password and press Return.**
The password is assigned and added to the NIS+ `passwd` table. You can also use `nispasswd` to define, change, and view password attributes, like password aging. See `nispasswd(1)` for information.

```
[70]saturn % su
Password:
saturn # nispasswd ignatz
New password:
Retype new password:
saturn #
```

To change an NIS+ password:

1. **Become root on the NIS+ server.**
2. **Type `nispasswd username` and press Return.**
The message `Changing NIS+ password for username on NIS+ server is displayed` and the prompt `Old password:` is displayed.
3. **Type the old password and press Return.**
The prompt `New password:` is displayed.
4. **Type the new password and press Return.**
The prompt `Re-enter new password:` is displayed.

5. Retype the password and press Return.

The password is assigned and added to the NIS+ passwd table.

```
[26]mercury }% su
Password:
mercury # nispasswd ignatz
Changing NIS+ password for ignatz on NIS+ server
Old password:
New password:
Re-enter new password:
mercury #
```

You can also use `nispasswd` to define, change, and view password attributes, like password aging. See `nispasswd(1)` for more information.

To create an NIS password:**1. Become root on any system in the NIS domain.****2. Type `yppasswd username` and press Return.**

The message `Changing NIS password for username` and the prompt `New password:` are displayed.

3. Type the new password and press Return.

The prompt `Retype new password:` is displayed.

4. Retype the password and press Return.

The password is assigned and added to the NIS master file. It may take a few minutes for the new password to be set up on all the NIS servers and clients.

```
[70]jupiter % su
Password:
jupiter # yppasswd ignatz
Changing NIS password for ignatz
New password:
Retype new password:
NIS entry changed on eucalyptus
jupiter #
```

To change an NIS password:

- 1. Become root.**
- 2. Type `yppasswd username` and press Return.**
The message Changing NIS password for *username* and the prompt Old yp password: are displayed.
- 3. Type the old password and press Return.**
The prompt New password: is displayed.
- 4. Type the new password and press Return.**
The prompt Retype new password: is displayed.
- 5. Retype the password and press Return.**
The message NIS entry changed on *nis-server* is displayed, and the password is assigned and added to the NIS master file.

```
[26]neptune }% su
Password:
neptune # yppasswd ignatz
Old password:
New password:
Retype new password:
NIS entry changed on eucalyptus
neptune #
```

To create a local password:

- 1. Become root.**
- 2. Type `passwd username` and press Return.**
The prompt New password: is displayed.
- 3. Type the new password and press Return.**
The prompt Re-enter new password: is displayed.

4. Retype the password and press Return.

The password is assigned and added to the `/etc/shadow` file.

```
[70]terra % su
terra # passwd ignatz
New password:
Re-enter new password:
terra #
```

You can also use `passwd` to define, change, and view password attributes, like password aging. See `passwd(1)` for more information.

To change a local password:**1. Become root.****2. Type `passwd username` and press Return.**

The prompt `New password:` is displayed.

3. Type the new password and press Return.

The prompt `Re-enter new password:` is displayed.

4. Retype the password and press Return.

The password is assigned and added to the `/etc/shadow` file.

```
[26]luna }% su
Password:
luna # passwd ignatz
New password:
Re-enter new password:
luna #
```


Part 2— Printing

Part 2 has five chapters and two appendixes:

Chapter 3, “Setting Up Printers,” describes how to set up basic printing services using the Printer Manager, an Administration Tool application.

Chapter 4, “Routine Printer Administration,” describes how to administer printing services after printers are installed and set up.

Chapter 5, “Managing Character Sets, Filters, Forms, and Fonts,” describes additional printer-specific tasks (such as setting up and administering print wheels, filters, forms, and fonts).

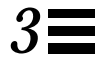
Chapter 6, “Setting Printing Policies,” describes the tasks for setting printing policies, such as whether banner pages print.

Chapter 7, “Troubleshooting Printing Problems,” describes how to troubleshoot printing problems which may occur when setting up or maintaining printing services.

Appendix A, “Setting Up Printer Services Using the Command-Line Interface,” describes how to set up basic printing services using the LP print service’s command-line interface.

Appendix B, “Understanding and Customizing the LP Print Service,” describes what the LP print service does, how it works, and the directory structure of the LP print service commands. It also explains how to adjust printer port characteristics, add entries to the `terminfo` database, write an interface program, and provide new print filters and forms.

Setting Up Printers



This chapter describes how to set up local printers, set up access to remote printers, and perform some printer administration tasks using Printer Manager, an Administration Tool application.

If you want to skip the background information that explains the concepts of printer setup, and proceed directly to step-by-step instructions, use the following table to find the page where the instructions for a specific task begin.

<i>How to Add a Printer</i>	<i>page 87</i>
<i>How to Set Up a Client</i>	<i>page 91</i>
<i>How to Modify Printer Information</i>	<i>page 93</i>
<i>How to Delete a Printer</i>	<i>page 94</i>
<i>How to View Available Printers</i>	<i>page 95</i>

If you want to review background information first, read the following section, “About Setting Up Printing Services” on page 68.

If you need reference material related to the instructions, the section “Reference Information for Setting Up Printing Services” on page 95 contains additional information.

If you want to set up printing services using the LP print service’s command-line interface, instead of Printer Manager, see Appendix A, “Setting Up Printer Services Using the Command-Line Interface,” on page 275.

You can refer to the following chapters and appendixes in this manual for additional information about printing:

- See Chapter 4, “Routine Printer Administration,” for information about how to administer printing services after the printers have been set up.
- See Chapter 5, “Managing Character Sets, Filters, Forms, and Fonts,” for tasks specific to different types of printers.
- See Chapter 6, “Setting Printing Policies,” for tasks to set printing policy, such as whether banner pages are required.
- See Chapter 7, “Troubleshooting Printing Problems,” if you have trouble setting up or administering printing.
- See Appendix A, “Setting Up Printer Services Using the Command-Line Interface,” for information about setting up printing services using the LP print service’s command-line interface.
- See Appendix B, “Understanding and Customizing the LP Print Service,” for a description of the LP print service and how to customize various aspects of it.

About Setting Up Printing Services

The instructions in this chapter describe how to use Printer Manager, a graphical user interface for setting up print services:

Most of your needs for setting up printing services—adding printers to servers or adding access from print clients to remote printers on print servers—should be met by Printer Manager. If you have special needs, like writing scripts, you may want to use the LP print service commands (which underlie Printer Manager) directly.

The background information presented before the instructions—except the overview of Printer Manager—applies whether you use Printer Manager or the LP print service command-line interface.

Before Using Administration Tool

The following hardware and software requirements must be met before you use Administration Tool:

- SunOS 5.x Printer Manager software on SunOS 5.x print servers and print clients.
- A bit-mapped display monitor. Administration Tool's applications can be used only on a system where the console is a bit-mapped screen.
- OpenWindows software. Start OpenWindows, if necessary, by typing `/usr/openwin/bin/openwin` and pressing return.

Required Privileges for Setting Up Printers

Table 3-1 describes the required privileges for setting up printers.

Table 3-1 Required Privileges for Printer Setup

To Set Up Printers Using the	The Required Access Privileges Are:
<code>/etc</code> files	Root access or membership in the <code>sysadmin</code> group (GID=14) on the local or remote system.
NIS+ printers table	<ul style="list-style-type: none"> • Membership in the <code>sysadmin</code> group (GID=14) in the NIS+ group table. • Read permissions on the NIS+ printers table (see <i>Name Services Administration Guide</i>) to be able to use the Show NIS+ Printers option. • Create and destroy permissions on the <code>printers</code> table to be able to use the Register with NIS+ option. NIS+ permissions are granted by membership in an NIS+ group.

Using Administration Tool

Start Administration Tool from an OpenWindows window as follows.

```
$ admintool &
```

Types of Print Clients and Print Servers

Most likely, your overall goal for setting up printers on a network is to give users on different systems access to one or more printers. Your network may include many systems, some of which may be running SunOS 5.x system software, while others may be running SunOS 4.1 software.

You need to decide which systems will have *local printers* directly cabled to them, and which systems will connect to printers over the network. The system that has the printer connected to it and makes the printer available to other systems is called a *print server*. The system that has its printing needs met by a print server is called a *print client*.

The following figure illustrates the different types of print client and print server combinations.

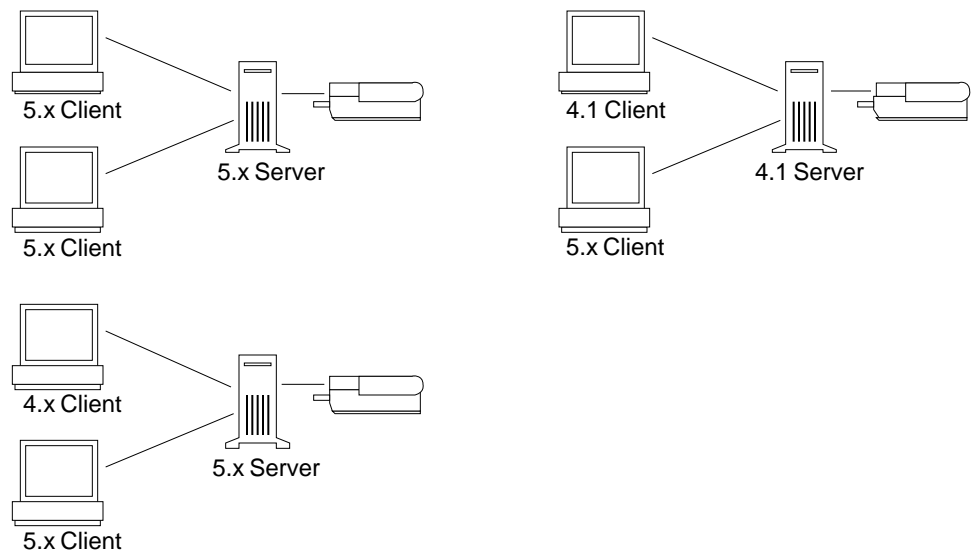


Figure 3-1 Print Client and Server Configurations

You can have the following client-server combinations:

- SunOS 5.x print clients with a SunOS 5.x print server
- SunOS 5.x and SunOS 4.1 print clients with a SunOS 5.x print server
- SunOS 5.x and SunOS 4.1 print clients with a SunOS 4.1 print server

This chapter focuses on setting up printing services using Printer Manager, an Administration Tool application, which interfaces with the SunOS 5.x LP print service. However, print clients installed with SunOS 5.x system software can take advantage of existing print servers running SunOS 4.1 system software, and print clients installed with SunOS 4.1 system software can become clients of print servers running SunOS 5.x.

The procedures—using Printer Manager—for setting up print servers or clients running SunOS 5.x system software are described in this chapter. The procedures for setting up print servers or clients running SunOS 4.1 system software are summarized in Appendix A. SunOS 4.1 print servers and clients are fully described in the SunOS 4.1 documentation.

Using Printer Manager

Printer Manager is a Administration Tool application that enables you to set up print servers and print clients.

Printer Manager has three top-level menus:

- *View* – Show and find printers
- *Edit* – Add, modify, and delete printers
- *Goto* – Change system on which to view or edit printers

Under the Edit menu, the Add Printer option has two main functions:

- Adding a local printer
- Adding access to a remote printer

The following figure illustrates Printer Manager's graphical user interface.

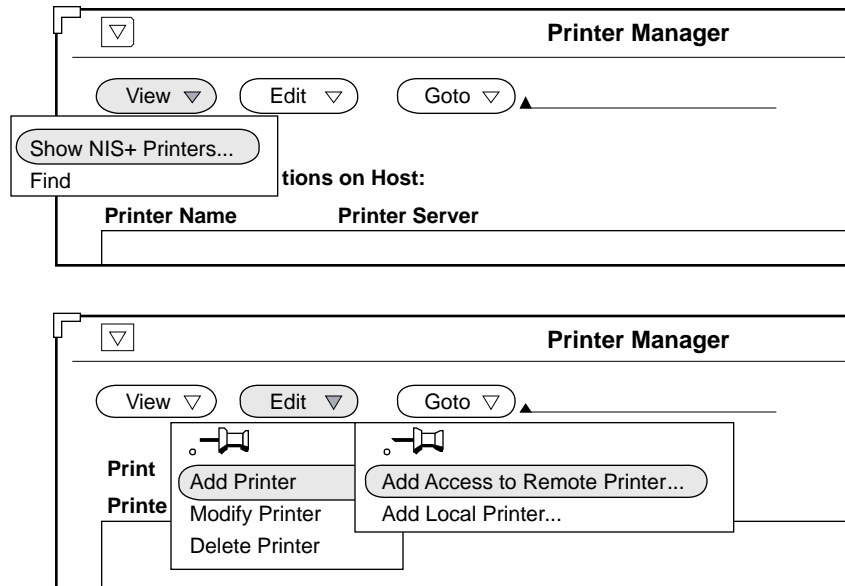


Figure 3-2 Printer Manager's Graphical User Interface

The Add Local Printer option (under Add Printer) lets you define the characteristics of a printer on the local system to which it is (or will be) connected. In addition, Add Local Printer enables the system to which the printer is connected to function as a print server on the network.

The Add Access to Remote Printer option (under Add Printer) lets you set up a system as a print client of a print server system.

You can also use Printer Manager to modify or delete the information that defines a printer as local or remote to a specific system.

Because Printer Manager is a distributed application, you can use it from one system to add, modify, or delete printers on other systems. All systems to be administered this way must have Printer Manager installed. Consequently, you cannot use Printer Manager to set up a system that has SunOS 4.1 software or has a non-SunOS version of SVR4 software as a print server or print client. However, you can set up a system that has Printer Manager installed as a print client of an existing SunOS 4.1 print server.

Distributed changes require you to run Administration Tool under your own user name, not as root. In addition, you must be a member of the sysadmin group (GID 14) on every system you modify. However, you can add access to a remote printer from your local system without having to be in the sysadmin group in effect on the print server.

You can perform local changes to your own system, running Administration Tool as local root. In this case, you do not need to be a member of the sysadmin group. In fact, you should not create or change the sysadmin group in your local `/etc/group` file, unless you intentionally want to allow or disallow others (network-wide administrators) from administering your system using Administration Tool.

The Goto option allows you to specify the target system, from whose perspective the available local and remote printers are listed and to which local or remote printers can be added. Printer Manager automatically reaches across the network and changes configuration information on the target system.

Printer Manager also helps you track of all the printers available across the network in various domains if you have NIS+ software. As you add printers, you can build an NIS+ table listing the printers. Then when you want to see which printers are available, you can use the Show NIS+ Printers option. (You must run Administration Tool under your own user name, not as local root.)

Network Prerequisites for Setting Up Printing

To set up print servers and clients, you must have a network that enables access among systems. If your network is running NIS+ software, see *Name Services Administration Guide* for instructions to enable access between systems. If your network is not running NIS or NIS+, make sure the Internet address and system name for each client system is included in the `/etc/hosts` file on the print server. Also, the Internet address and system name for the print server must be included in the `/etc/hosts` file of each print client system.

In addition, to use Printer Manager to set up SunOS 5.x print servers and clients, Printer Manager software must reside on both the servers and clients.

Local Printer Setup

You generally perform the following tasks to set up a local printer:

- Physically connect the printer to the system.
- Set the printer switches and/or configure baud rate, port, and other settings.
- Plug the printer into a power outlet.
- Define the characteristics of the printer to the LP print service (using Printer Manager).

Consult the printer vendor's documentation and the installation documentation for your system for information you may need to perform the first three tasks. The last task is described in this chapter.

Print Server Setup

To set up a print server, use the Add Local Printer option in Printer Manager. (The system to which the local printer is attached becomes the print server.)

When you set up a print server, you define the printer characteristics to the LP print service. On a print server, Printer Manager then does the following automatically:

- Configures the port monitor
- Registers the network listen service
- Identifies the print clients

See Appendix A, "Setting Up Printer Services Using the Command-Line Interface," for instructions to do Printer Manager functions manually.

Print Client Setup

On the print client, Printer Manager completes the following tasks so the print client can use the printer connected to the print server:

- Identifies the printer and server system to which the printer is connected (with the information you supply).
- Defines the characteristics of the printer.

- Configures the port monitor and registers the listen services with the port monitor.

Print Server Configurations

As a network administrator, you must decide how to allocate printers, and whether the printers are dedicated to one system or available to many systems and users. The decisions you make determine how you will administer printing at your site.

When you establish printing services for a network, you can “try out” printer configurations by changing the configuration until it best meets your users’ needs. You can move printers around on a network with a minimum of effort.

There are several possible arrangements for printers on a network. If your site has a network with a printer connected to every system, it has a *distributed printing configuration*, as shown in Figure 3-3.

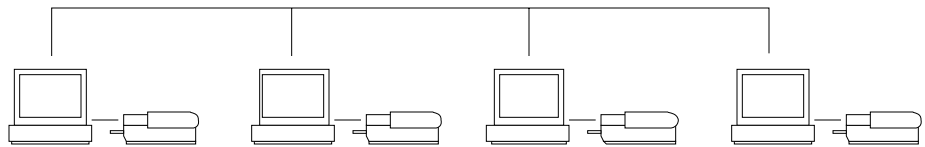


Figure 3-3 Distributed Printing Configuration

Even with this configuration, print resources can be shared over the network, and there often are several reasons to do so. For instance, one printer may be faster than the others, and more suitable for printing drafts than final copies. One user may need to print many documents quickly, while other users may not have such printing demands. If a printer is down, the user of the system to which the printer is connected can access another printer while the local printer is being fixed.

In a *centralized print server configuration*, all printers are in one physical location, like a computer center, and are connected to one print server system, as shown in Figure 3-4. Users of the print client systems who want to use a printer access it over the network.

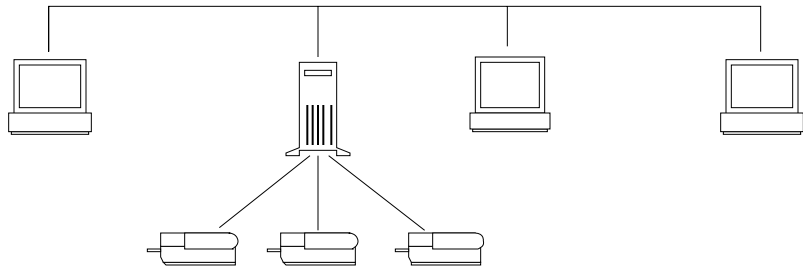


Figure 3-4 Centralized Print Server Configuration

Note – Most systems have two serial ports, and possibly a parallel port. Unless you add ports, you cannot connect more than two serial printers and a parallel printer to one system.

In a third arrangement, a *combined distributed and print server configuration*, some print server systems are connected to several printers, and other systems are connected to their own printers, as shown in Figure 3-5.

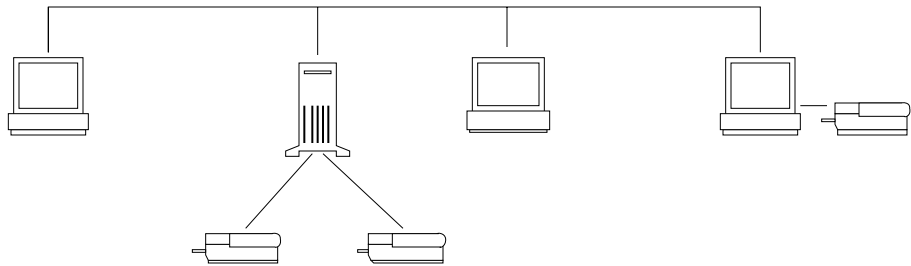


Figure 3-5 Print Server and Distributed Printing Configuration

System Requirements for a Print Server

You can attach a printer to a standalone system, or to any system on the network. Any networked system with a printer can be a print server, as long as the system has adequate resources to manage the printing load.

Each print server should have the following system resources:

- Spooling directory space of 8 Mbytes (or more)
- Hard disk (strongly recommended but not required)
- Memory of 16 Mbytes (or more)
- Swap space of 20 to 24 Mbytes (or more)

Spooling Space

Spooling space is the amount of disk space that is used to store and process requests in the print queue. Spooling space is the single most important factor to consider when deciding which systems to designate as print servers. When users submit files for printing, they are stored in the `/var/spool/lp` directory until they have been printed. The size of the `/var` directory depends on the size of the disk and how the disk is partitioned. Spooling space may be allocated in the `/var` directory on the print server hard disk, or mounted from a file server and accessed over the network.

Note – If `/var` is not created as a separate partition, the `/var` directory uses space in the root partition, which is likely to be quite small.

When evaluating systems as possible print servers, consider the available disk space. A large spool directory could consume 600 Mbytes of disk space. Look at the size and partitioning of the disks available on systems that could be designated as print servers.

Also carefully evaluate the printing needs and usage patterns of the print client systems. If users in a small group typically print only short email messages—simple ASCII files without sophisticated formatting requirements—a print server with 20 to 25 Mbytes of disk space allocated to `/var` is probably sufficient. If, however, many print client users are printing lengthy PostScript™ files, they probably will fill up the spooling space quite frequently. When users cannot queue their jobs for printing, work flow is interrupted. Requests for more spooling space may force the system administrator to either add disk space for spooling, or to designate a different system as the print server for the group.

Table 3-2 provides some common disk configuration information and recommendations for the number of average users the configuration can serve.

Table 3-2 Typical Disk Configuration Information

Disk Size	/var Partition (Mbytes)	Spooling Space (Mbytes)	Number of Users
104 Mbytes	8	4	1-3
207 Mbytes	16	12	1-16
424 Mbytes	212	206	1-32
669 Mbytes	335	328	1-64
991 Mbytes	500	490	1-64 or more
1.36 Gbytes	335	206	1-32

If the print server has a /var directory that resides in a small partition, and if a large amount of disk space is available elsewhere, you can use that space as spooling space by mounting it on the /var directory on the print server. See the *File System Administration* for information about mounting file systems and editing the `vfstab` file.

Hard Disk

The print server should have a hard disk and a local /var directory. You can, however, mount spooling space for a print server from a *file server* (a system that shares its file systems with other systems) that is accessed over the network. In that case, the file system must be exported with Write permission granted to user `lp`. Network traffic, however, can slow printing. If a print server has its own hard disk and a local /var directory, printing is much faster, and you can more accurately predict the time needed to process print requests.

Memory

A minimum of 12 Mbytes of memory is required for the print server. More memory usually means quicker overall system response.

Swap Space

The swap space allocation on the print server should be sufficient to handle LP print service requirements. See the *File System Administration* for information about how to increase swap space.

Printer Configuration Information

This section describes the configuration information you give to Printer Manager to set up a printer on the network:

- The serial or parallel device name (required)
- A unique name for the printer (required)
- The printer type (required)
- The file content, which is the type of file that can be printed (required)
- The description of the printer to convey to users (recommended, optional)
- The default printer for each system (recommended, optional)

Configuration information is stored in the LP configuration files in the `/etc/lp` directory. See “LP Configuration Files” on page 309 for information.

Printer Device Name

You must indicate the *printer device name* for the port to which the printer is connected. Printer Manager automatically displays a list of the ports on the system from which to select. The LP print service initializes the printer port using the settings from the standard printer interface program. See “Filtering Print Files” on page 305 for more information about printer interface programs. If you have a parallel printer or a serial printer for which the default settings do not work, see “Adjusting Printer Port Characteristics” on page 327 for information about how to customize the port settings.

Printer Name

When adding a printer to a system, you choose a *printer name* for the printer. A printer name must be unique among all printers known to the system, and can contain a maximum of 14 alphanumeric characters, which may include dashes and underscores. When administering printers in a complex network, keep printer names unique within the bounds of the administrative domain.

You should also establish conventions when naming printers. Choose printer names that are meaningful and easy to remember. A printer name can identify the type of printer, its location, or the print server name. Establish a naming convention that works for your site. For example, if you have different types of printers on the network, including the printer type as part of the printer name can help users choose an appropriate printer. You could identify, for instance, PostScript printers with the letters `PS`. If, however, all of the printers at your site are PostScript printers, you would not need to include the initials `PS` as part of the printer name.

You use printer names to:

- Add the printer to the LP print service
- Change the configuration of the printer
- Monitor the print queue
- Check the status of the printer
- Accept or cancel print requests for the printer
- Enable or disable the printer
- Specify a default printer
- Submit a print job to a particular printer

Printer Type

A *printer type* is the “generic name” for a type of printer. By convention, it is often derived from the manufacturer’s model name. For example, the printer type name for the DECwriter™ printer is `decwriter`. However, the common printer type, `PS`, does not follow this convention. `PS` is used as the printer type name for many different models of PostScript printers, like LaserWriter® I and LaserWriterII printers.

Printer Manager requires you to select the printer type from a list of available printer types. For a local PostScript printer, use either `Postscript (PS)` or `Postscript Reverse (PSR)`. If your PostScript printer prints pages face up, documents appear to be printed backwards—the first page ends up on the bottom and the last page ends up on the top. In this case, specifying the printer’s type as `PSR` causes the LP print service to reverse the order of the pages before sending them to the printer, so the last page is printed first. Of course, the intention is to get the pages printed in forward order. However, the LP print service can reliably change the page order only for PostScript files that

conform to the standards in Appendix C of the *PostScript Language Reference Manual* (written by Adobe Systems Incorporated; published by Addison-Wesley, 1990).

If the printer type appears in the list, but also supports PostScript, choose Postscript (`PS` or `PSR`) to designate that printer.

If a printer can emulate more than one kind of printer, you can assign it several types by using the `lpadmin -T` command. If you specify more than one printer type, the LP print service uses the type that is appropriate for each print request.

Information about each printer type is stored in the `terminfo` database (`/usr/share/lib/terminfo`). This information includes the printer capabilities and initialization control data. See “Printer Entries in the `terminfo` Database” on page 84 for more information about the `terminfo` database.

For local printers, Printer Manager uses the printer type to access data in the `terminfo` database. For remote printers, Printer Manager uses the default printer type of `unknown`.

You can view or change the type assigned to the printer by choosing the Edit option Modify Printer in Printer Manager. Or, you can use the `lpadmin -T` command to change the printer type. If you omit the `-T` option, the printer has the default printer type of `unknown` and does not get initialized properly before it prints a file.

File Content Type

The *file content type* tells the LP print service the types of files—in addition to those that match the specified printer type—whose contents can be printed directly on a printer; filtering is done for other types of files. In addition, direct printing requires the necessary fonts to be available in the printer.

Many printers can print two types of files directly:

- The same type as the printer type (for example, `PS` for a PostScript printer)
- The type `simple` (an ASCII file)

Some printers can handle several types of files, in which case you should specify those file content types in Printer Manager.

Content type names should be meaningful to you and the users you support. Your users need to know these names to use them consistently in making print requests. They consist of up to 14 letters, numerals, and underscores. In addition, they must match the input and output names used with filters.

When submitting a file for printing, the user indicates the content type of the file (`lp -T type`). Otherwise, a file is assumed to be `simple` (ASCII text). The LP print service uses the file content type to match jobs to printers—to look for a suitable printer that can print the file directly or to determine which filters to use to convert the file contents into a type the printer can handle.

Some common file content types for local printers are `postscript` and `simple`.

Printer Manager provides you with a list of file content types for a local printer from which to select. The choices are translated to the names the LP print service uses. Table 3-3 describes the underlying `lpadmin -I` command for each choice and the implications of each choice.

Table 3-3 Choosing File Content Types With Printer Manager

File Contents Choice	Underlying <code>lpadmin</code> Option	Description
PostScript	<code>-I postscript</code>	PostScript files (<code>lp -T postscript</code>) are not filtered. ASCII files (<code>lp -T simple</code>) are filtered.
ASCII	<code>-I simple</code>	PostScript files (<code>lp -T postscript</code>) are filtered. ASCII files (<code>lp -T simple</code>) are not filtered.
Both PostScript and ASCII	<code>-I simple,postscript</code>	Neither PostScript files (<code>lp -T postscript</code>) nor ASCII files (<code>lp -T simple</code>) are filtered.
None	<code>-I ""</code>	All files are filtered, except those matching the printer's type (<code>lp -T printer-type</code>).
Any	<code>-I any</code>	No filtering is done. If a user sends a file type to the printer that the printer cannot handle directly, the file will not be printed.

You can choose `any` for a local printer, but be careful. It means no filtering will be done. If a user sends a file type to the printer that the printer cannot handle directly, the file will not be printed. The safest choice may be `None`, which forces the LP print service to find filters for most print requests. If you are sure a printer can directly print a file content type, you may want to specify that specific file content type when you print a file to bypass unnecessary filtering. However, you should read the printer's documentation or try experimenting with the printer to determine whether given types of files need filtering, or not.

Printer Manager's default choice `Postscript` (which means filtering is not needed for PostScript files) is probably correct most of the time.

Note - If you specify more than one printer type for a printer that can emulate other printers, you must specify `simple` as the content type.

You should use a file content type of `any` for a SunOS 5.x print client. Printer Manager automatically assigns `any`, which causes files to be filtered on the print server, rather than on the print client. If you do not want that, you can use the `lpadmin -I` command directly. Or, after adding the printer, you can use Printer Manager's Modify Printer option to change the file content type(s). If you specify file content types other than `any`, the filtering is done on the client. You may want to do that for content type `simple` or `postscript`. These types can be used in combination.

Note - The name `simple` means ASCII file, and `any` means any file content type. Be sure to use them consistently. The name `terminfo` is reserved as a reference to all types of printers.

Printer Description (Optional)

You can define a *printer description* for a printer. The description should contain helpful information for users. For example, the description could say where the printer is located, or whom to call when the printer has problems.

Users can display the printer's description by using Printer Manager's Modify Printer option or by using the command:

```
% lpstat -D -p printer-name
```

Default Printer (Optional)

You can specify a *default printer* for each system, even if it is the only printer connected to the system. When you specify a default printer, users do not need to type the default printer name when they use LP print service commands. However, they can override the default by explicitly naming a printer or setting the `LPDEST` environment variable. Before you can designate a default printer, it must be known to the LP print service on the system.

Printer Entries in the `terminfo` Database

The printer you install must correspond to an entry in the `terminfo` database. (You can also create a printer entry in the `terminfo` database. See Appendix B, “Understanding and Customizing the LP Print Service,” for information about adding a new entry to the `terminfo` database.)

This section explains the structure and organization of the `terminfo` database directory, which will help you find out if the `terminfo` database contains an entry for your printer.

Directory Structure of the `terminfo` Database

The default `terminfo` database has 42 subdirectories in the `/usr/share/lib/terminfo` directory.

```

ganymede % pwd
/usr/share/lib/terminfo
ganymede % ls
1  4  7  A  M  a  d  g  j  m  p  s  u  x
2  5  8  B  P  b  e  h  k  n  q  t  v  y
3  6  9  H  S  c  f  i  l  o  r  ti  w  z
ganymede %

```

Each of these subdirectories contains compiled database entries for terminals or printers.

The entries are organized by the first letter of the printer or terminal type. For example, if you have an Epson® printer, look in `/usr/share/lib/terminfo/e` to find your particular model of Epson printer.

```
ganymede % cd /usr/share/lib/terminfo/e
ganymede % ls
emots                ep2500+high        ep48                ergo4000
exidy2500
env230               ep2500+low        epson2500          esprit
envision230         ep40               epson2500-80      ethernet
ep2500+basic        ep4000            epson2500-hi      ex3000
ep2500+color        ep4080            epson2500-hi80    exidy
ganymede %
```

As you can see, there are four entries for Epson printers.

If an entry matches the model of the Epson printer you have, use the entry as the printer type for the `lpadmin -T` command, for example, `lpadmin -p printer-name -T epson2500`.

If you have a NEC® printer, look in the `/usr/share/lib/terminfo/n` directory for your NEC printer model.

```
ganymede % cd /usr/share/lib/terminfo/n
ganymede % ls
ncr7900              ncr7901           netty-Tabs         newhpkeyboard
ncr7900-na           nec               netty-vi           nuc
ncr7900i            net               network           nuclterm
ncr7900i-na         netronics         netx
ncr7900iv           netty             newhp
ganymede %
```

There is only one entry in this directory for NEC. Try using `nec` as the printer type for the `lpadmin -T` command, for example, `lpadmin -p printer-name -T nec`.

You may not find the printer type in the appropriate `terminfo` directory. The type of printer is important to the LP print service, but the type may not necessarily be linked to the manufacturer's name on the printer. For example,

for any type of PostScript printer, you can use the `PS` entry (found in the `/usr/share/lib/terminfo/P` directory) instead of an entry specific to different manufacturer or product names.

If you have an unusual type of printer, you may need to try different entries before you can determine whether a particular `terminfo` entry works for your specific model of printer. If possible, find an entry in the `terminfo` database that works for your printer. It will be much easier than trying to create an entry. If you have to create your own entry, Appendix B, “Understanding and Customizing the LP Print Service,” contains some useful tips.

Instructions for Setting Up Printing Using Printer Manager

This section contains instructions for using Printer Manager to:

- Add local printers
- Add access to remote printers
- Modify or delete printers
- View available printers

Printer Manager lets you set policy related to:

- Fault recovery
- User access restriction
- Banner page printing

See Chapter 5, “Managing Character Sets, Filters, Forms, and Fonts,” for additional setup tasks required for:

- Character sets and print wheels
- Filters
- Form
- Fonts

See Chapter 6, “Setting Printing Policies,” for additional instructions on how to set printing policy and change defaults related to:

- Fault recovery
- User access restriction
- Inclusion of banner page
- Class membership

Adding a Printer

Prerequisites

- Start OpenWindows
- Start Administration Tool
- Verify required access privileges

See “Before Using Administration Tool” on page 69 for more information.

Information You Need

- Printer name
- Print server name
- Device name of port to which printer is (or will be) connected
- Printer type and file content types

Instructions

You can use Printer Manager to add a printer to a system, set up the system as a print server, and (optionally) register the printer with a network-wide NIS+ table.

▼ How to Add a Printer

1. (Optional) Connect the printer to the system and turn on the power to the printer.

You can perform this step before or after using Printer Manager to add a printer. See the printer vendor’s manual and your system’s installation manual for information about switch settings and cabling requirements. You usually connect printer cables to a serial port, but in some cases, depending on the requirements of the printer involved, you can use a parallel port.

2. Start Printer Manager by clicking on the Printer Manager icon.

The Printer Manager window is displayed, listing the printers available to the current system (the system from which Printer Manager is run).

- 3. If the system to which the printer is being added (attached) is not the current system, type the system name in the Goto field and click on the Goto button.**

The selected system is indicated in the Print Configurations on Host field and the list of available printers (if any) is updated to reflect the selected system.

- 4. Select Add Local Printer from the Edit menu.**

The Local Printer window is displayed. The Print Server field is filled in and cannot be changed. It reflects the system to which the printer is being added, the system to which the printer is local. This system may be remote from the system on which you are running Printer Manager.

- 5. Type the printer name in the corresponding field.**

You should choose a name unique to the network, or at least your administrative domain within the network.

- 6. (Optional) Type a comment in the corresponding field.**

You may attach a description giving the user helpful information, like where the printer is located. The *comment* is displayed as part of the printer status.

- 7. Choose the desired port from the Printer Port menu.**

Printer Manager sets the ownership and permissions on the device file to grant access to the special user `lp`. If you have a parallel printer, look for a device file like `bpp0` in the list. For cases not listed, choose Other and type the device name in the pop-up window that is displayed.

- 8. Choose the desired printer type from the Printer Type menu.**

Your choice is mapped to the name of an entry in the `terminfo` database, which is used to initialize the printer.

If your printer type is not included, choose Other and type the name (matching a `terminfo` entry) in the pop-up window that appears. The printer type also implies a file content type that can be directly printed on the printer without filtering.

Note – If you specify a printer type of Postscript (PS) and find that pages print in reverse order, use the Delete Printer option under the Edit menu to delete the printer and then add the printer again, changing the Printer Type to PostScript Reverse (PSR).

9. Choose the desired file contents from the File Contents menu.

The file content(s) you specify in this step are in addition to the file contents type matching the printer type. When a user submits a file for printing (`lp -T type`), the `lp` command looks at the file content type(s) defined for available printers to try to find a printer that can print the file without filtering. Many printers can directly print two types of files: simple ASCII files and files whose type matches the printer type.

The File Contents choices have the following meanings:

- Postscript – Files submitted by `lp -T postscript` go directly to the printer without filtering; other types (for example, `lp -T simple`) are filtered.
- ASCII – Files submitted by `lp -T simple` go directly to the printer without filtering; other types (for example, `lp -T postscript`) are filtered.
- Both Postscript and ASCII – Files submitted by `lp -T postscript` or `lp -T simple` go directly to the printer without filtering.
- None – All files will be filtered to convert them from the type specified by `lp -T` to the printer type.
- Any – All files go directly to the printer without any filtering.

10. Choose how you want to be notified of printer fault messages.

- Write to superuser – Fault messages are displayed on root's terminal (default).
- Mail to superuser – Fault messages are sent as email to the root user's mailbox.
- None – No fault messages are delivered.

11. If you want the printer to be the default printer for the system to which it is being added, click on Yes in the System Default field.

The Yes choice will be turned on, unless a default printer has already been specified. Of all the printers available to the system, the printer designated as the default is the one used for printing when users do not specify a destination printer with the `lp` command.

12. If you do not want banner pages to be required, click on Not Required in the Print Banner field.

This does not turn on and off printing of banner pages. It turns on or off the ability of users to tell the `lp` command not to print a banner page; otherwise, banner pages are always printed.

13. (Optional) If you want the printer to be registered with NIS+, click on Yes in the Register with NIS+ field.

This option is only available when NIS+ is running on the network. As you add printers, you should register them with NIS+. In doing so, you will create an NIS+ table of all the printers (except those on SunOS 4.1 print servers) available on the network; otherwise, you can only see the printers available to the system currently selected (by Goto).

You can then use the Show NIS+ Printers option under the View menu to see the system to which a printer is attached. If you want to add an existing local printer to the table, you can use the Modify Printer option under the Edit menu. Another way to add entries to the table is to use the `nistbladm` command directly. For example, to add a printer named `luna` on server `terra` with the comment `final drafts`, you should enter:

```
# nistbladm -a printer_name=luna printer_host=terra \  
description="Tech pubs final drafts" printers.org_dir
```

Note – The NIS+ Printers table is created by Printer Manager the first time a printer is registered. The ownership and permissions on the table are set to match those established for the standard NIS+ tables, like `hosts`. If you do not have the Create, Modify, or Destroy permissions to make changes to the NIS+ tables, you will not be able to register printers. For information about NIS+ security, see *Name Services Administration Guide*.

14. (Optional) If you want only a few users to have access to the printer, type the (login) names one at a time in the User Access List field and click on Edit after each user name.

Repeat this step for each user. The names are listed in the pane. All users not in the list are denied access to the printer. If you do not create a list, all users have access to the printer. If you make a mistake and want to delete an entry from the list, highlight the entry and use the Delete option from the Edit menu to the right of the User Access List field.

15. Click on Add.

The information defined in the Add Local Printer window is recorded in the `/etc/lp` configuration files on the selected system. If you make a mistake and want to re-enter information, click on Reset to reset all the fields to their original values.

Setting Up a Client

Prerequisites

- Start OpenWindows
- Start Administration Tool
- Verify required access privileges

See “Before Using Administration Tool” on page 69 for more information.

Information You Need

- Printer name
- Print server name
- Type of system software running on the print server (BSD versus SVR4)

Instructions

You can use Printer Manager to add a print client to an existing print server.

▼ How to Set Up a Client

- 1. If the system that you want to add as a client to a print server is not the current system (the one shown in the Printer Configurations on Host field in Printer Manager window), type the client system name in the Goto field and click on the Goto button. Otherwise, go to step 2.**

The list of available printers changes to reflect the selected system.

The Goto button keeps a list of system names that includes the system on which Printer Manager is running. Systems previously specified with the Goto option are also listed. You may choose from the listed systems.

- 2. Click on Edit to bring up the Access to Remote Printer window.**

The Access to Remote Printer window is displayed.

- 3. In the Printer Name field, type the name of the remote printer for which you want the selected system to be a print client.**

The name used to refer to the printer on the print client is the same as the name used on the printer server. If you want the print client to use a different name, you can set it up directly with the `lpadmin` command. See “How to Set Up SunOS 5.x Print Clients” on page 292.

4. In the Printer Server field, type the name of the server system (the system to which the remote printer is attached).

The printer is local to that server system.

5. In the Comment field, type a description of the printer.

Try to include information that will help the users of the printer.

6. Select the Print Server OS.

Print Server OS is actually a misnomer because the option actually indicates the print server's printing protocol—BSD or System V. If the print server is running SunOS 5.x system software (or some other variation of System V), click on System V in the Print Server OS field.

The default setting is BSD, which should be used for print servers running SunOS 4.x system software.

7. Designate whether you want the printer to be the default printer for the print client system.

Note – The Access to Remote Printer window does not provide any fields for specifying the printer type and file contents. These are set to `unknown` and `any`, respectively. To change the file contents type for the printer, you could use the Modify Printer option under Edit.

8. Click on Add.

The information defined in this window is recorded in the `/etc/lp` configuration files on the selected print client system and on the specified print server system. If you make a mistake and want to re-enter information, click on Reset to reset all the fields to their original values.

9. (Optional) If you want to add access to additional remote printers (define this system as a print client for other print servers), repeat steps 3 to 8.

You should select a client system with the Goto option and add access to all the remote printers desired for that system, before selecting another client system and adding access to remote printers.

Modifying or Deleting Printer Information

Prerequisites

- Start OpenWindows
- Start Administration Tool
- Verify required access privileges

See “Before Using Administration Tool” on page 69 for more information.

Information You Need

- Printer name
- Print server or print client name

Instructions

You can use Printer Manager to modify or delete the information for local and remote printers.

▼ How to Modify Printer Information

- 1. If the current system is not the one you want to work with, use the Goto menu button to select the system for which printer information will be modified or deleted. Otherwise, go to step 2.**

The printer can be local or remote to the system. In either case, you can only change the information about a printer on one system at a time.

- 2. From the list of available printers in Printer Manager window, click on the entry for the desired printer.**

Note – You must select from the list displayed in the Printer Manager window, not from the list of printers displayed by the Show NIS+ Printers option.

- 3. If you want to modify (or just view) the information for a printer, do the following:**

- a. Choose Modify Printer from the Edit menu.**

You can also bring up the Modify Printer window by double-clicking on an entry in Printer Manager window.

The Modify Printer window is displayed.

b. Change the contents of the fields as allowed.

You can change any of the fields, except Printer Name, Printer Server, and Printer Type. To change the name of the printer, the name of the printer server, or the model of the printer, you have to delete and add the printer or print client again.

When modifying a local printer, if there is no user access list and you want to specify one, delete the `all` entry before adding user names.

The Enable Print Queue and Accept Print Jobs fields are not available when you add a printer. They help you administer existing printers. For example, you may want to stop printing while a printer is being repaired.

▼ How to Delete a Printer

- ◆ **Choose Delete Printer from the Edit menu, and confirm this choice by pressing the Delete button in the window that is displayed.**

If you delete a printer from a print server system that has several clients, you must also delete the printer from each client by using the Goto option to “go to” the print client and repeating this step.

Viewing Available Printers***Prerequisites***

- Start OpenWindows
- Start Administration Tool
- Verify required access privileges

See “Before Using Administration Tool” on page 69 for more information.

Information You Need

- Printer name
- Print server or print client name

▼ How to View Available Printers

1. Click on the Printer Manager icon.

The Printer Manager window is displayed.

2. (Optional) Click on View.

This selects the Show NIS+ Printers option, which is only available when NIS+ is running on your network. The contents of an NIS+ Printers table are displayed for the current domain. Entries are made in the table when you use the Register with NIS+ option in the Add Printer or Modify Printer window. You can change the domain by typing the name in the NIS+ Domain Name field and clicking on Apply.

3. Use the Goto menu button in the Printer Manager window to specify the system for which you want to view the available printers. Type the system name in the Goto field.

Printers can be local or remote to the system. In either case, you can only view the information about a printer from the perspective of one system (client or server) at a time.

4. Double-click on the desired entry.

This brings up the Modify Printer window, which shows you the information defined for the printer.

Reference Information for Setting Up Printing Services

This section contains reference information you need to perform some of the tasks in “Instructions for Setting Up Printing Using Printer Manager” on page 86.

Frequently Used Printers

This section provides information about printer type and file content type for specific printer models. When you configure a printer, you must identify the manufacturer and model of the printer in a way that the LP print service understands: the printer type. You also must identify the content type for each printer. The content type tells the LP print service what types of files can be printed by each printer.

This section provides the printer type and content type for the printers most commonly used with SunOS 5.x system software. Although not shown, many of these printers can also directly print files with `simple` content type.

If you have a PostScript printer, use a Printer Type of PS or PSR and a Content Type of `postscript`.¹

Table 3-4 lists additional non-PostScript printers and shows the printer type and content type to use for configuring each printer.

Note – Sun does not support the following printers, but they are listed here for your convenience. If you have questions about one of these products, contact the printer manufacturer.

Table 3-4 Non-PostScript Printers

Printer	Printer Type	Content Type
Daisy	daisy	simple
Datagraphix	datagraphix	simple
DEC LA100	la100	simple
DEC LN03	ln03	simple
DECwriter	decwriter	simple
Diablo	diablo	simple
	diablo-m8	simple
Epson 2500 variations	epson2500	simple
	epson2500-80	simple
	epson2500-hi	simple
	epson2500-hi80	simple
Hewlett-Packard HPCL printer	hplaser	simple

1. PSR reverses the pagination, printing the pages in reverse order, with the banner page last.

Table 3-4 Non-PostScript Printers (Continued)

Printer	Printer Type	Content Type
IBM Proprinter	ibmproprinter	simple
Qume Sprint 5	qume5	simple
Texas Instruments 800	ti800	simple

If you want to set up a printer that is not listed in this table, see “Printer Entries in the terminfo Database” on page 84.

Routine Printer Administration



This chapter describes how to administer printers after they are set up.

If you want to skip the background information that explains the concepts of administering printers, and proceed directly to step-by-step instructions, use the following table to find the page where the instructions for a specific task begin.

<i>How to Check the Status of Printers and Print Requests</i>	<i>page 105</i>
<i>How to Cancel Print Requests</i>	<i>page 111</i>
<i>How to Accept or Reject Print Requests</i>	<i>page 112</i>
<i>How to Move Print Requests to Another Printer</i>	<i>page 113</i>
<i>How to Change the Priority of a Print Request</i>	<i>page 114</i>
<i>How to Hold, Resume, or Print a Request Next</i>	<i>page 115</i>
<i>How to Enable or Disable a Printer</i>	<i>page 116</i>
<i>How to Remove a Printer From Service</i>	<i>page 118</i>
<i>How to Start or Stop the Print Service Scheduler</i>	<i>page 120</i>
<i>How to Monitor and Clean Out Log Files</i>	<i>page 122</i>

If you want to review background information first, read “About Routine Printer Administration” on page 100.

You can refer to the following chapters and appendixes for additional information about printing:

- See Chapter 3, “Setting Up Printers,” for information about setting up basic printing services using Printer Manager, an Administration Tool application.
- See Chapter 5, “Managing Character Sets, Filters, Forms, and Fonts,” for tasks specific to different types of printers.
- See Chapter 6, “Setting Printing Policies,” for to set printing policy, such as whether banner pages are required.
- See Chapter 7, “Troubleshooting Printing Problems,” if you have trouble setting up or administering printing.
- See Appendix A, “Setting Up Printer Services Using the Command-Line Interface,” for information about setting up basic printing services using the LP print service’s command-line interface.
- See Appendix B, “Understanding and Customizing the LP Print Service,” for a description of the LP print service and how to customize various aspects of it.

About Routine Printer Administration

This section describes the tasks you will perform repeatedly to support the printing needs of the users in your network. These tasks include:

- Obtaining information about printers and print requests
- Managing print requests—canceling, moving, or changing the priorities of print requests; putting print requests on hold; and resuming print requests
- Making printers available to users
- Starting and stopping the LP print service scheduler
- Removing a printer
- Cleaning out log files

For a quick reference to the commands for routine printer administration, see Table 4-3 on page 124.

Checking the Status of Printers and Print Requests

Many routine administration tasks you perform for printers require information about the printer and the print requests queued to it. For example, to cancel a print request, you have to know the print request's identification number (ID).

You use the `lpstat` command to find out the status of the LP print service. You also can check the status of print jobs in print queues, determine which printers are available for use, and examine the characteristics of those printers.

Managing Print Requests

While the LP print service is operating on one or more systems, you may have to manipulate the queued print requests. You can cancel, move, hold, or resume print requests, as well as change the priorities of print requests. These actions can help you to keep printing services operating smoothly. For example, if a printer has to be shut down for repairs, you can move the jobs to another printer.

Canceling Print Requests

Use the `cancel` command to cancel print requests from printer queues or to cancel jobs that are printing. There are three ways to use the `cancel` command:

- To cancel requests by identification number (ID)
- To cancel requests from a specific user on all, or specified, printers
- To cancel the job currently printing

Only the user who submitted the request, or someone logged in as `root` or `lp` can cancel a print request.

To cancel a specific request, you need to know its request ID. The request ID always includes the name of the printer, a dash, and the number of the print request. When you submit the print request, the request ID is displayed. If you do not remember the print request ID, you can get it easily by using the `lpstat` command without any options.

Moving Print Requests

As an administrator, if you plan to change the way a printer is used, or decide to take a printer out of service, you should have the print service reject additional print requests, and then move or cancel any requests that are currently queued to the printer. The LP print service will try to find alternate printers, if the user has not specified a printer for the job. Requests for a specific printer are not automatically moved. If you do not move them first, the LP print service will cancel them. Use the `lpmove` command to move such requests.

When you move requests, their print request IDs are not changed, so users can still find their requests. Requests that have requirements (for example, content type, forms, and so on) that cannot be met by the newly designated printer are not moved.

Changing the Priority of a Print Request

Once a user has submitted a print request, you can change its priority in the print server queue in the following ways:

- You can put any print request on hold that has not finished printing. Putting a request on hold stops it, if it is currently printing, and keeps it from printing until you resume printing it. Other print requests go ahead of the on-hold request.
- You can move any print request to the head of the queue, where it will be the next job eligible for printing. If you want a job to start printing immediately, you can interrupt the job that is currently printing by putting it on hold.
- You can change the priority of a job still waiting to be printed, moving it in the queue so that it is ahead of lower priority requests and behind requests at the same level or at a higher priority.

See the section, “How to Set Default Priority and Priority Limits for Users” on page 167 in Chapter 6, “Setting Printing Policies,” for information about changing user priority limits.

Users may put their own print requests on hold and resume them, but they may not resume a print request that has been put on hold by the administrator.

Making Printers Available

You can control the availability of printers in two ways:

- By controlling the flow of requests to the print queue
- By controlling the availability of the printer

After you have set up printers on the network, you may want to change the ability of users to submit jobs to a particular print queue. To do this, you use the `accept` and `reject` commands. You also may need to use the `disable` command to disable a printer for troubleshooting or maintenance, and then use the `enable` command to put the printer back into service. You can restrict access to the `enable` and `disable` commands to root and lp, or give users permission to access these commands. See “Permitting Users to Enable or Disable a Printer” on page 155 for more information.

When configuring a print server or a print client, you must instruct the LP print service both to accept print requests and to enable the printer to process those requests so that the requests can be printed.

Controlling the Flow of Print Requests to the Queue

The LP print service either accepts a file for printing and puts it in the print queue, or it rejects the file. If the file is rejected, the print service returns a message to the user who submitted the request, saying that requests are not being accepted for a specific printer. The `accept` and `reject` commands control the flow of print requests to the print queue. The `reject` command stops any further print requests from being added to the queue for the specified printer or printer class. It does not affect print requests that are already in the print queue.

You can reject requests for several printers or printer classes by using one `reject` command in which you list the printer names on the same line, separating the names with spaces. You can specify a reason for not accepting requests. The reason will be displayed on the user’s system whenever he or she tries to send a file to that printer.

Controlling the Availability of Printers

The printer is either enabled and capable of printing from the print queue, or it is disabled and does not print any files. When a printer is enabled, it continues to print jobs from the print queue until the queue is empty, even if the LP print service rejects additional files for the print queue.

The `enable` and `disable` commands control whether a printer processes queued requests. You use the `enable` command when a printer is newly configured, when a printer has been disabled, or when you want to force the LP print service to override a time-delay due to a fault.

When you use the `enable` or `disable` command on a print client, the command only enables or disables the transfer of the print requests from the print client to the print server, because a print client does not have a printer connected to it. You have to run these commands on the print server to have the printer actually stop or start processing print requests. You can use the `disable` command to temporarily disconnect a print client from any of the printers it is configured to use. The named printer itself is not disabled, and other clients can continue to use it.

Removing a Printer

When you move a printer from one system to another, rename the printer, or remove it from service, you should remove the old printer definitions from the LP print service on both the print server system and print client systems.

Starting and Stopping the LP Print Service Scheduler

Printer daemons may fail either on a print server or on a print client, and need to be restarted. You use the `/usr/lib/lp/lpsched` command to run the print scheduler. When `lpsched` is not running, no jobs are printed. When `lpsched` is restarted, it begins handling queued printing requests. Any jobs that were being printed when `lpsched` was stopped are reprinted in their entirety.

Cleaning Out Log Files

The `lpsched`, `lpNet`, and `requests` log files in the `/var/lp/logs` directory grow as information is appended. The LP print service uses a default cron job to clean out the log files. The `lp cron` job is located in the `/var/spool/cron/crontabs` file. It periodically moves the contents of the log files. The contents of `log` are moved to `log.1`, and the contents of `log.1` are moved to `log.2`. The contents of `log.2` are lost (that is, replaced by the former contents of `log.1`) when `log.2` gets overwritten.

Instructions for Performing Routine Printer Administration

This section provides step-by-step instructions for performing tasks related to routine printer administration. For many tasks, you will find an example of user input and system output after the instructions.

▼ How to Check the Status of Printers and Print Requests

To check the status of printers and print requests:

◆ **Type `lpstat [option]` and press Return.**

You can obtain different types of printing status information.

Use the table below to choose the option you want. Items in lists (for example, *printer-names*) are separated by commas. To indicate all items of the given type, you can substitute `all` for the list or omit the list.

Table 4-1 Options to the `lpstat` Command

Option	Description
<code>none</code>	Show all current users' print requests.
<code>-a [printer-names]</code>	Accept. Show if printers are accepting requests.
<code>-c [class-names]</code>	Class. Show classes and their members.
<code>-d</code>	Destination. Show the system's default printer.
<code>-f</code>	Forms. Show forms.
<code>-o [list]</code>	Output. Show status of print requests. <i>list</i> can be printer names, printer class names, or request IDs.

Table 4-1 Options to the lpstat Command (Continued)

Option	Description
-p [<i>printer-names</i>] [-D] [-l]	Printer. Show status of printers. -D displays the printer description. -l displays printer's characteristics.
-r	Running. Show if the LP scheduler is running.
-R <i>request-id</i>	Request. Show position of job in the queue.
-s	Status. Show summary of LP print service status.
-S	Sets. Show character sets.
-t	Total. Show complete status of LP print service (-s option information, plus the acceptance and idle/busy status of all printers).
-u [<i>user-list</i>]	User. Show requests by user.
-v [<i>printer-names</i>]	Verify. Show device for local printer, or server name for remote printer.

See the reference manual page for `lpstat(1)` for more information.

To review the status of requests already printed:

1. Become root or lp.
2. Type `cd /var/lp/logs` and press Return.
3. Look for the request ID in the `requests` log file.
A history of print requests is kept in `/var/lp/logs/requests`. For a description of the codes that occur at the beginning of each line, see Table B-6 on page 316.

```
# pwd
/var/lp/logs
# tail requests
s 0x1010
= slw2-20, uid 200, gid 200, size 5123, Mon Nov 18 01:24:01 EST
1992
z slw2
C 1
D slw2
```

```
F /etc/motd
P 20
t simple
U irving
s 0x1010
#
```

Examples of Checking the Status of Printers and Print Requests

The examples in this section show many ways to obtain the information you may need for performing routine printer administration. Each example uses the `lpstat` command.

Checking the Status of the Current User's Print Requests

You can use the `lpstat` command without any options to check the status of all your print jobs (requests). In the following example, on the system `terra`, one file is queued to the printer `luna`.

```
terra% lpstat
luna-1          fred          1261   Mar 12 17:34 being filtered
terra%
```

When used to check the status of print jobs, the `lpstat` command displays one line for each print job. From left to right, the line shows the request ID, the user, the output size in bytes, the date and time of the request, and variable information about the request, like “being filtered.”

Checking the Status of Print Requests by Printer

You can use the `lpstat -o` command to check the status of print requests to a specific printer. You can omit the printer name to check the status of all requests to all printers.

```
saturn% lpstat -o slw2
slw2-78          root          1024   Jan 14 09:07
slw2-79          root          1024   Jan 14 09:08
```

Checking Available Printers

The following example shows how to obtain the status of the print service scheduler, the default printer destination, and a list of the printers (and the print servers they are on) available to your system.

```
saturn% lpstat -s
scheduler is running
system default destination: slw2
device for slw2: /dev/term/a
system for luna: terra
system for ganymede: jupiter
saturn%
```

This example shows that on saturn the scheduler is running, the default printer is luna, and two network printers are available, one of which is luna.

Obtaining All Status Information

You can use the `lpstat -t` command to obtain a complete status of the LP print service. You get all the information shown by the `-s` option, plus the status of all printers, whether they are accepting requests and are busy, or whether they are idle.

In the following example, there are no jobs in the print queue. When files are spooled for printing, the status of those print requests is also displayed.

```
saturn% lpstat -t
scheduler is running
system default destination: slw2
device for slw2: /dev/term/a
system for ganymede: jupiter
luna accepting requests since Wed Jan  2 18:20:10 PST 1992
ganymede accepting requests since Mon Mar  4 15:53:47 PST 1992
printer luna is idle. enabled since Wed Jan  2 18:20:22 PST 1992.
available.

printer ganymede is idle. enabled since Mon Mar  4 15:53:44 PST
1992. available.
saturn%
```

Obtaining Printer Status

You can request printer status information for all printers or individual printers by using the `-p` option to `lpstat`. This option shows whether the printer is active or idle, when it was enabled or disabled, and whether it is available to accept print requests. You can request the status for an individual printer by specifying its name; otherwise, all available printers are checked.

For example, the status for the printer `luna` might appear as follows:

```
saturn% lpstat -p luna
printer luna is idle. enabled since Wed Jan 2 18:20:22 PST 1992.
available.
saturn%
```

If the printer had jobs in the print queue, the active job ID would be displayed instead of the “idle” status.

Obtaining Printer Characteristics

If you want to check all characteristics for a printer, use the `-p` option with the `-l` (long) option to `lpstat`. This command displays a table of all the configuration information that is used by the LP print service for a printer. This command is especially useful for finding the printer type and content type. You can request the characteristics for an individual printer by specifying its name; otherwise, the characteristics of all available printers are displayed.

The following example is for a remote printer on a SunOS 4.1 server.

```
saturn% lpstat -p ganymede -l
printer ganymede is idle. enabled since Wed Jan 2 18:20:22 PST
1992. available.
    Content types: any
    Printer types: unknown
    Description: remote
    Users allowed:
        (all)
    Forms allowed:
        (none)
    Banner not required
    Character sets:
        (none)
    Default pitch:
    Default page size:
saturn%
```

The following example is for a local SunOS 5.x printer.

```
saturn% lpstat -p slw2 -l
printer slw2 is idle. enabled since Wed Jan 2 18:20:22 PST 1992.
available.
    Content types: postscript
    Printer types: PS
    Description: local printer - files
    Connection: direct
    Interface: /usr/lib/lp/model/standard
    After fault: continue
    Users allowed:
        (all)
    Forms allowed:
        (none)
    Banner not required
    Character sets:

    Default pitch:
    Default page size: 80 wide 66 long
saturn%
```

▼ How to Cancel Print Requests

1. (Optional) Type `lpstat [option]` and press Return.

Use this step if you do not know the request IDs of the jobs to cancel. You do not need to use an option to see your own requests. Use `-o printer-name` to see jobs queued to a printer, or `-o all` to see all jobs queued to all printers. Use `-u username` to see jobs belonging to a specific user.

2. Use the `cancel` command as follows:

◆ To cancel a list of print requests, type `cancel request-ids` and press Return.

When you use `cancel`, a message tells you the requests are canceled, and the next job is printed.

```
saturn% cancel luna-3 luna-4
request "luna-3" cancelled
request "luna-4" cancelled
saturn%
```

◆ To cancel all your requests, or requests belonging to other users, type `cancel -u user-list [printer-names]` and press Return.

You must be root or lp to cancel requests of other users. Separate user names with commas, or enclose the list in quotation marks.

```
saturn# cancel -u ignatz luna
request "luna-1" cancelled
request "luna-2" cancelled
request "luna-5" cancelled
saturn#
```

◆ To cancel the job currently printing, type `cancel printer-name` and press Return.

You must be root or lp to cancel a print request that is not your own.

```
saturn% cancel luna
request "luna-3" cancelled
saturn%
```

▼ How to Accept or Reject Print Requests

To perform these tasks, you must become root or lp.

To set a printer to accept print requests:

◆ **Type `accept printer- or class-names` and press Return.**

The printer (or class of printer) begins accepting print requests. You can specify several printers or classes by listing the printer names, and separating the names with a space. If you want a print server to accept print requests from a print client, run the `accept` command on both the print client and the print server.

```
luna# accept luna  
destination "luna" now accepting requests
```

To set a printer to reject (stop accepting) print requests:

◆ **Type `reject -r "reason" printer- or class-names` and press Return.**

The printer (or class of printer) stops accepting new requests. You can specify several printers or classes by listing the printer names, and separating the names with a space. The queued requests will continue printing as long as a printer is enabled. You can reject print requests for all users of a printer by typing the `reject` command on the print server. Alternatively, you can reject print requests for a specific print client or group of print clients by typing the `reject` command on each of the print clients. If you do not want to specify a reason, omit `-r` and the reason for rejecting print requests.

```
terra# reject -r "luna is down for repairs" luna  
destination "luna" will no longer accept requests
```


▼ How to Move Print Requests to Another Printer

You can use the `lpmove` command to move specific print requests or all print requests from one printer or printer class to another.

1. Become root or lp.

- (Recommended) Type `reject -r "reason" printer-name` and press Return.**
You should prevent any further print requests from being sent to the print queue while you are in the process of moving requests to another printer. This step may not be necessary if you are going to move all requests from one printer to another, because, in that case, `lpmove` automatically stops accepting requests for the first printer. However, by explicitly rejecting requests, you can give users a reason that explains why the printer is not accepting requests.

```
terra# reject -r "luna will be down today for repairs" luna
```

3. (Optional) Type `lpstat -o printer-name1` and press Return.

If you are going to move individual (rather than all) requests, you need to know the request IDs. To move all print requests from one printer (or class of printers) to another, you do not need to know the request IDs. However, it is a good idea to see how many print jobs are affected before you move the jobs.

4. (Optional) Type `lpstat -p printer-name2` and press Return.

You should check if the destination printer is accepting requests, before moving requests to it.

5. Move specific or all print requests:

- ♦ **To move individual requests, type `lpmove request-id-list printer-name` and press Return.**

This moves the specified print requests from the original printer to the new destination printer.

```
terra# lpmove luna-10 luna-11 slw2
```

Note – If the user has specified a destination of any printer (using the `-d any` command as an argument to `lp`), the `lpmove` destination is changed from `any` to the name of the printer you specified. You can change the destination back to `any` by typing `lp -i request-id -d any` and pressing Return.

♦ **To move all requests, type `lpmove printer-name1 printer-name2` or `lpmove class1 class2` and press Return.**

This moves all print requests from the first printer or printer class named to the second printer or printer class named. The LP print service also stops accepting requests for the first printer, just as if you ran the command `reject printer-name1`. If some requests cannot be printed on the new printer, they are left in the queue for the original printer.

```
terra# lpmove luna slw2
```

6. After the first printer is ready for use again, type `accept printer-name1` and press Return.

```
terra# accept luna
```

▼ How to Change the Priority of a Print Request

1. Become root or lp.

2. Type `lp -i request-id -q new-priority-level` and press Return.

The request is repositioned in the queue. You change the priority level from 0, the highest priority, to 39, the lowest priority. The default is 20.

```
saturn% lp -i ganymede-79 -q 1
```

Note – You can change the priority of only one print request at a time.

▼ How to Hold, Resume, or Print a Request Next

To perform these tasks, you must become root or lp.

To put a print request on hold:

- ◆ **Type `lp -i request-id -H hold` and press Return.**

The print request is placed on hold until you cancel it or instruct the print service to resume printing the request.

```
saturn% lp -i ganymede-79 -H hold
```

To resume a print request:

- ◆ **Type `lp -i request-id -H resume` and press Return.**

The print request that was put on hold is placed back in the queue and will be printed according to its priority and placement in the queue. If you put a hold on a print job that is already printing, `resume` puts the print request at the head of the queue so that it becomes the next request printed.

```
saturn% lp -i ganymede-79 -H resume
```

To set the next print request:

- ◆ **Type `lp -i request-id -H immediate` and press Return.**

The print request is placed at the head of the queue. If a job is already printing, you can put it on hold to allow the next request to print immediately.

```
saturn% lp -i ganymede-79 -H immediate
```

▼ How to Enable or Disable a Printer

To perform these tasks, you must become root or lp.

To enable a printer:**♦ Type `enable printer-names` and press Return.**

The printer begins processing queued requests. You can list several printer names, separating the names with a space. If you want a print server to process files from a print client, run the `enable` command on both the print client and the print server.

```
terra# enable luna
printer "luna" enabled
```

Note – You can enable or disable individual printers, not classes of printers.

To disable a printer:♦ **Type** `disable [option] printer-names` and press Return.

The printer stops processing queued requests. You can list several printer names, separating the names with a space. If you want a print client to stop sending requests to a print server, run the `disable` command (without `-c` or `-W`) on the print client.

Use the following table to choose an option.

Table 4-2 Options to the `disable` Command

Option	Description
<code>none</code>	Disable the printer, saving the current job for reprinting when the printer is re-enabled.
<code>-c</code>	Cancel the current job, then disable the printer. The current job is not printed later.
<code>-W</code>	Wait until the current job is finished before disabling the printer.
<code>-r "reason"</code>	Provide users with a reason why the printer is disabled. The reason is stored and displayed whenever a user checks on the status of the printer (<code>lpstat -p</code>).

For example, suppose you want to stop the current job, and save it to print later, and give a reason why the printer is disabled.

```
terra# disable -r "changing the form" luna
terra# lpstat -p luna
printer luna disabled since Mon Mar  2 12:11:31 EST 1992.
available.
      changing the form
terra#
```

Note – You cannot enable or disable classes of printers. Only individual printers can be enabled or disabled.

See “Permitting Users to Enable or Disable a Printer” on page 155 for information about who is authorized to use the `enable` and `disable` commands. The `enable` and `disable` commands, by default, are owned by `lp`, and may only be run by `lp` or `root`.

To disconnect a print client from a print server:

- ◆ **On the print client, type `disable printer-name -r "comment"` and press Return.**

The transfer of files from the print client to the print server is stopped. The printer can still serve other clients. The `-w` and `-c` options do not apply.

```
saturn# disable luna -r "luna is down for repairs"
printer "luna" disabled
```

▼ How to Remove a Printer From Service

The following steps show you how to use the LP print service commands to remove a printer. Alternatively, you could use the Printer Manager to perform all the steps, except step 4. The Printer Manager's Delete Printer option is the equivalent of `lpadmin -x`.

Information You Need

- Print server's superuser password
- Names of printer, server, and clients

- 1. On the system to which the printer is connected, become root or lp.**
- 2. Type `reject printer-name` and press Return.**
This prevents any new requests from being queued to the printer, while you are in the process of removing it.
- 3. Type `disable printer-name` and press Return.**
This discontinues printing any requests that may still be in the queue.
- 4. Type `lpmove printer-name1 printer-name2` and press Return.**
This moves the requests still in the queue to another printer.
- 5. Type `lpadmin -x printer-name` and press Return.**
The configuration information for the printer you specify is removed from the LP configuration files.

6. (Optional) Type `lpssystem -r client-name1 client-name2...` and press Return.

Specify the names of print clients that were using the printer, unless they are still using another printer on this server. This removes the print client names from the `/etc/lp/Systems` file.

7. On each client, become root or lp.

8. Type `lpadmin -x printer-name` and press Return.

The configuration information for the printer you specify is removed from the LP configuration files.

9. (Optional) Type `lpssystem -r server-name` and press Return.

Specify the name of the print server to which the printer was connected, unless the print client is still using another printer on this server. This removes the print server name from the `/etc/lp/Systems` file.

Example of Removing a Printer From Service

In this example, the printer `luna` on the print server `terra` is being removed from the print client `jupiter`.

```
jupiter% su
Password:
jupiter# lpstat -r
scheduler is running
jupiter# lpadmin -x luna
jupiter# lpssystem -r terra
Removed "terra".
jupiter# rlogin terra
Password:
terra%
su
Password:
terra# lpssystem -r jupiter
Removed "jupiter".
```

▼ How to Start or Stop the Print Service Scheduler

To start the print service scheduler:

1. **Become root or lp.**
2. **(Optional) Check if the printer is running. To do this, type `lpstat -r` and press Return.**
If the scheduler is not running, the message `scheduler is not running` is displayed.
3. **Type `/usr/lib/lp/lpsched` and press Return.**

```
saturn# /usr/lib/lp/lpsched
Print services started.
```

4. If the scheduler does not start:

- a. **Type `ls /var/spool/lp` and press Return.**

```
saturn# ls /var/spool/lp
SCHEDLOCK  bin      logs      requests  temp
admins     fifos    model     system    tmp
saturn#
```

- b. **If a SCHEDLOCK file is listed, type `rm /var/spool/lp/SCHEDLOCK` and press Return.**
The SCHEDLOCK file is removed. (The file `/var/spool/lp/SCHEDLOCK` is created when the scheduler starts, to make sure that only one copy of `lpsched` is running. The `lpshut` command (described on page 138) automatically removes the SCHEDLOCK file. If `lpsched` is stopped in any other way, you must remove the SCHEDLOCK file before you can restart `lpsched`.)
- c. **Type `/usr/lib/lp/lpsched` and press Return.**
The print service scheduler should start. If the message `Print services started` does not appear, check the `lpsched` log in the `/var/lp/logs` directory for error messages that may explain the problem and point to a solution.

5. If the scheduler appears to start, but does not start:

You may get the message `Print services started` but `lpstat -r` says the scheduler is not running and its process does not show up with the `ps` command. In addition, when you try to use any `lp` command, error messages appear.

a. **Check the `lp sched` log in the `/var/lp/logs` directory for error messages that may explain the problem and point to a solution.**

b. **If there is no apparent error, look for recent files, like logs, that the scheduler may be getting stuck on.**

While the scheduler is starting up, it checks many of the directories and files it uses or creates during normal operation. If there is a problem with one of the files, it may prevent the scheduler from completing its initialization. You may need to remove files recently created by the LP print service, like those in `/var/spool/lp/logs`.



Caution – The next step is a last resort. Be very careful not to delete files critical to the operation of the LP print service, like the configuration files in `/etc/lp`.

c. **Delete files that appear troublesome, but first copy them to other file names.**

d. **Type `/usr/lib/lp/lpsched` and press Return.**
The scheduler should start.

To stop the print scheduler:

1. (Optional) Check if the scheduler needs to be stopped. To do this, type `lpstat -r` and press Return.

You may want to check if the scheduler needs to be stopped. When the scheduler is running, the message `scheduler is running` is displayed.

2. Type `lpshut` and press Return.

```
saturn# lpshut
Print services stopped.
```

3. If the scheduler does not stop:**a. Type `ls /var/spool/lp` and press Return.**

```
saturn# ls /var/spool/lp
SCHEDLOCK  bin          logs         requests    temp
admins     fifos        model        system      tmp
saturn#
```

b. If a SCHEDLOCK file is not listed,**type `cd /var/spool/lp; touch SCHEDLOCK; chown lp\ SCHEDLOCK; chmod 664 SCHEDLOCK` and press Return.**

The SCHEDLOCK file is created. This file is normally created when the scheduler starts, to make sure that only one copy of lpsched is running. The lpshut command automatically removes the SCHEDLOCK file. If the file is removed in any other way, you must re-create the SCHEDLOCK file before you can stop lpsched.

c. Type `lpshut` and press Return.

The print scheduler is stopped.

▼ How to Monitor and Clean Out Log Files

The LP print service records a history of print services in three logs located in the `/var/lp/logs` directory:

- requests
- lpsched
- lpNet

To perform the following tasks, you must become root or lp.

To monitor the logs:**1. Type `cd /var/lp/logs` and press Return.****2. Type `ls -l *` and press Return.****3. Use the `head` and `tail` commands to look at the oldest and newest information in the files.**

To change how the log files are cleaned out:**1. Type** `cd /var/spool/cron/crontabs` **and press Return.**

This gets you to the directory where the `lp` crontab file is located.

2. Edit the `lp` file.

The `lp` file has entries like the following:

```
# At 03:13am:
# Move each days 'requests' to 'requests.1'.
# If there was an 'requests.1' move it to 'requests.2'.
# If there was an 'requests.2' then it is lost.
#
13 3 * * * cd /var/lp/logs; if [ -f requests ]; \
then if [ -f requests.1 ]; then /bin/mv requests.1 requests.2;\
fi; /usr/bin/cp requests requests.1; >requests; fi
#
```

The above entry periodically moves the contents of the current log file to *log.1*, and the contents of *log.1* to *log.2*. The contents of *log.1* and *log.2* are overwritten. This entry is one line in the `lp` crontab file, but it is split into several lines here for readability. It cleans up `requests` at 3:13 a.m. each day, changing the name to `requests.1`, and moving the previous day's contents to `requests.2`. This entry is sufficient to keep the old print request records from accumulating in the spooling file system.

You may want to condense information in the `requests` log to produce a report about the use of the LP print service or to help generate accounting information. You can produce a different script that examines the file and extracts information just before the cleanup procedure. See “Log Files” on page 313 for a description of the structure of the `requests` log.

3. Change the entries.

You can change how frequently the logs are cleaned out by changing the last character at the beginning of the entry; use an asterisk (*) for daily, or a zero (0) for once a week on Sundays.

Reference Information for Routine Printer Administration

This section provides additional information that you may need to understand the concepts of routine printer administration or to perform routine printer administration tasks.

Summary of LP Print Service Commands

Table 4-3 lists frequently used LP print service commands. All commands listed in Table 4-3 are either found in or linked to the `/usr/lib` directory. If you expect to use these commands frequently, consider including the `/usr/lib` directory in your `PATH` variable. See “Using the `PATH` Variable” on page 40 for information about setting the `PATH` variable. To use these administrative commands (except `lpstat`), you must be root or lp.

Table 4-3 Quick Reference to LP Print Service Commands

Task	Command
Activate a printer	<code>enable(1)</code>
Cancel a print request	<code>cancel(1)</code>
Send one or more file(s) to a printer	<code>lp(1)</code>
Report the status of the LP print service	<code>lpstat(1)</code>
Deactivate one or more printers	<code>disable(1)</code>
Permit print requests to be queued for a specific destination	<code>accept(1M)</code>
Prevent print requests from being queued for a specific destination	<code>reject</code> Described on <code>accept(1M)</code> manual page
Set up or change printer configuration	<code>lpadmin(1M)</code>
Set up or change filter definitions	<code>lpfilter(1M)</code>
Set up or change preprinted forms	<code>lpforms(1M)</code>
Mount a form	<code>lpadmin(1M)</code>
Move output requests from one destination to another	<code>lpmove</code> Described on <code>lpsched(1M)</code> manual page
Start the LP print service scheduler	<code>lpsched(1M)</code>
Stop the LP print service scheduler	<code>lpshut(1M)</code> Described on <code>lpsched(1M)</code> manual page
Set or change the default priority and priority limits that can be requested by users of the LP print service	<code>lpusers(1M)</code>

Managing Character Sets, Filters, Forms, and Fonts

5

This chapter describes how to set up and administer features of the LP print service that are specific to different types of printers.

If you want to skip the background information that explains the concepts of managing these features, and proceed directly to step-by-step instructions, use the following table to find the page where the instructions for a specific task begin.

<i>How to Set Up Hardware-Mounted or Selectable Character Sets</i>	<i>page 136</i>
<i>How to Mount, Unmount, and Set Alerts for Hard Character Sets</i>	<i>page 138</i>
<i>How to Add, Change, Remove, or Restore Print Filters</i>	<i>page 140</i>
<i>How to View Print Filter Definitions</i>	<i>page 142</i>
<i>How to Add, Change, or Remove a Form</i>	<i>page 143</i>
<i>How to Mount, Unmount, and Specify Alerts for Forms</i>	<i>page 145</i>
<i>How to Obtain Information About Forms</i>	<i>page 147</i>
<i>How to Allow or Deny User Access to Forms</i>	<i>page 149</i>
<i>How to Allow or Deny Use of Forms on a Printer</i>	<i>page 150</i>
<i>How to Install Downloaded and Host-Resident PostScript Fonts</i>	<i>page 151</i>

If you want to review background information first, read the next section, “About Character Sets, Filters, Forms, and Fonts” on page 126.

You can refer to the following chapters and appendixes for additional information about printing:

- See Chapter 3, “Setting Up Printers,” for information about setting up basic printing services using Printer Manager, an Administration Tool application.
- See Chapter 4, “Routine Printer Administration,” for information about how to administer printing services once the printers have been set up.
- See Chapter 6, “Setting Printing Policies,” for tasks to set printing policy, such as whether banner pages are required.
- See Chapter 7, “Troubleshooting Printing Problems,” if you have trouble setting up or administering printing.
- See Appendix A, “Setting Up Printer Services Using the Command-Line Interface,” for information about setting up basic printing services using the LP print service’s command-line interface.
- See Appendix B, “Understanding and Customizing the LP Print Service,” for a description of the LP print service and how to customize various aspects of it.

About Character Sets, Filters, Forms, and Fonts

Depending on your site’s requirements and the types of printers you have on the network, you may have to set up and administer printer-specific features of the LP print service. For example, you can assign different print wheels, filters, and forms to different printers.

Setting Up and Administering Character Sets

Printers differ in the method they use to print text in various font styles. For instance, PostScript printers treat text as graphics. These printers can generate text in different fonts, and place the text in any position, size, or orientation on the page. Other types of printers support a more limited number of font styles and sizes, using either print wheels, font cartridges, or preprogrammed, selectable character sets. Only one of these printing methods usually applies to a given printer type.

Print wheels (or print balls) and font cartridges, from the point of view of the LP print service, are similar, because someone must intervene and mount the hardware on the printer, when needed. Character sets that require you to

physically mount a wheel or cartridge are referred to as *hardware character sets*. Character sets that do not require hardware mounting, that come preprogrammed with the printer, and can be selected by a print request, are referred to as *software character sets*.

When you set up a non-PostScript printer, you need to tell the LP print service which print wheels or selectable character sets are available to users. When users make print requests, the `lp -S` command lets them specify a print wheel or selectable character set to use for the print job. The user does not have to know which type of character set applies; he or she just refers to the font style by the name you have defined.

Selectable Character Sets

The selectable character sets supported by a printer are listed in the `terminfo` entry for that printer. For example, the entry for the `ln03` printer is `/usr/share/lib/terminfo/l/ln03`. You can find the names of selectable character sets for any printer type in the `terminfo` database by using the `tput` command. The syntax for the `tput` command is:

```
tput -T printer-type csnm n
```

The `csnm` option is an abbreviation for *character set number*. The number starts with 0, which is always the default character set number after the printer is initialized. You can repeat the command, using 1, 2, 3, and so on in place of the 0, to display the names of the other character sets. For each selectable character set, a `terminfo` name (for example, `usascii`, `english`, `finnish`, and so forth) is returned.

In general, the `terminfo` character set names should match closely the character set names used in the manufacturer's documentation for the printer. Because some manufacturers do not use the same character set names, the `terminfo` names may differ from one printer type to the next.

You do not have to register the selectable character set names with the LP print service. However, you can give them more meaningful names or aliases.

Note – If you do not specify the selectable character sets that can be used with a printer, the LP print service assumes that a printer can accept any character set name (cs0, cs1, cs2, etc.) or the `terminfo` name known for the printer.

Users can employ the `lpstat -p -l` command to display the names of the selectable character sets that you have defined for each printer on a print server.

Note – Character sets for PostScript printers are not listed when you use the `lpstat -p -l` command because the PostScript fonts are controlled by PostScript filters, not by entries in the `terminfo` database. See “Installing and Maintaining PostScript Fonts” on page 132 for information about how to administer PostScript fonts.

Hardware-Mounted Character Sets

Another method to obtain alternative character sets is to use removable daisy wheels (print balls) or font cartridges that are physically mounted on a printer.

To administer hardware-mounted character sets, you inform the LP print service of the names you want to use for the available print wheels, and how you want to be alerted when a printer needs a different print wheel. Then, when a user requests a particular character set, using the `lp -S` command, the scheduler sends an alert to mount the print wheel, and the print request is placed in the print queue. When you mount the correct print wheel, and tell the LP print service that the print wheel is mounted, the job is printed.

If you do not specify multiple print wheels or cartridges for a printer, the LP print service assumes the printer has a single, fixed print wheel or cartridge, and users may not ask for a special print wheel or cartridge when using the printer.

Unlike selectable character sets, the names you use for print wheels or cartridges are not tied to entries in the `terminfo` database. Print wheel or cartridge names are used just to communicate with the LP print service and its users.

The names you choose for print wheels or cartridges, however, should have meaning to the users; the names should refer to font styles. In addition, the names should be the same across printers that have similar print wheels or

cartridges, or selectable character sets. That way, the user can ask for a font style (character set), without regard to which printer, or even whether a print wheel or cartridges, or selectable character set will be used.

Of course, you and the printer users should agree on the meanings of print wheel or cartridge names. Otherwise, what a user asks for, and what you mount, may not be the same character set.

Alerts for Mounting Print Wheels or Cartridges

You request alerts for mounting print wheels or cartridges in the same way you request other alerts from the LP print service. See “Setting Printer Fault Alerts” on page 156 for general information about alerts.

Setting Up and Administering Print Filters

Print filters are programs that convert the content type of a file to a content type that is acceptable to a destination printer.

The LP print service uses filters to:

- Convert a file from one data format to another so that it can be printed properly on a specific type of printer
- Handle the special modes of printing, like two-sided printing, landscape printing, or draft- or letter-quality printing
- Detect printer faults and notify the LP print service of them so that the print service can alert users and system administrators

Not every print filter can perform all these tasks. Because each task is printer-specific, they can be implemented separately.

The LP print service provides the PostScript filters listed in Table A-1 on page 279. The filter programs are located in the `/usr/lib/lp/postscript` directory. Usually, for PostScript printing, you do not need to do anything beyond installing them when setting up a print server. The Printer Manager automatically enables the supplied filters.

If, however, you administer other printers, you may need to administer print filters for them.

Creating a Print Filter

Creating a new print filter is not easy, and usually requires extensive programming experience. Filters contain input types, output types, and complex options that provide a language to process command-line arguments within the filter.

Adding, Changing, Removing, and Restoring Print Filters

Print filters are added, changed, or removed on both the print server and the print clients.

You use the `lpfilter` command to manage the list of available filters. System information about filters is stored in the `/etc/lp/filter.table` file. The `lpfilter` command gets the information about filters to write to the table from filter descriptor files. The filter descriptor files supplied (PostScript only) are located in the `/etc/lp/fd` directory. The actual filter programs are located under `/usr/lib/lp`.

The LP print service imposes no fixed limit on the number of print filters you can define. You may remove filters that are no longer used to avoid extra processing by the LP print service, which must examine all filters to find one that works for a specific print request. If in doubt, do not remove a filter.

As you add, change, or delete filters, you may overwrite or remove some of the original filters provided by the LP print service. You can restore the original set of filters, if necessary, and remove any filters you have added.

Setting Up and Administering Forms

A *form* is a sheet of paper on which information is printed in a predetermined format. Unlike plain paper stock, forms usually have text or graphics preprinted on them. Common examples of forms are company letterhead, invoices, blank checks, receipts, and labels.

The term *form* includes two senses: the physical medium (the paper) and the software information that defines a form to the LP print service.

The LP print service allows you to control the use of forms. This chapter describes how to add, change, remove, and mount forms, and how to control access to forms.

Adding, Changing, or Removing Forms

When you add a form, you tell the LP print service to include the form in its list of available forms. You also have to supply the information required to describe or define the form. Although you can enter such definitions when you add the form, it helps to create the definitions first and save them in files. You can then change the form definition by editing the file. See “How to Provide a New Form” on page 353 for information about how to create form definitions.

Note – No form definitions are supplied with the LP print service.

Changing a form is done by re-adding the form with a different definition.

The LP print service imposes no limit on the number of forms you can define. However, you should remove forms that are no longer appropriate. Obsolete forms may result in unnecessary processing by the print service.

Mounting Forms

To mount a form, you must load the paper into the printer, and notify the LP print service that the form is mounted. If you use one printer for different types of printing, including forms, you should disable the printer before you load the paper and mount the form. Then re-enable the printer when the form is ready; otherwise, the LP print service will continue to print files that do not need the form on the printer.

When you mount a form, make sure that it is aligned properly. If an alignment pattern has been defined for the form, you can ask that it be repeatedly printed after you have mounted the form, and if needed, until you have adjusted the printer so that the alignment is correct.

When you want to change or discontinue using a form on a printer, you have to notify the LP print service by unmounting the form.

Defining Alerts for Mounting Forms

You request alerts for mounting forms in the same way you request other alerts from the LP print service. See “Setting Printer Fault Alerts” on page 156 for general information about alerts.

Checking Forms

When you have defined a form for the LP print service, you can check it with either of two commands, depending on the type of information you want to check.

- Use the `lpforms` command to show the attributes of the form. You can also redirect the output of the command into a file to save it for future reference.
- Use the `lpstat` command to display the current status of the form. To protect potentially sensitive content, the alignment pattern is not shown.

If you are not sure about the name of an existing form, you can list the contents of the `/etc/lp/forms` directory to see the names of the forms there.

Controlling Access to Forms

You can control which printers and users have access to some or all of the forms available on the network. For example, you may want only the people in the payroll or accounts payable department to be able to print check forms. In addition, you may want the check forms to be available only on certain printers.

You restrict user access to forms by creating `allow` and `deny` lists. See “Creating allow and deny Lists” on page 160 for a general description of allow and deny lists.

If you want users on your system to have access to forms on a print server, include all users in the `allow` list for *both* the client system and in the `allow` list for the print server. If, on the other hand, a user is denied permission to use forms on a print server, you do not need to include that user in a `deny` list on the print client system.

It is a good idea to make the `deny` lists for print servers and clients match.

Installing and Maintaining PostScript Fonts

If you have a laser printer, you may need to install and maintain PostScript fonts. You may have to decide where PostScript fonts are installed and how they are managed. For many printers, the fonts are set up as part of the printer installation process.

PostScript fonts are stored in outline form, either on the printer or on a system that communicates with the printer. When a document is printed, the PostScript interpreter generates each character as needed (in the appropriate size) from the outline description of it. If a font required for a document is not stored on the printer being used, it must be transmitted to that printer before the document can be printed. This transmission process is called *downloading fonts*.

Fonts are stored and accessed in several ways:

- *Printer-resident fonts* are stored permanently on a printer. These fonts are installed in read-only memory (ROM) on the printer by the manufacturer. If the printer has a disk, you may need to install fonts on that disk. Most PostScript printers are shipped with 35 standard fonts.
- A *permanently downloaded font* is transmitted to a printer with a PostScript `exitserver` program. A permanently downloaded font remains in printer memory until the printer is turned off. Memory allocated to a downloaded font reduces the memory available on the server for PostScript print requests. Use of an `exitserver` program requires the printer system password and may be reserved for the printer administrator. You should permanently download fonts when most print requests serviced by that printer use the font.
- Fonts that are used infrequently or for special purposes can be stored on a user's system. The user can specify these fonts when submitting the print request. The fonts are appended to the print request and transmitted to the printer. When the print request is processed, the space allocated for the font is freed for other print requests.
- *Host-resident fonts* are stored on a system shared by many users. The system that stores the fonts may be a print server or a print client. Each user may request fonts in the document to be printed. This method is useful when there are numerous available fonts, or when these fonts are not used by all print requests. If the fonts will be used only on printers attached to a server, they should be stored on the print server. If the fonts are to be used by users on one system, who may send jobs to multiple printers on a network, they should be stored on the user's system.

The LP print service provides a special download filter to manage host-resident fonts. It also supplies `troff` width tables for the 35 standard PostScript fonts which reside on many PostScript printers, for use by the `troff` program.

Managing Printer-Resident Fonts

Most PostScript printers come equipped with fonts resident in the printer ROM. Some printers have a disk on which additional fonts are stored. When a printer is installed, you should add the list of printer-resident fonts to the font list for that printer. By identifying printer-resident fonts, you prevent fonts from being transmitted unnecessarily across a network. Each printer has its own list of resident fonts, which is contained in the file:

```
/etc/lp/printers/printer-name/residentfonts
```

When the printer is attached to a print server, make sure the list in the `residentfonts` file includes fonts that are on the print server and which are available for downloading to the printer.

You must edit the files containing the list of printer-resident fonts by using a text editor like `vi`.

Downloading Host-Resident Fonts

When a PostScript document contains a request for fonts not loaded on the printer, the *download filter* manages this request. The download filter uses PostScript document structuring conventions to determine which fonts to download.

LP print filters are either fast or slow. A fast filter quickly prepares a file for printing, and it must have access to the printer while the filter is processing. A slow filter takes longer to convert a file, and it does not need to access the printer while the filter is processing. An example of a slow filter is ASCII to PostScript.

The download filter is a fast filter; it downloads fonts automatically if the fonts are on the print server. The download filter may also be used to send fonts to a print server. To do this, you may create a new filter table entry which calls the download filter as a slow filter through the `-y` option to the `lp` command. Alternatively, you may force selection of this filter by changing the input type. For information about defining print filters, see “Defining a Print Filter” on page 336.

The download filter performs five tasks:

1. It searches the PostScript document to determine which fonts are requested. These requests are documented with the following PostScript structuring comments: `%%DocumentFonts: font1 font2 ...` in the header comments.
2. It searches the list of printer-resident fonts to determine if the requested font must be downloaded.
3. If the font is not resident on the printer, the download filter searches the host-resident font directory (by getting the appropriate file name from the map table) to determine if the requested font is available.
4. If the font is available, the filter takes the file for that font and appends it to the file to be printed.
5. It sends the font definition file and the source file (the file to be printed) to the PostScript printer.

Installing and Maintaining Host-Resident Fonts

Some fonts reside on the host system and are transmitted to the printer as needed for particular print requests. As the administrator, you make PostScript fonts available to all users on a system. To do so, you must know how and where to install these fonts. Because fonts are requested by name and stored in files, the LP print service keeps a map file that shows the correspondence between the names of fonts and the names of the files containing those fonts. Both the map and the font list must be updated when you install host-resident fonts.

The fonts available for use with PostScript printers are stored in directories you create called `/usr/share/lib/hostfontdir/typeface/font`, where *typeface* is replaced by a name like `palatino` or `helvetica`, and *font* is replaced by a name like `bold` or `italic`.

Managing Character Sets, Filters, Forms, and Fonts

This section provides step-by-step instructions for performing tasks related to setting up and administering character sets, print filters, forms, and fonts. For many tasks, you will find an example of user input and system output after the instructions.

▼ How to Set Up Hardware-Mounted or Selectable Character Sets

To set up print wheels or font cartridges:

1. **On the print server, become root or lp.**
2. **Type** `lpadmin -p printer-name -S hardware-charset-name1,hardware-charset-name2,...` **and press Return.**
This command defines which print wheels or cartridges can be used with the specified printer. The named print wheels or cartridges are registered with the LP print service. Do not put spaces in the comma-separated list. Define names that are meaningful to users, and inform users of the names.
3. **Specify how you want to be alerted to mount the print wheel.**
See “How to Mount, Unmount, and Set Alerts for Hard Character Sets” on page 138.
4. **On each print client, become root or lp.**
5. **Type** `lpadmin -p printer-name -S hardware-charset-name1,hardware-charset-name2...` **and press Return.**
This command defines which print wheels or cartridges can be used with the specified printer. The named print wheels or cartridges are registered with the LP print service. Do not put spaces in the comma-separated list. Define names that are meaningful to users, and inform users of the names.

To set up selectable character sets:

1. **On the print server, become root or lp.**
2. **Type** `tput -T printer-type csnm n` **and press Return.**
The name of the character set as known to the `terminfo` database is displayed. The printer type must match an entry in the `terminfo` database. See “Printer Entries in the `terminfo` Database” on page 84.

3. Repeat the `tput` command, typing 0 as the number you substitute for n and increasing the numbers by one (0, 1, 2, 3, 4, 5 and so on) until no more character sets are displayed.

You must determine which character sets are available in the `terminfo` database before you can name the character sets to the LP print service.

```
# tput -T ln03 csnm 0
usascii# tput -T ln03 csnm 1
english# tput -T ln03 csnm 2
finnish# tput -T ln03 csnm 3
japanese# tput -T ln03 csnm 4
norwegian# tput -T ln03 csnm 5
swedish# tput -T ln03 csnm 6
germanic# tput -T ln03 csnm 7
french# tput -T ln03 csnm 8
canadian_french# tput -T ln03 csnm 9
italian# tput -T ln03 csnm 10
spanish# tput -T ln03 csnm 11
line# tput -T ln03 csnm 12
security# tput -T ln03 csnm 13
ebcdic# tput -T ln03 csnm 14
apl# tput -T ln03 csnm 15
mosaic# tput -T ln03 csnm 16
#
```

4. (Recommended) On both the print server and print clients,

**type `lpadmin -p printer-name \`
`-S charset-namen=aliasn,charset-namem=aliasm...` and press Return.**

The named character sets are given the specified aliases. Do not put spaces in the comma-separated list. Character sets not given aliases in this command must be referenced by their `terminfo` names, or numbers with the `cs` prefix (for example, `cs1`, `cs2`). Aliases can be used in addition to the `terminfo` names and `cs` numbers.

```
# lpadmin -p slw2 -S usascii=text,line=printouts
```

Note – You do not need to perform this step to provide a list of aliases for the selectable character sets if the `terminfo` names are adequate.

▼ How to Mount, Unmount, and Set Alerts for Hard Character Sets

You must be root or lp to perform the following procedures.

To mount a print wheel or cartridge:

1. Put the print wheel or cartridge on the printer.

2. Type `lpadmin -p printer-name -M -S hardware-charset-name` and press Return.

Notifies the LP print service that the print wheel or cartridge is mounted. The mounted print wheel remains active until it is unmounted or until a new print wheel is mounted.

For example, to mount the print wheel named `pica` on `slw2`:

```
# lpadmin -p slw2 -M -S pica
```

To unmount a print wheel or cartridge:

♦ Type `lpadmin -p printer-name -M -S none` and press Return.

No print wheels or cartridges are mounted. You unmount print wheels or cartridges by mounting “none” of the character sets. For example:

```
# lpadmin -p slw2 -M -S none
```

To specify alerts to mount a print wheel or cartridge:

◆ **Type** `lpadmin -S hardware-charset-name -A type\
-Q requests -W minutes` and press **Return**.

Use Table 5-1 to choose the type of alert. If you do not specify what type of alert to give for print wheels, the default `none` is used.

Table 5-1 Types of Hardware-Mounted Character Set Alerts

Type	Description
'mail [username]'	Send the alert message by email to the administrator (<code>root</code> or <code>lp</code> , depending on who ran <code>lpadmin</code>), or the specified user.
'write [username]'	Send the alert message to the system console window of the current user or the specified user.
'command'	Run the <i>command</i> file for each alert. The environment variables and current directory are saved and restored when the file is executed.
none	Do not send any alert for the specified print wheel or cartridge (a useful option for the default or a permanently hardware-mounted character set).

For a specified print wheel or cartridge, this command asks for notification, according to the desired alert type, when the print queue contains a specific number of requests (`-Q`) needing that print wheel or cartridge. This command also specifies the number of minutes (`-W`) between alert reminders. You do not specify the alert message. It is a predefined message that says to mount the named print wheel or cartridge, and names one or more printers that have been set up to use such a print wheel or cartridge.

To stop the alerts for the current print wheel or cartridge:

◆ **Type** `lpadmin -S hardware-charset-name -A quiet` and press **Return**.

When you start receiving repeated alerts, this command suppresses the alerts until the print wheel or cartridge has been mounted and unmounted again, and the number of requests exceeds the `-Q` threshold.

Note – If *hardware-charset-name* is all in any of the previous commands, the alerting condition applies to all print wheels or cartridges for which an alert already has been defined. If you do not define an alert method for a print wheel or cartridge, you will not receive an alert to mount it. If you do define a method without the `-w` option, you will be alerted once for each occasion.

Examples of Setting Mount Alerts for Print Wheels or Cartridges

To receive email alerts every five minutes for a print wheel named `elite` when there are 10 requests for `elite` in the queue:

```
# lpadmin -S elite -A mail -Q 10 -W 5
```

To receive email alerts every minute for a cartridge named `finnish` when there are five requests for `finnish` in the queue:

```
# lpadmin -S finnish -A mail -Q 5 -W 1
```

To receive message alerts every 10 minutes to the console window for a print wheel named `elite`, when there are five requests for `elite` in the queue:

```
# lpadmin -S elite -A write -Q 5 -W 10
```

To receive no alerts for a print wheel named `elite`:

```
# lpadmin -S elite -A none
```

▼ How to Add, Change, Remove, or Restore Print Filters

You must be root or lp to perform the following procedures, which apply to *either* the print server or the print client.

To add a print filter using a definition from a file:

- ◆ **To use a file, type `lpfilter -f filter-name -F filename.fd` and press Return.**

The print filter definition from the file named is added to the `/etc/lp/filter.table` file using the filter name you specify.

```
# lpfilter -f daisytroff -F /etc/lp/fd/daisytroff.fd
```

To add a print filter from the command line:

- ◆ **Type `lpfilter -f filter-name - filter-definition` and press Control-d.**
The filter definition you type is added to the `/etc/lp/filter.table` file using the filter name you specify. The print filter definition has the following format:

```
Command: command-pathname [options]  
Input types: input-type-list  
Output types: output-type-list  
Printer types: printer-type-list  
Printers: printer-list  
Filter type: fast or slow  
Options: template-list
```

To change a print filter:

- ◆ **Follow the steps in the previous procedure for adding a filter, except include only the changed items.**
The print filter definition in `/etc/lp/filter.table` is changed. You need to provide information only for those items that must be changed; items for which you do not specify new information remain the same.

To remove a print filter:

- ◆ **Type `lpfilter -f filter-name -x` and press Return.**
The filter entry is removed from the `/etc/lp/filter.table` file. You can remove a print filter from a print client, and not remove it from the print server, so other print clients can still use the print filter.

```
# lpfilter -f daisytroff -x
```

To restore the LP print service to its original set of default print filters:

- ◆ To restore all filters, type `lpfilter -f all -i` and press Return. To restore a specific filter, type `lpfilter -f filter-name -i` and press Return.

▼ How to View Print Filter Definitions

You must be either root or lp to perform the following procedures, which apply to *either* the print server or the print client.

To view a print filter definition:

- ◆ Type `lpfilter -f filter-name -l` and press Return. The definition of the print filter is displayed. Use `all` for *filter-name* to see the definitions of all installed print filters.

```
# lpfilter -f postdaisy -l
Input types: daisy
Output types: postscript
Printer types: any
Printers: any
Filter type: slow
Command: /usr/lib/lp/postscript/postdaisy
Options: PAGES * = -o*
Options: COPIES * = -c*
Options: MODES group = -n2
Options: MODES group\=\([2-9]\) = -n\1
Options: MODES portrait = -pp
Options: MODES landscape = -pl
Options: MODES x\=\(\-[0-9]*\) = -x\1
Options: MODES y\=\(\-[0-9]*\) = -y\1
Options: MODES magnify\=\([0-9]*\) = -m\1
```

To redirect the definition of the print filter to a file:

- ◆ Type `lpfilter -f filter-name -l > filename` and press Return. The output from the command is saved in the file name that you specify. You may want to save the output in a file that you can use later to redefine the print filter if you inadvertently remove the filter from the LP print service.

To check which filters are installed:

- ◆ **Type** `lpfilter -f all -l | grep Filter` **and press Return.**
This command gives you a list of all the filters and their types installed in the `/etc/lp/filter.table` file.

```
# lpfilter -f all -l | grep Filter
(Filter "download")
Filter type: fast
(Filter "postio")
Filter type: fast
(Filter "postior")
Filter type: fast
(Filter "postreverse")
Filter type: slow
(Filter "postdaisy")
Filter type: slow
```

▼ How to Add, Change, or Remove a Form

You must be either root or lp to perform the following procedures, which apply to *either* the print server or the print client.

To add a form using a definition from a file:

- ◆ **Type** `lpforms -f form-name -F filename` **and press Return.**
The form definition from the file named is added to the `/etc/lp/forms/form-name` directory.

```
terra# lpforms -f medical -F /etc/lp/forms/medical.fmd
```

To add a form from the command line:

- ◆ **Type** `lpforms -f form-name - form-definition` and **press Control-d**.
The form definition is added to the `/etc/lp/forms/form-name` directory.
The form definition has the following format:

```
Page length: scaled number
Page width: scaled number
Number of pages: integer
Line pitch: scaled number
Character pitch: scaled number
Character set choice: character-set-name [,mandatory]
Ribbon color: ribbon-color
Comment:
informal notes about the form
Alignment pattern: [content-type] alignment pattern
```

Note – Before the form can be used, one or more printers must be granted access to the form. See “How to Allow or Deny Use of Forms on a Printer” on page 150.

To change a form:

- ◆ **Follow the steps in the previous procedure for adding a form, except include only the changed items.**
You need to provide information only for those items that must be changed; items for which you do not specify new information remain the same.

To remove a form:

- ◆ **Type** `lpforms -f form-name -x` and **press Return**.
The `/etc/lp/forms/form-name` directory is removed from the system. You can remove a form from a print client, and not remove it from the print server, so other print clients can still use the form.

```
# lpforms -f medical -x
```


▼ How to Mount, Unmount, and Specify Alerts for Forms

You must be either root or lp to perform the following procedures, which apply only to the print server.

To mount a form:

1. Type `disable printer-name` and press Return.

This disables the printer so new print jobs (which may not be appropriate for the form) are not allowed to be queued to the printer.

2. Load the form paper into the printer.

3. Mount the form by typing

```
lpadmin -p printer-name -M -f form-name -a -o filebreak
```

and pressing Return.

When you mount a form that has an alignment pattern defined for it, and you include the `-a -o filebreak` option following the form name, you are instructed to press Return to print a copy of the alignment pattern. After the pattern is printed, you can adjust the printer and press Return again. If the form does not have an alignment pattern, or you do not want to use it, omit the `-a -o filebreak` from the command.

4. Type `enable printer-name` and press Return.

Now the printer is ready to print on the newly mounted form.

```
terra# disable luna
printer "luna" disabled
terra# lpadmin -p luna -M f medical -a -o filebreak
terra# enable luna
printer "luna" enabled
```

To unmount a form:

1. Type `disable -p printer-name` and press Return.

The printer is disabled.

2. Type `lpadmin -p printer-name -M -f none` and press Return.

The form is unmounted from the LP print service.

3. Remove the form paper from the printer.

After you remove the form paper, add the appropriate paper for the next print job.

4. Type `enable -p printer-name` and press Return.

The printer is enabled and ready to print.

```
terra# disable luna
printer "luna" disabled
terra# lpadmin -p luna -M f none
terra# enable luna
printer "luna" enabled
```

To specify alerts for mounting forms:

◆ **Type `lpforms -f form-name -A 'type' -Q requests -W minutes` and press Return.**

Use Table 5-2 to choose the type of alert you want to receive when print requests require a form to be mounted. If you do not define an alert, you will not receive any.

Table 5-2 Types of Form-Mounting Alerts

Type	Description
'mail [username]'	Send the alert message by email to the administrator (root or lp, depending on who ran lpforms), or to the specified user.
'write [username]'	Send the alert message written to the system console window of the current user, or to the specified user.
'command'	Run the <i>command</i> file for each alert. The environment variables and current directory are saved and restored when the file is executed.
none	Do not send any alert for the specified form (a useful option for the default form or one that is mounted permanently).

For a specified form, the lpforms command asks for notification, according to the desired alert type, when the print queue contains a specific number of requests (-Q) needing that form. This command specifies the number of minutes (-W) between alerts. You do not specify the alert message. It is a predefined message that says to mount the named form, and names one or more printers that have been set up to use such a form.

To stop alerts for the current form:

◆ **Type** `lpforms -f form-name -A quiet` **and press Return.**

When you start receiving repeated alerts, this suppresses them for the specified form until the form has been mounted and unmounted again, and the number of requests exceeds the `-Q` threshold.

Note – If *form-name* is `all` in any of the previous commands, the alerting condition applies to all forms for which an alert already has been defined. If you do not define an alert method for a form, you will not receive an alert to mount it. If you define a method without the `-W` option, you will be alerted once for each occasion.

Examples of Specifying Alerts for Mounting Forms

To receive email alerts on the print server every five minutes for a form named `letterhead`, when there are 10 requests for `letterhead` in the queue:

```
# lpforms -f letterhead -A mail -Q 10 -W 5
```

To receive message alerts to the console window of the print server every 10 minutes when there are 5 requests for `letterhead` in the queue:

```
# lpforms -f letterhead -A write -Q 5 -W 10
```

To receive no alerts for a form named `letterhead`:

```
# lpforms -f letterhead -A none
```

▼ How to Obtain Information About Forms

The following procedures apply to either the print server or the print client.

To review the definition of a form:

1. **Become root or lp.**

2. Type `lpforms -f form-names -l` and press Return.

The definition of the forms is displayed. If you specify multiple forms, separate names with commas, or use spaces and enclose the list in double quotation marks. You can use `all` instead of *form-names* to review all forms.

```
# lpforms -f medical -l
Page length: 62
Page width: 72
Number of pages: 2
Line pitch: 6
Character pitch: 12
Character set choice: pica
Ribbon color: black
Comment:
Prudential medical claim form
```

To redirect the definition of the form to a file:**1. Become root or lp.****2. Type `lpforms -f form-name -l > filename` and press Return.**

The output from the command is saved in the file name that you specify. This is useful for saving the form definition, which you could use to change the definition or restore it, if it gets removed unintentionally.

To show the current status of forms:**◆ Type `lpstat -f form-names` and press Return.**

The status of the forms is displayed. If you specify multiple forms, separate names with commas, or use spaces and enclose the list in quotation marks. You can use `all` instead of *form-names* to review all forms.

```
# lpstat -f medical,payroll
form medical is available to you
form payroll is available to you
```

To show a form's current status and definition:

- ◆ **Type** `lpstat -f form-name -l` **and press Return.**

The definition of the form is displayed. If you specify multiple forms, separate names with commas, or use spaces and enclose the list in quotation marks. You can use `all` instead of `form-name` to review all forms.

```
# lpstat -f medical -l
form medical is available to you
  Page length: 62
  Page width: 72
  Number of pages: 2
  Line pitch: 6
  Character pitch: 12
  Character set choice: pica
  Ribbon color: black
  Comment:
  Prudential medical claim form
```

▼ How to Allow or Deny User Access to Forms

You must be either root or lp to perform the following procedures, which apply to *both* the print server or the print client.

To allow users access to forms:

- ◆ **Type** `lpforms -f form-name -u allow:user-list` **and press Return.**

Separate user names with commas but no spaces. Or, you can use spaces to separate names, enclosing the word `allow:` and the names in quotation marks.

```
# lpforms -f check -u allow:ignatz,ziggy
```

To deny users access to forms:

- ◆ **Type** `lpforms -f form-name -u deny:user-list` **and press Return.**
Separate user names with commas but no spaces. If you use spaces to separate names, enclose the word `deny:` and the names in quotation marks.

```
# lpforms -f dental -u deny:jones,smith
```

To allow all users access to forms:

- ◆ **Type** `lpforms -f form-name -u allow:all` **and press Return.**

```
# lpforms -f check -u allow:all
```

To deny all users access to forms:

- ◆ **Type** `lpforms -f form-name -u deny:all` **and press Return.**

```
# lpforms -f check -u deny:all
```

▼ **How to Allow or Deny Use of Forms on a Printer**

You must be either root or lp to perform the following procedures, which apply to *both* the print server and print clients.

To allow a printer to access forms:

- ◆ **Type** `lpadmin -p printer-name -f allow:form-list` **and press Return.**
The printer now has access to the specified forms. Separate form names with commas but no spaces. Or, you can use spaces to separate names, enclosing the word `allow:` and the list of forms in quotation marks. Replace *form list* with `all` to allow access to all forms.

```
# lpadmin -p luna -f allow:medical,dental,check
```

To deny a printer access to forms:

- ◆ **Type** `lpadmin -p printer-name -f deny:form-list` and press **Return**.
Separate form names with commas but no spaces. Or, you can use spaces to separate names, enclosing the word `deny:` and the list of forms in quotation marks. Replace *form-list* with `all` to deny access to all forms.

```
terra# lpadmin -p luna -f deny:medical,dental,payroll
```

Note – If you do not use `allow` or `deny` options, the LP print service denies the use of all forms. You can, however, mount any form, even if it is denied, making it available for use.

▼ How to Install Downloaded and Host-Resident PostScript Fonts

The following procedures apply to *both* the print server and print clients.

To install downloaded fonts:

1. **Type** `cd /etc/lp/printers/printer-name` and press **Return**.
This accesses the directory that contains the configuration information for the printer.
2. **Edit the file** `residentfonts`, using a text editor like `vi`.
The file may not exist, if this is the first time you are adding permanently downloaded fonts.
3. **List all printer-resident fonts and fonts to be permanently downloaded.**

To install host-resident PostScript fonts, on the server:

1. **Become root or lp.**
2. **(If necessary) Create the** `hostfontdir` **directory by typing**
`cd /usr/share/lib;mkdir hostfontdir; chmod 775 hostfontdir`
and pressing Return.
3. **(If necessary) Type** `mkdir typeface` **and press Return.**
Create a directory for a new typeface, if one does not exist already.

4. **Type `cp filename /usr/share/lib/hostfontdir/typeface/font` and press Return.**

Copy the font file to the appropriate directory.

5. **Add the name of the font and the name of the file in which it resides to the map table.**

- a. **Type `cd /usr/share/lib/hostfontdir` and press Return.**

- b. **Edit the `map` file using a text editor like `vi`.**

- c. **Add a one-line entry for each font you want to add to the table, putting the font name first, followed by a space, followed by the name of the file (copied in step 4) where the font resides. For example:**

```
Palatino-Bold /usr/share/lib/hostfontdir/palatino/bold
```

- d. **Save the file.**

For example, once the example entry exists in the map table on the appropriate system, users will be able to apply the Palatino Bold font in their print jobs. When they submit a print request containing this font, the LP print service appends a copy of the file

`/usr/share/lib/hostfontdir/palatino/bold` to that file before sending it to the printer.

6. **If you are using `troff`, you must create new width tables for this font in the standard `troff` font directory.**

Setting Printing Policies

6

This chapter describes how to change LP print service defaults to set printing policies appropriate for your site.

If you want to skip the background information that explains the concepts of setting printing policies, and proceed directly to step-by-step instructions, use the following table to find the page where the instructions for a specific task begin.

<i>How to Set the System Printer Destination</i>	<i>page 162</i>
<i>How to Control Printing of Banner Pages</i>	<i>page 163</i>
<i>How to Control Access to enable and disable Commands</i>	<i>page 163</i>
<i>How to Specify Alerts for Printer Faults</i>	<i>page 164</i>
<i>How to Specify Fault Recovery</i>	<i>page 165</i>
<i>How to Define a Class of Printers</i>	<i>page 166</i>
<i>How to Add a Printer Description</i>	<i>page 166</i>
<i>How to Set Default Priority and Priority Limits for Users</i>	<i>page 167</i>
<i>How to Allow or Deny Users Access to a Printer</i>	<i>page 168</i>

If you want to review background information first, read the following section, “About Setting Printing Policies” on page 154.

You may want to refer to the following chapters and appendixes for additional information about printing:

- See Chapter 3, “Setting Up Printers,” for information about setting up basic printing services using Printer Manager, an Administration Tool application.
- See Chapter 4, “Routine Printer Administration,” for information about how to administer printing services, once printers have been set up.
- See Chapter 5, “Managing Character Sets, Filters, Forms, and Fonts,” for additional tasks specific to different types of printers.
- See Chapter 7, “Troubleshooting Printing Problems,” if you have trouble setting up or administering printing.
- See Appendix A, “Setting Up Printer Services Using the Command-Line Interface,” for information about setting up basic printing services using the LP print service’s command-line interface.
- See Appendix B, “Understanding and Customizing the LP Print Service,” for a description of the LP print service and how to customize various aspects of it.

About Setting Printing Policies

In addition to, or as part of, setting up printing services at your site, the LP print service offers several options you may want to consider when setting the printing policies at your site.

Controlling the System Printer Destination

You should provide a printer destination for each print client system. There are two ways you can set the default printer destination:

- Set the `LPDEST` environment variable.
- Set a printer destination for the LP print service.

Some applications attempt to control the printer destination for print requests that they originate.

When an application provides a printer destination, that destination is used by the LP print service, regardless of whether the LP print service has a destination or the `LPDEST` environment variable is set.

When a destination is specified for the LP print service, it is used when no printer name is included as part of the print request, and the `LPDEST` environment variable is *not* set.

Setting Banner Page Policy

A *banner page* identifies the user who submitted the print request, the print request ID, and when the request was printed. A banner page can also have an optional title that the requester can use to better identify a printout.

Banner pages make identifying the owner of a print job easy, especially when many users submit jobs to the same printer. Printing banner pages uses more paper, however, and may not be necessary if a printer has only a few users. In some cases, printing banner pages is undesirable. For example, if a printer has special paper or forms mounted, like paycheck forms, printing banner pages may cause problems.

By default the LP print service forces banner pages to be printed, regardless of whether or not the user specifies the `nobanner` option. You can change this policy to allow users to turn off the printing of a banner page when they make a print request.

Permitting Users to Enable or Disable a Printer

You must also decide whether users are allowed to enable or disable printers. The `enable` and `disable` commands, by default, are owned by `lp`, and may only be run by `lp` or `root`. You can, however, give users the ability to control access to printers with the `enable` and `disable` commands.

If your organization decentralizes responsibility for printer operations, you probably want to permit users access to the `enable` and `disable` commands. If you want to control printers and restrict others from interfering with print service operation, then do not change the permissions that restrict access to these commands.

You can also change the permissions on other LP administrative commands to grant users access to them.

Setting Printer Fault Alerts

You can choose one of several ways to receive an alert about a printer fault, including:

- Electronic mail
- Message written to your login terminal (usually system console window)
- Message specified by a program of your choice
- No alerts

The default is to write a message to the terminal to which you (root) are logged in. Unless you specify a program to deliver alerts, the content of the alert is predefined.

The LP print service provides a way to detect printer faults and alert the system administrator to them. Some faults are as simple as running out of paper or needing to replace a toner cartridge. Other more serious problems may include complete printer failure or power failure.

The LP print service recognizes only two fault indicators:

- A drop in carrier
- An XOFF not followed in reasonable time by an XON

Print filters recognize other printer fault indicators, and rely on the LP print service to alert you to a fault when the filter detects it.

Note – If you choose to receive no alerts, you need a way to find out about printing faults, so you can correct the problem. The LP print service will not continue to use a printer that has a fault.

Using the `lpadmin -A` command, you can control the following aspects of printer fault alerts:

- Choose the method by which alerts are sent
 - By email, using the `-A mail` option
 - By message to the console window, using the `-A write` option
 - By message specified by a program, using the `-A 'command'` option
 - No alerts, using the `-A none` option
- Choose to receive repeated alerts every few minutes until the fault is cleared, using the `-W minutes` option

Note – You can specify which user receives email or message alerts. If you choose to receive email or a message written to your console window, the mail or message is sent to the current ID, which may not be your login ID if you use the `su` command to change IDs. You can set alerts for these parts of the LP print service:

- Printer faults
- Mounting forms
- Mounting print wheels

Specifying Printer Fault Recovery

As part of printer fault handling, the LP print service sometimes disables those printers for which it encounters faults. When the printer is ready for printing again, the LP print service recovers in one of three ways:

- It restarts printing at the beginning of the print request that was active when the fault occurred.
- It waits for you to tell the LP print service to re-enable the printer.
- It continues printing at the top of the page where printing stopped.

Note – A print filter is required for the LP print service to be able to continue printing at the top of a page where printing stops. Such a filter tracks the control sequences used by the printer to track page boundaries and where in a file printing stopped. The default filters used by the LP print service cannot do this. You will be notified by the LP print service if recovery cannot proceed as you specify because of the filter. See Appendix B, “Understanding and Customizing the LP Print Service,” for information about writing filters.

You can tell the LP print service which way to recover from a printer fault. If you do not specify how the LP print service will recover after a printer fault is cleared—and if a suitable filter is provided—the print service tries to continue printing at the top of the page where printing stopped. If it cannot, the print service starts printing at the first page of the print request.

If you specify `continue` printing from the top of the page for failure recovery, but the interface program does not continue running so that it can detect when the printer fault has been cleared, printing is attempted every few minutes until it succeeds. You can force the LP print service to retry immediately by issuing an `enable` command.

Defining Classes of Printers

The LP print service allows you to group several local printers into one class. This allows a user to specify a class (rather than individual printers) as the destination for a print request. The first printer in the class that is free to print is used. The result is faster turnaround, as printers are kept as busy as possible.

There are no default printer classes known to the LP print service; printer classes exist only if you define them. You are not obligated, however, to define printer classes. You should add them only if you determine that using printer classes will benefit the network users.

Here are some ways you could define printer classes:

- By printer type (for example, PostScript)
- By location (for example, 5th floor)
- By work group or department (for example, Accounting)

Alternatively, a class might contain a group of printers that are used in a particular order. The LP print service always checks for an available printer in the order that printers were added to a class. Therefore, if you want a high-speed printer to be accessed first, you would add it to the class before you add a low-speed printer. As a result, the high-speed printer would handle as many print requests as possible. The low-speed printer would be reserved as a backup printer when the high-speed printer is in use.

Note – Print requests are balanced between printers in a class only for local printers. When a print client attempts to print to a class of printers on a print server, only the first printer defined in the class is used.

Class names, like printer names, must be unique and may contain a maximum of 14 alphanumeric characters and underscores.

Adding a Printer Description

You can add a descriptive message for each printer to the LP print service to help network users identify printers. The description can contain any message you like. You might include the room number where the printer is located and the name of the person to call if there are printing problems. The message is displayed by using the `lpstat -D -p printer-name` command.

Setting Priority Limits

You can set a default queue priority for all print requests, set queue priority limits for each user, and adjust the queue priority for individual print requests.

The LP print service provides a simple priority mechanism so that users and administrators can adjust the position of a print request in the queue. Users can ask for a priority level—a number from 0 to 39—when they submit a print request. The lower the number, the higher the priority level. Requests with higher priority are placed ahead of requests with lower priority.

You can adjust the priority levels of users in the following ways:

- Assign each user a priority limit. Users cannot submit a print request with a priority higher than the assigned limit, although they can submit a request with a lower priority.
- Assign a default priority limit for the users not assigned a personal limit.

You can also set a default priority, which is automatically given requests to which users do not assign a priority. The LP print service automatically assigns every print request a priority of 20.

By setting the priorities according to your needs, you can prevent lower-priority tasks from interfering with higher-priority printing tasks.

You can also assign “immediate” or “hold” priorities if you need to pre-empt the job currently printing.

Creating allow and deny Lists

You can control users' access to printers and forms by creating `allow` and `deny` lists. An `allow` list contains the names of users granted access to the specified printer or form; a `deny` list contains the names of users denied access to the specified printer or form.

This method of allowing or denying access to printers and forms is similar to the method that allows or denies access to the `cron` and `at` facilities.

The rules for `allow` and `deny` lists are:

- When you do not create an `allow` or `deny` list, all users may use the printer or form.
- When both `allow` and `deny` lists are empty, there are no restrictions on who may use the printer or form.
- Specifying `all` in the `allow` list allows all users access to the printer or to print on the form. Specifying `all` in the `deny` list denies access to all users except `lp` and `root`.
- When the `allow` list contains entries, only those users who are listed can access the printer or print on the form. The `deny` list is ignored.
- When the `allow` list is empty or does not exist, users who are listed in the `deny` list are not allowed access to the printer or to print on the form.

Each item in the *user-list* can take any form shown in Table 6-1. Separate each item by either a space or a comma. If you use spaces to separate the names, enclose the entire list (including the `allow:` or `deny:`) in quotation marks.

Table 6-1 Syntax for `allow` or `deny` User Lists

Item	Description
<code>user</code>	User on any system
<code>all</code>	All users on all systems
<code>system!user</code>	User on <i>system</i> only
<code>!user</code>	User on local system only
<code>all!user</code>	User on any system

Table 6-1 Syntax for `allow` or `deny` User Lists (Continued)

Item	Description
<code>all!all</code>	All users on all systems
<code>system!all</code>	All users on <i>system</i>
<code>!all</code>	All users on local system

Note – Make sure that the `allow` and `deny` lists for print servers and their print clients match. If the two sets of lists do not match, users may receive conflicting messages about a printer accepting jobs and then refusing jobs.

You can also create `allow` and `deny` lists to control a certain printer's access to specific forms. In that case, the lists contain form names rather than user names.

Controlling User Access to Printers

You can control which users on any system can have access to some or all of the available printers. For example, you may want to prevent some users from printing on a high-quality printer to minimize expense. To restrict user access to printers, you can create `allow` and `deny` lists (using `lpadmin`; the Printer Manager enables you to create only `allow` lists). If you create neither, all users have access.

If you use `allow` access lists on print clients and print servers, make sure the lists match—because after an `allow` list on a system limits access to only the users in the list. If a user is in the `allow` list on the print client but not in the `allow` list on the print server, his or her print request is allowed on the client, but when it gets to the print server, the request is denied. If, on the other hand, a user is denied permission to use a printer on a print server, you do not need to include that user in the `deny` list on the print client system. Users in the print server `deny` list are denied access to the remote printer.

As a courtesy to users, however, you might want to make the `deny` lists for print servers and clients match. In that way, you can be sure that when client users try to access a printer that they are not authorized to use, an immediate message is displayed saying that permission to use the printer is denied.

However, you have to decide whether it is worth it—maintaining matching lists on multiple systems—to achieve the benefit of getting messages to users faster. On the downside, the messages may be wrong if the access lists change over time. If you create access lists only on the print server, it is much easier to administer. This policy has been adopted by the Printer Manager, which allows you to create and maintain only an `allow` access list on the print server. The inability to create a `deny` list may be inconvenient at times—like when you want to exclude a few out of many users—but the simplicity of dealing with only one type of access list is worth it.

Instructions for Setting Printing Policies

This section provides step-by-step instructions for performing tasks related to setting printing policies. For many tasks, you will find an example of user input and system output after the instructions.

▼ How to Set the System Printer Destination

1. On the print server or print client, become root or lp.

2. Type `lpadmin -d printer-name or class-name` and press Return.

The printer you specify is established as the default printer for the system. The printer can reside on a remote print server. You should perform this step during initial printer setup, but you can set or change the default print request destination later. When a user does not specify the destination for a print request, the `LPDEST` environment variable is checked. If that fails, then the default printer for the system defined in this step is used.

▼ How to Control Printing of Banner Pages

♦ **Type `lpadmin -p printer-name -o nobanner or banner` and press Return.**

The default is the `banner` option, which forces a banner page to print with every print request, even if users ask for no banner page. When you specify `-o nobanner`, users are allowed to ask for no banner page (`lp -o nobanner`), and the request is honored.

▼ How to Control Access to enable and disable Commands

By default, you must become root or lp to use the LP administrative commands.

To give all users permission to run enable and disable commands:

1. On the print server, type `ls -l /usr/bin/enable\
/usr/bin/disable` **and press Return.**
Check that lp is the owner of these commands.

2. If lp does not own the commands, type
`chown lp /usr/bin/enable /usr/bin/disable` **and press Return.**
lp now owns the commands.

3. Type `chmod u+s /usr/bin/enable /usr/bin/disable` **and press Return.**
The set user ID bit is set so that users can access the commands.

To prevent others from running enable and disable commands:

♦ **Type** `chmod u-s /usr/bin/enable /usr/bin/disable` **and press Return.**
Clearing the set user ID bit prevents users other than root or lp from running the enable and disable commands.

Note – You can also change the permissions on other LP administrative commands to grant or deny users access to them.

▼ How to Specify Alerts for Printer Faults

To specify alerts for printer faults:

- ◆ **Type** `lpadmin -p printer-name -A type -W minutes` and press **Return**.
Use the table below to choose the *type* of alert. If you do not specify the *type* of alert to give for printer faults, a message will be written once to the terminal on which you (root) are logged in.

Table 6-2 Types of Printer Fault Alerts

Type	Description
'mail [username]'	Send the alert message by email to the administrator (root or lp, depending on who ran lpadmin), or the specified user.
'write [username]'	Send the alert message to the system console window of the current user, or the specified user.
'command'	Run the <i>command</i> file for each alert. The environment variables and current directory are saved and restored when the file is executed.
none	Do not send any messages.

This command asks for notification. The notification is sent according to the desired alert type and number of minutes specified between alert reminders. If you omit `-W`, the message is sent once. You do not specify the alert message. It is a predefined message that says the printer has stopped printing and needs to be fixed.

To stop alerts for the current printer fault:

- ◆ **On the print server, type** `lpadmin -p printer-name -A quiet` and press **Return**.
When you (root or specified user) receive repeated alerts, this command suppresses alerts for the specified printer until the printer fault has been fixed, and the printer is enabled.

Note – If *printer-name* is `all` in any of the previous commands, the alert condition applies to all printers on the system.

Examples of Specifying Alerts for Printer Faults

To send email alerts to a user named `joe` for a printer named `mars`, with reminders every 5 minutes:

```
# lpadmin -p mars -A 'mail joe' -W 5
```

To receive message alerts at the terminal on which `root` is logged in (console window) for a printer named `venus`, with reminders every 10 minutes:

```
# lpadmin -p venus -A write -W 10
```

To receive no alerts for a printer named `mercury`:

```
# lpadmin -p mercury -A none
```

▼ How to Specify Fault Recovery

1. On the server, become `root` or `lp`.
2. Type `lpadmin -p printer-name -F fault-recovery` and press **Return**.
Use the table below to choose the type of recovery you want for any print request stopped because of a printer fault.

Table 6-3 Types of Printer Fault Recoveries

Type	Description
<code>beginning</code>	Start printing the request again from the beginning.
<code>continue</code>	Continue printing the request from the top of the page where printing stopped. This requires a filter to wait for the fault to be cleared before automatically continuing.
<code>wait</code>	Disable printing on the printer, and wait until it is re-enabled. Resume printing at the top of the page where the request stopped, unless <code>lp -i</code> is used to specify where printing should resume.

Re-enable the printer after you clear a fault, so that printing can resume immediately.

▼ How to Define a Class of Printers

Note – Print requests are balanced between printers in a class only for local printers. When a print client attempts to print to a class of printers on a print server, only the first printer defined in the class is used.

1. **On the print server, become root or lp.**
2. **Type `lpadmin -p printer-name -c printer-class` and press Return.**
The printer you named is added to the end of the list in the class you named. If the printer class does not exist, it is created.

```
# lpadmin -p slw2 -c roughdrafts
```

▼ How to Add a Printer Description

1. **On both the print server and print clients, become root or lp.**
2. **Type `lpadmin -p printer-name -D "comment"` and press Return.**
The *comment* message describes characteristics of the printer, like location or administrative contact. Use single quotation marks if the message contains characters that the shell might interpret (like *, ?, \, !, ^).

```
# lpadmin -p slw2 -D "Laser across from Ken's office"
```

▼ How to Set Default Priority and Priority Limits for Users

1. On both the print server and print clients, become root or lp.
2. Type `lpusers [option]` and press Return.
Use the table below to choose the option you want.

Table 6-4 Options for Setting User Priority Limits

Option	Description
<code>-d level</code>	Sets the system-wide priority level for print requests. This is the priority level a user's print request is given when the user does not explicitly set the request's priority. (Default is 20.)
<code>-q level</code>	Sets the default highest priority level that applies to all users not explicitly assigned an upper limit with <code>-q</code> and <code>-u</code> options combined.
<code>-u user-list</code>	Removes the priority limit set for the specified users. If you have set a default limit, it now applies.
<code>-q level -u user-list</code>	Sets the highest priority level at which the specified users can submit print requests. Otherwise, the highest priority is 0. (Priority runs from 0, highest, to 39, lowest.) Type a list of user names separated by commas or enclose the list in quotation marks. You can qualify user names as follows: <i>system-name!user-name</i> - Named user on named system <i>system-name!all</i> - All users on named system <i>all!user-name</i> - Named user on all systems <i>all</i> - All users on all systems
<code>-l</code>	Lists the default priority level, and the priority limit explicitly assigned to users.

Note - If the default priority is higher than the limit for the user, the user's limit is applied instead. For example, if the default priority is 20 and the user's priority limit is 18, a print job for that user will have a priority of 20. Users can always specify a lower priority than their priority limit. For example, if the user's priority limit is 18, that user can submit a job with a priority of 25, but not one with a priority of 12. Remember, the lower the number (from 0–39), the higher the priority.

▼ How to Allow or Deny Users Access to a Printer

To allow users access to a printer:

1. On the print server, become root or lp.
2. Type `lpadmin -p printer-name -u allow:user-list or all` and press Return. By default, all users are allowed access to a printer. This step limits printer access to only those users listed. Separate user names with commas but no spaces. Or, you can use spaces to separate names, enclosing the word `allow:` and the list of users in quotation marks. You can substitute `all` for the *user-list*.

Suppose you want to allow only the users `ignatz` and `ziggy` access to a local printer `slw2` on system `jupiter` and a remote printer `luna` connected to the print server `terra`. You could type the following:

```

jupiter% su
Password:
jupiter# lpadmin -p slw2 -u allow:ignatz,ziggy
jupiter# lpadmin -p luna -u allow:ignatz,ziggy
jupiter# exit
jupiter% rlogin terra
terra% su
Password:
terra# lpadmin -p luna -u allow:ignatz,ziggy
```

To deny users access to a printer:

1. On both the print server and the print clients, become root or lp.
2. Type `lpadmin -p printer-name -u deny:user-list or all` and press Return. All users are allowed access to the printer, unless they are explicitly listed in the deny list. Separate user names with commas but no spaces. Or, you can use spaces to separate names, enclosing the word `deny:` and the list of users in quotation marks. You can substitute `all` for the *user-list*.

Suppose you want to deny the users `ignatz` and `ziggy` access to a local printer `slw2` on system `jupiter` and a remote printer `luna` on the print server `pine`. You would type:

```
jupiter% su
Password:
jupiter# lpadmin -p slw2 -u deny:ignatz,ziggy
jupiter# lpadmin -p luna -u deny:ignatz,ziggy
jupiter# exit
jupiter% rlogin terra
terra% su
Password:
terra# lpadmin -p luna -u deny:ignatz,ziggy
```


Troubleshooting Printing Problems



This chapter explains how to troubleshoot printing problems that may occur when you set up or maintain printing services.

If you want to skip the background information that explains the concepts of troubleshooting printing problems, and would rather proceed directly to step-by-step instructions, use the following table to find the page where the instructions for a specific task begin.

<i>How to Troubleshoot No Output</i>	<i>page 179</i>
<i>How to Troubleshoot Incorrect Output</i>	<i>page 199</i>
<i>How to Unhang the LP Commands</i>	<i>page 204</i>
<i>How to Troubleshoot an Idle (Hung) Printer</i>	<i>page 205</i>
<i>How to Resolve Conflicting Status Messages</i>	<i>page 207</i>

If you want background information related to the instructions given in this chapter, read the section “About Troubleshooting Printing Problems” on page 172.

You can refer to the following chapters and appendixes for additional information about printing:

- See Chapter 3, “Setting Up Printers,” for information about setting up basic printing services using Printer Manager, an Administration Tool application.
- See Chapter 4, “Routine Printer Administration,” for information about how to administer printing services, once the printers have been set up.

- See Chapter 5, “Managing Character Sets, Filters, Forms, and Fonts,” for tasks specific to different types of printers.
- See Chapter 6, “Setting Printing Policies,” for tasks to set printing policy, such as whether banner pages are required.
- See Appendix A, “Setting Up Printer Services Using the Command-Line Interface,” for information about setting up basic printing services using the LP print service’s command-line interface.
- See Appendix B, “Understanding and Customizing the LP Print Service,” for a description of the LP print service and how to customize various aspects of it.

About Troubleshooting Printing Problems

Sometimes after setting up a printer, you find that nothing prints. Or, you may get a little further: something prints, but it is not what you expect—the output is incorrect or illegible. Then when you get past these problems, other problems may occur, like:

- LP commands hanging
- Printers becoming idle
- Users getting conflicting messages

Note – Although many of the suggestions in this chapter are relevant to parallel printers, they are geared toward the more common serial printers.

Troubleshooting No Output (Nothing Prints)

When nothing prints, there are three basic areas to check:

- The printer hardware
- The network
- The LP print service

If you get a banner page, but nothing else, this is a special case of incorrect output. See “Troubleshooting Incorrect Output” on page 174.

Check the Hardware

The hardware is the first area to check. As obvious as it sounds, you should make sure the printer is plugged in and turned on. In addition, you should refer to the manufacturer's documentation for information about hardware settings. Some computers use hardware switches that change the characteristics of a printer port.

The printer hardware includes the printer, the cable that connects it to the computer, and the ports into which the cable plugs in at each end. As a general approach, you should work your way from the printer to the computer. Check the printer. Check where the cable connects to the printer. Check the cable. Check where the cable connects to the computer.

Check the Network

Problems are more common with remote jobs—those going from a print client to a print server. You should make sure that network access between the print server and print clients is enabled.

If the network is running the Network Information Service Plus (NIS+), see *Name Services Administration Guide* for instructions to enable access between systems. If the network is not running the Network Information Service (NIS) or NIS+, before you set up print servers and print clients, include the Internet address and system name for each client system in the `/etc/hosts` file on the print server. Also, the Internet address and system name for the print server must be included in the `/etc/hosts` file of each print client system.

Check the LP Print Service

For printing to work, the LP scheduler must be running on both the print server and print client. If it is not running, you need to start it using the `/usr/lib/lp/lpsched` command. If you have trouble starting the scheduler, see “How to Start or Stop the Print Service Scheduler” on page 120.

In addition to the scheduler running, a printer must be enabled and accepting requests before it will produce any output. If the LP print service is not accepting requests for a printer, the submitted jobs (print requests) are rejected. Usually, in that instance, the user receives a warning message when a job is submitted. If the LP print service is not enabled for a printer, jobs remain queued on the system until the printer is enabled.

In general, you should analyze a printing problem as follows:

- Follow the path of the print request step-by-step.
- Examine the status of the LP print service at each step.
 - Is the configuration correct?
 - Is the printer accepting requests?
 - Is the printer enabled to process requests?
- If the request is hanging on transmission, examine the `lpNet` log (`/var/lp/logs/lpNet`).
- If the request is hanging locally, examine the `lpsched` log (`/var/lp/logs/lpsched`).
- If the request is hanging locally, have notification of the printer device errors (faults) mailed to you, and re-enable the printer.

The procedures in the section “Instructions for Troubleshooting Printing Problems” on page 178 use this strategy to help you troubleshoot various problems with the LP print service.

If basic troubleshooting of the LP print service does not solve the problem, you need to follow the troubleshooting steps for the specific client-server case that applies:

- SunOS 5.x print client using a SunOS 4.1 print server (for instructions, see page 190)
- SunOS 5.x print client using a SunOS 5.x print server (for instructions, see page 185)
- SunOS 4.1 print client using a SunOS 5.x print server (for instructions, see page 194)

Troubleshooting Incorrect Output

If the printer and the print service software are not configured correctly, the printer may print, but it may provide output that is not what you expect.

Check the Printer Type and File Content Type

If you used the wrong printer type when you set up the printer with the LP print service, inappropriate printer control characters can be sent to the printer. The results are unpredictable: nothing may print, output may be illegible, or output may be printed in the wrong character set or font.

If you specified an incorrect file content type on the SunOS 5.x print client or SunOS 5.x print server, the banner page may print, but that is all. The file content types specified for a printer indicate the types of files the printer can print directly without filtering. When a user sends a file to the printer, the file is sent directly to the printer without any attempt to filter it. The problem occurs if the printer cannot handle the file content type.

When setting up print clients, you increase the chance for a mistake because the file content types must be correct on both the print server and the print client. As recommended, if the print client is set up with `any` as the file content type, files are sent directly to the server and the need for filtering is determined on the server. Consequently, the file content types have to be specified correctly only on the server.

You can specify a file content on the print client to off-load filtering from the server to the client, but the content type must be supported on the print server.

Check the `stty` Settings

Many formatting problems can result when the default `stty` (standard terminal) settings do not match the settings required by the printer. The following sections describe what happens when some of the settings are incorrect.

Wrong Baud Settings

When the baud setting of the computer does not match the baud setting of the printer, usually you get some output, but it does not look like what you submitted for printing. Random characters are displayed, with an unusual mixture of special characters and undesirable spacing. The default for the LP print service is 9600 baud.

Note – If a printer is connected by a parallel port, the baud setting is irrelevant.

Wrong Parity Setting

Some printers use a parity bit to ensure that data received for printing has not been garbled during transmission. The parity bit setting for the computer and the printer must match. If they do not match, some characters either will not be printed at all, or they will be replaced by other characters. The output will look approximately correct, with the word spacing all right and many letters in their correct place. The LP print service does not set the parity bit by default.

Wrong Tab Settings

If tabs are set, but the printer expects no tabs, the printed output may contain the complete contents of the file, but the text may be jammed against the right margin. Also, if the tab settings for the printer are incorrect, the text may not have a left margin, it may run together, it may be concentrated to a portion of the page, or it may be incorrectly double-spaced. The default is for tabs to be set every eight spaces.

Wrong Return Setting

If the output is double-spaced, but it should be single-spaced, either the tab settings for the printer are incorrect or the printer is adding a line feed after each Return. The LP print service adds a Return before each line feed, so the combination causes two line feeds.

If the print zigzags down the page, the `stty` option `onlcr` that sends a Return before every line feed is not set. The `stty=onlcr` option is set by default, but you may have cleared it while trying to solve other printing problems.

Troubleshooting Hung LP Print Service Commands

If you type any of the `lp` commands (`lpssystem`, `lpadmin`, `lpstat`, `lpshut`), and nothing happens (no error message, status information, or prompt is displayed), chances are something is wrong with the LP scheduler. Such a problem can usually be resolved by stopping and restarting the LP scheduler.

Troubleshooting Idle (Hung) Printers

You may find a printer that is idle, even though it has print requests queued to it. A printer may seem idle when it should not be for one of the following reasons:

- The current print request is being filtered.
- The printer has a fault.
- Networking problems may be interrupting the printing process.

Check the Print Filters

Slow print filters run in the background to avoid tying up the printer. A print request that requires filtering will not print until it has been filtered.

Check Printer Faults

When the LP print service detects a fault, printing resumes automatically, but not immediately. The LP print service waits about five minutes before trying again, and continues trying until a request is printed successfully. You can force a retry immediately by enabling the printer.

Check Network Problems

When printing files over a network, you may encounter the following types of problems:

- Requests sent to print servers may back up in the client system (local) queue.
- Requests sent to print servers may back up in the print server (remote) queue.

Jobs Backed Up in the Local Queue

Jobs earmarked for a print server may back up in the client system queue for the following reasons:

- The print server is down.
- The printer is disabled on the print server.
- The network between the print client and print server is down.
- Underlying SunOS 5.x network software was not set up properly.

While you are tracking down the source of the problem, you should stop new requests from being added to the queue.

Jobs Backed Up in the Remote Queue

If jobs back up in the print server queue, the printer has probably been disabled. When a printer is accepting requests, but not processing them, the requests are queued to print. Unless there is a further problem, once the printer is enabled, the print requests in the queue should print.

Troubleshooting Conflicting Status Messages

A user may enter a print request and be notified that the client system has accepted it, then receive mail from the print server that the job has been rejected. These conflicting messages may occur for the following reasons:

- The print client may be accepting requests, while the print server is rejecting requests.
- The definition of the printer on the print client might not match the definition of that printer on the print server. More specifically, the definitions of the print job components, like filters, character sets, print wheels, or forms are not the same on the client and server systems.

You should check that identical definitions of these job components are registered on both the print clients and print servers so that local users can access printers on the print servers.

Instructions for Troubleshooting Printing Problems

This section contains instructions that explain:

- How to troubleshoot no output
- How to troubleshoot incorrect output
- How to unhang the LP commands
- How to troubleshoot an idle (hung) printer
- How to resolve conflicting status messages

▼ How to Troubleshoot No Output

This task includes the following troubleshooting procedures to try when you submit a print request to a printer and nothing prints:

- Check the hardware (page 179)
- Check the network (page 180)
- Check the LP print service basics (page 181)
- Check printing from SunOS 5.x print client to SunOS 5.x print server (page 185)
- Check printing from SunOS 5.x print client to SunOS 4.1 print server (page 190)
- Check printing from SunOS 4.1 print client to SunOS 5.x print server (page 194)

Try the procedures in the listed order, before going to the specific print client/server case that applies. However, if the banner page prints, but nothing else does, turn to the instructions under “How to Troubleshoot Incorrect Output” on page 199.

To check the hardware:

- 1. Check that the printer is plugged in and turned on.**
- 2. Check that the cable is connected to the port on the printer and to the port on the system or server.**
- 3. Check that the cable is the correct cable and that it is not defective.**
Refer to the manufacturer’s documentation. If the printer is connected to a serial port, make sure that the cable supports hardware flow control. A NULL Modem adapter supports this.

Table 7-1 Pin Configuration for NULL Modem Cables

	Host	Printer
Mini-Din-8	25-Pin D-sub	25-Pin D-sub
-	1 (FG)	1(FG)
3(TD)	2(TD)	3(RD)
5(RD)	3(RD)	2(TD)

Table 7-1 Pin Configuration for NULL Modem Cables (Continued)

	Host	Printer
Mini-Din-8	25-Pin D-sub	25-Pin D-sub
6(RTS)	4(RTS)	5 (CTS)
2(CTS)	5 (CTS)	4(RTS)
4(SG)	7(SG)	7(SG)
7 (DCD)	6(DSR), 8(DCD)	20(DTR)
1(DTR)	20(DTR)	6(DSR), 8(DCD)

4. Check that any hardware switches for the ports are set properly.
5. Check that the printer is operational.
Use the printer's self-test feature, if the printer has one. Check the printer documentation for information about printer self-testing.
6. Check that the baud settings for the computer and the printer are correct.
If the baud settings are not the same for both the computer and the printer, sometimes nothing will print, but more often you get incorrect output. For instructions, see "How to Troubleshoot Incorrect Output" on page 199.

To check the network:

1. On a print client, type `ping server-name` and press Return. On the print server, type `ping client-name` and press Return.
This command helps you check that the network link between the print server and the print client is set up correctly.

```
# ping neptune
neptune is alive
# ping jupiter
jupiter not available
```

If the message tells you the system is alive, you know you can reach it, so the network is all right. It also tells you that either a naming service or the local `/etc/hosts` file has translated the host (system) name you entered into an IP address; otherwise, you would have to enter the IP address.

If you get a not available message, try to answer the following questions: How is NIS or NIS+ set up at your site? Do you need to take additional steps so that print servers and print clients can communicate with one another? If your site is not running NIS or NIS+, have you entered the IP address for the print server in each print client's `/etc/hosts` file, and entered all print client IP addresses in the `/etc/hosts` file of the print server?

2. **Check that the port monitor is configured correctly on the print server.**
See "How to Set Up a SunOS 5.x Print Server" on page 288.
3. **Check that the network listen services are registered with the port monitor on the print server.**
See "How to Set Up a SunOS 5.x Print Server" on page 288.

To check the basic functions of the LP print service:

1. **On both the print server and print client, make sure that the LP print service is running.**
 - a. **Type `lpstat -r` and press Return.**
This command shows if the LP scheduler is running.

```
# lpstat -r
scheduler is running
```

- b. **If the scheduler is not running, become root or lp, type `/usr/lib/lp/lpsched` and press Return.**
If you have trouble starting the scheduler, see "How to Unhang the LP Commands" on page 204.
2. **On both the print server and print client, make sure that the printer is accepting requests.**

a. Type `lpstat -a` and press Return.

This command verifies that the LP system is accepting requests for each printer configured for the system.

```
# lpstat -a
red accepting requests since Wed Mar 13 20:37:07 PST 1994
luna not accepting requests since Wed Apr 17 19:10:55 PDT 1994
unknown reason
```

b. If the printer is not accepting requests, become root or lp, type `accept printer-name` and press Return.

The printer you specify now accepts requests.

3. On both the print server and print client, make sure the printer is enabled to print requests.

a. Type `lpstat -p printer-name` and press Return.

Information about printer status is displayed. You can omit the printer name to obtain information about all printers set up for the system. The example shows a printer that is disabled.

```
# lpstat -p luna
printer luna disabled since Wed Apr 17 19:13:33 PDT 1994.
available.
unknown reason
```

b. If the printer is disabled, become root or lp, type `enable printer-name` and press Return.

The printer you specify is enabled.

```
# enable luna
printer "luna" now enabled.
```

4. On the print server, make sure that the printer is connected to the correct serial port.

a. Type `lpstat -t` and press Return.

This command tells you the port to which the printer is connected. The message `device` for *printer-name* shows the port address. Is the cable connected to the port to which the LP print service thinks it is connected? If the port is correct, skip to step 5. In the example, the printer is connected to `/dev/term/a`.

```
# lpstat -t
scheduler is running
system default destination: luna
device for luna: /dev/term/a
```

b. Become root or lp.

c. Type `chown lp device-filename` and press Return.

This command assigns the special user `lp` as the owner of the device file.

d. Type `chmod 600 device-filename` and press Return.

This command allows only `lp` (or root) to access the printer port device file.

5. On both the print server and print client, make sure that the printer is configured properly.

a. Type `lpstat -p printer-name -l` and press Return.

The example shows a PostScript printer that is configured properly, and that is available to process print requests. If the printer type and file content type are correct, skip to step 6.

```
# lpstat -p luna -l
printer luna is idle. enabled since Wed Feb  4
20:17:21 PST 1970. available.
    Content types: postscript
    Printer types: PS
```

b. If the printer type or file content type is incorrect, type `lpadmin -p printer-name -T printer-type -I file-content-type` and press Return.

On the print client, try setting the print type to `unknown` and the content type to `any`.

6. On the print server, make sure that the printer is not waiting because of a printer fault.

- a. **Type** `lpadmin -p printer-name -F` continue **and press Return**.
You have instructed the LP print service to continue if it is waiting because of a fault.
- b. **Type** `enable printer-name` **and press Return**.
This command forces an immediate retry.
- c. **(Optional) Type** `lpadmin -p printer-name -A 'write root'` **and press Return**.
You have instructed the LP print service to set a default policy of writing root—sending the printer fault message to the terminal on which root is logged in—if the printer fails. This may help you to get quick notification of faults as you try to fix the problem.

7. Make sure that the printer is not set up incorrectly as a login terminal.

Note – It is easy to mistakenly set up a printer as a login terminal.

- a. **Type** `ps -ef` **and press Return**.
Look for the printer port entry. In the example, port `/dev/term/a` is incorrectly set as a login terminal. You can tell by the `"passwd\n##` information at the end of the line. If the port is set correctly, skip the last steps in this procedure.

```
# ps -ef
root 169 167 0 Apr 04 ? 0:08 /usr/lib/saf/listen tcp
root 939 1 0 19:30:47 ? 0:02 /usr/lib/lpsched
root 944 939 0 19:30:47 ? 0:00 lpNet
root 859 858 0 19:18:54 term/a 0:01 /bin/sh -c /etc/lp/interfaces/luna \
luna-294 rocket!smith "passwd\n##
#
```

- b. **Type** `cancel request-id` **and press Return**.

```
# cancel luna-294
```

- c. **Type** `lpadmin -p printer-name -h` **and press Return**.
The printer port is set to be a non-login device.

d. Type `ps -ef` and press Return.

Verify that the printer port is no longer a login device.

If you do not find the source of the printing problem in the basic LP print service functions, continue to one of the following procedures for the specific client/server case that applies.

To check printing from SunOS 5.x client to SunOS 5.x print server:

1. Check the basic functions of the LP print service on the print server if you haven't checked these already.

For instructions, see page 189. Make sure the printer works locally before trying to figure out why nothing prints when a request is made from a print client.

2. Check the basic functions of the LP print service on the print client.

You should have done this already. For instructions, go back to the previous section. On the print client, the LP scheduler has to be running, and the printer has to be enabled and accepting requests before any request from the client will print.

Note – For most of the following steps, you must be logged in as root or lp.

3. Make sure that the print server is accessible.

a. On the print client, type `ping print-server-name` and press Return.

An “are you there?” request is sent to the system you specify.

```
# ping neptune
neptune is alive
# ping jupiter
jupiter not available
```

If you receive the message *system* not available, you have a network problem.

4. On the print client, make sure the print server is identified as type `s5`.

a. Type `lpsystem -l` and press Return.

The example shows a print server, `neptune`, that is properly identified as type `s5` (SunOS 5.x).

```
# lpsystem -l
System:                neptune
Type:                  s5
Connection timeout:   never
Retry failed connections: after 10 minutes
Comment:               none
#
```

b. If the print server is incorrectly identified, type `lpsystem -t S5 print-server-name` and press Return.

```
# lpsystem -t S5 neptune
```

5. On the print client, check the print queue.

a. Type `cd /var/spool/lp/requests/system-name` and press Return.
A record of print requests still in the queue is kept in this directory. For more information about the content of the request logs, see the section “Log Files” on page 313.

b. Type `ls -l` and press Return.
A list of the print jobs is displayed.

c. For the print job you want to check, type `lpstat -o request-id` and press Return.

The example shows a job that is queued successfully.

```
# cd /var/spool/lp/requests/neptune
# ls -l
total 12
-rw-rw----  1 lp      lp          43 May 22 19:44 11-0
# lpstat -o luna-11
luna-11      root          364  May 22 19:59
#
```

If the job is not queued successfully, the client/server connection may be faulty.

6. Make sure that the client/server connection is not faulty.

a. On the print client, type `tail /var/lp/logs/lpNet` and press Return.

This command shows you if lpNet can connect to the print server. The example shows the log for a job that could not connect to the print server.

```
# tail /var/lp/logs/lpNet
05/21/94 19:36 p 1780 <none> Starting.
05/21/94 19:36 p 1780 <none> Starting lpNetParent.
05/21/94 19:36 p 1780 <none> Initialized & Polling.
05/21/94 19:36 c 1781 neptune Starting.
05/21/94 19:36 p 1780 <none> Started child for neptune, pid =
1781
***05/21/94 19:36 c 1781 neptune Could not connect to remote
child.
05/21/94 19:36 c 1781 neptune Normal process termination.
#
```

b. If the connection is *not* being made, on the print server,

type `lpstat -t` and press Return.

This command will show you if the print server is operating properly. A connection cannot be made otherwise.

The example shows a print server up and running.

```
# lpstat -t
scheduler is running
system default destination: luna
device for luna: /dev/term/a
luna accepting requests since Thu May 23 20:56:26 PDT 1994
printer red is idle. enabled since Sun May 19 17:12:24 PDT 1994.
available.
printer luna now printing luna-314. enabled since Fri May 24
16:10:39 PDT 1994. available.
luna-129          root          488    May 23 20:43 filtered
#
```

c. If the print server is not operating properly, go back to step 1, otherwise go to step d.

d. On the print server, type `tail /var/lp/logs/lpNet` and press Return.

Examine the lpNet log to see if the print server is connecting to the client. If there is no entry, lpNet is not transmitting correctly. The example shows the log for a job that connected to the print server.

```
# tail /var/lp/logs/lpNet
08/10/94 15:17 p   708 <none> Normal process termination.
08/10/94 15:17 p  3802 <none> Starting.
08/10/94 15:17 p  3802 <none> Starting lpNetParent.
08/10/94 15:17 p  3802 <none> Initialized & Polling.
08/10/94 15:17 p  3802 <none> Started child for saturn, pid = 3804
08/10/94 15:17 c  3804 saturn lpd starting (active)
08/10/94 15:17 c  3804 saturn lpd connected to saturn
08/10/94 15:17 c  3804 saturn lpd disconnecting from saturn
08/10/94 15:17 c  3804 saturn lpd connected to saturn
08/10/94 15:17 c  3804 saturn lpd disconnecting from saturn
```

7. On the print server, make sure that the print client is correctly specified as an s5 system.

a. Type `lpssystem -l` and press Return.

The example shows a print client, `neptune`, configured correctly.

```
# lpssystem -l
System:                neptune
Type:                  s5
Connection timeout:   never
Retry failed connections: after 10 minutes
Comment:               none
```

b. If the print client configuration is incorrect,

type `lpssystem -t s5 client-system-name` and press Return.

```
# lpssystem -t s5 neptune
```

8. On the print server, make sure that the port monitor and network listen service are set up properly.**a. Type `sacadm -l` and press Return.**

The example shows a print server configured correctly.

```
# sacadm -l
PMTAG      PMTYPE      FLGS RCNT STATUS      COMMAND
tcp        listen      -    9999 ENABLED    /usr/lib/saf/listen tcp #
#
```

b. Type `pmadm -l` and press Return.

The example shows a print server configured for all three types of services.

```
# pmadm -l
PMTAG      PMTYPE      SVCTAG      FLGS ID      <PMSPECIFIC>
tcp        listen      lp          -    root      - - p - /var/spool/lp/fifos/listenS5 #
tcp        listen      lpd        -    root      \x000202038194143a0000000000000000
- p - /var/spool/lp/fifos/listenBSD #
tcp        listen      0          -    root      \x00020ACE8194143a0000000000000000
- c - /usr/lib/saf/nlps_server #
```

If either step 7 or step 8 shows a problem, see “How to Set Up a SunOS 5.x Print Server” on page 288 for instructions on setting up the port monitor and network listen services.

To check printing from SunOS 5.x client to SunOS 4.1 print server:

1. Check the basic functions of the LP print service on the print client if you have not checked these already.

For instructions, see “To check the basic functions of the LP print service:” on page 181.

2. Make sure the print server is accessible.

a. On the print client, type `ping print-server-name` and press Return.

An “are you there?” request is sent to the system you specify.

```
# ping neptune
neptune is alive
# ping jupiter
jupiter not available
```

If you receive the message *system not available*, you have a network problem.

3. Make sure the `lpd` daemon on the print server is running.

a. On the print server, type `ps -ax | grep lpd` and press Return.

If the `lpd` daemon is running, a line is displayed, as shown in the example. If it is not running, no process information is shown.

```
% ps -ax | grep lpd
 126 ?  IW   0:00 /usr/lib/lpd
 200 p1 S    0:00 grep lpd
%
```

b. If `lpd` is not running on the print server, become root on the print server, type `/usr/lib/lpd &` and press Return.

4. Make sure the remote `lpd` daemon is configured properly.

- a. **On the print server, become root, and type `/usr/etc/lpc` and press Return.**

The `lpc>` prompt is displayed.

- b. **Type `status` and press Return.**

Status information is displayed. In the following example, the daemon is not running and needs to be restarted.

```
# /usr/etc/lpc
lpc> status
red:
queuing is enabled
printing is enabled
no entries
no daemon present
lpc>
```

- c. **If no daemon is present, at the `lpc>` prompt, type `restart` and press Return.**

The daemon is restarted.

- d. **Type `status` and press Return.**

Verify that the `lpd` daemon has started.

- e. **Type `quit` and press Return.**

You are returned to the shell prompt.

5. **Make sure the print client has access to the print server.**

- a. **Check if there is an `/etc/hosts.lpd` file on the 4.1 print server.**

On a 4.1 print server, if this file exists, it is used to determine whether an incoming print request can be accepted. If the file does not exist, all print client systems have access, so skip steps b and c.

- b. **If the file exists, see if the print client is listed in the file.**

Requests from client systems not listed in the file are not transferred to the print server.

- c. **If the client is not listed, add the print client to the file.**

Note – If you get this far without pinpointing the problem, the SunOS 4.1 system is probably set up and working properly.

6. Make sure that the connection to the remote `lpd` print daemon from the print client is made correctly.

a. On the print client, become root, and type `ps -ef | grep lp` and press Return.

The `lpNet` and `lpsched` daemons should be running, as shown in the example.

```
# ps -ef | grep lp
root  162  154 51   Jan 07 ?           0:01 lpNet
root  154    1 80   Jan 07 ?           0:02 /usr/lib/lpsched
```

If the `lpNet` daemon is running, skip to step 6.

b. Type `lpshut` and press Return.

The LP print service is stopped.

c. Type `/usr/lib/lp/lpsched` and press Return.

The LP print service is restarted, including the `lpNet` daemon.

7. Make sure that the remote print server is identified correctly as a SunOS 4.1 system.

a. On the print client, become root, type `lpsystem -l` and press Return.

The example shows a SunOS 4.1 print server, `jupiter`, that is specified correctly, as shown by `Type` being set to `bsd`.

```
# lpsystem -l
System:                jupiter
Type:                  bsd
Connection timeout:   never
Retry failed connections: after 10 minutes
Comment:               none
```

b. If the print server is incorrectly identified, type `lpsystem -t bsd print-server-name` and press Return.

8. Make sure that the print client is *not* having trouble connecting to the print server.

- a. On the print client, type `tail -100 /var/lp/logs/lpNet` and press **Return**.

By examining the lpNet log, you can tell if the print client (for example, jupiter) is reaching the print server. Normally, the contents will be similar to the following:

```
# tail -100 /var/lp/logs/lpNet
04/18/94 09:40 p 1097 <none> Starting.
04/18/94 09:40 p 1097 <none> Starting lpNetParent.
04/18/94 09:40 p 1097 <none> Initialized & Polling.
04/17/94 19:32 c 965 jupiter lpd connected to jupiter
04/17/94 19:32 c 965 jupiter lpd disconnecting from jupiter
#
```

If the results appear normal, skip the last steps in this procedure. If there is a problem, you will see retries to the BSD system, as shown below.

```
# tail -100 /var/lp/logs/lpNet
05/23/94 14:39 c 120 jupiter lpd retrying connection to jupiter
05/23/94 14:51 c 120 jupiter lpd retrying connection to jupiter
05/23/94 15:02 c 120 jupiter lpd retrying connection to jupiter
#
```

- b. On the print client, type `lpssystem -l print-server-name` and press **Return**.

This command shows you the retry and time-out parameters currently set.

- c. Type `lpssystem -T {n,0,N} -R {n,0,N} print-server-name` and press **Return**.

The `-T` option specifies the length of time a network connection can be idle before it is dropped. Choose either `n` (never time out), `0` (drop immediately), or enter a number (wait `N` minutes, then drop connection). The default is `n`. The `-R` option specifies the length of time to wait before trying to re-establish a connection. Choose either `n` (don't retry until

there is more work), or 0 (try to reconnect immediately), or enter a number (wait *N* minutes before trying to reconnect). The default is wait 10 minutes before trying to reconnect.

```
# lpsystem -T n -R 0 saturn
"saturn" has been modified.
```

To check printing from SunOS 4.1 client to SunOS 5.x print server:

Note – You should be logged in as root or lp on the system specified in these steps.

1. Check the basic functions of the LP print service on the print server if you have not checked these already.

For instructions, go back to “To check the basic functions of the LP print service:” on page 181. Make sure that the printer works locally before trying to figure out why nothing prints when a request is made from a print client.

2. Make sure that the print client is accessible.

- a. On the SunOS 5.x print server, type `ping print-client-name` and press Return.**

```
# ping rocket
rocket is alive
```

If you receive the message *system not available*, you have a network problem.

3. On the print client, type `lpr -P printer-name filename` and press Return.

This command shows if the print client is working. The example shows that the print client is not working correctly.

```
# lpr -P luna /etc/fstab
lpr: cannot access luna
#
```

4. Make sure the `lpd` daemon is running on the print client.

a. Type `ps -ax | grep lpd` and press Return.

This command shows if the `lpd` daemon is running on the print client. The example shows that the daemon is running.

```
% ps -ax | grep lpd
118 ? IW 0:02 /usr/lib/lpd
%
```

b. On the print client, become root, type `/usr/lib/lpd &` and press Return.

5. On the print client, make sure there is a `printcap` entry identifying the print server.

a. Type `lpr -P printer-name filename` and press Return.

The example shows that the `/etc/printcap` file does not have an entry for the specified printer.

```
% lpr -P mercury /etc/fstab
lpr: mercury: unknown printer
%
```

b. If there is no entry, edit the `/etc/printcap` file and add the following information:

```
printer-name | print-server-name: \
:lp=:rm=print-server-name:rp=printer-name:br#9600:rw:\
:lf=/var/spool/lpd/printer-name/log:\
:sd=/var/spool/lpd/printer-name:
```

The example shows an entry for printer `luna` connected to print server `neptune`.

```
luna|neptune:\
:lp=:rm=neptune:rp=luna:br#9600:rw:\
:lf=/var/spool/lpd/luna/log:\
:sd=/var/spool/lpd/luna:
```

c. Create a spooling directory (`/var/spool/lpd/printer-name`) for the printer.

- 6. Make sure that the print client `lpd` is not in a wait state by forcing a retry.**
If the print server is up and responding, the print client `lpd` may be in a wait state before attempting a retry.
 - a. As root on the print client, type `lpc` and press Return.**
The `lpc>` prompt is displayed.
 - b. Type `restart printer-name` and press Return.**
 - c. Type `quit` and press Return.**
The shell prompt is redisplayed.

```
# lpc
lpc> restart luna
luna:
    no daemon to abort
luna:
    daemon started
#quit
terra%
```

- 7. Check the connection to the print server.**
 - a. On the print client, become root, type `more /var/spool/lpd/printer-name/log` and press Return.**
Frequently, no information is displayed.
 - b. Type `more /var/spool/lpd/printer-name/status` and press Return.**

```
# more /var/spool/lpd/luna/status
waiting for neptune to come up
#
```

c. If the connection is all right, on the print server, type `lpstat -t` and press Return.

This command shows you if the print server is operating properly. The example shows a print server that is up and running. If the print server is not running, go back to step 1 before continuing.

```
# lpstat -t
scheduler is running
system default destination: luna
device for luna: /dev/term/a
luna accepting requests since Thu May 23 20:56:26 PDT 1994
printer uranus is idle. enabled since Sun May 19 17:12:24 PDT
1994.
available.
printer luna now printing luna-314. enabled since Fri May
24 16:10:39 PDT 1994. available.
luna-129          root          488    May 23 20:43 filtered
#
```

d. On the print server, type `tail /var/lp/logs/lpNet` and press Return.

Examine the messages to see if the connection is being made from the print client to the lpNet process on the print server. The example shows a print client that is transmitting correctly.

```
# tail /var/lp/logs/lpNet
# tail /var/lp/logs/lpNet
05/24/94 16:26 c 3651 rocket lpd exiting, status=0
05/24/94 16:33 c 3727 rocket lpd starting (passive)
05/24/94 16:33 c 3727 rocket rocket requests rcvjob luna
05/24/94 16:33 c 3727 rocket lpd exiting, status=0
05/24/94 16:43 c 3835 rocket lpd starting (passive)
05/24/94 16:43 c 3835 rocket rocket requests rcvjob luna
05/24/94 16:43 c 3835 rocket lpd exiting, status=0
#
```

If there is no entry, lpNet is not transmitting correctly. If the connection is being made, it implies the problem is on the print client, and you should check the basic functions of the LP print service on the print client before continuing.

8. On the print server, make sure the print client is a BSD system.

a. Type `lpsystem -l` and press Return.

The example shows a print client, *rocket*, that is configured correctly.

```
# lpsystem -l
System:                rocket
Type:                  bsd
Connection timeout:   never
Retry failed connections: after 10 minutes
Comment:              none
#
```

b. If the print client is not specified correctly,

type `lpsystem -t bsd print-client-name` and press Return.

9. On the print server, make sure that the port monitor and network listen service are set up properly.

a. Type `sacadm -l` and press Return.

The example shows a print server that is configured correctly.

```
# sacadm -l
PMTAG      PMTYPE      FLGS RCNT STATUS      COMMAND
tcp        listen      -    9999 ENABLED    /usr/lib/saf/listen tcp #
#
```

b. Type `pmadm -l` and press Return.

The example shows a server that is configured for all three services.

```
# pmadm -l
PMTAG      PMTYPE      SVCTAG      FLGS ID      <PMSPECIFIC>
tcp        listen      lp          -    root      - - p - /var/spool/lp/fifos/listenS5 #
tcp        listen      lpd        -    root      \x000202038194143a0000000000000000
- p - /var/spool/lp/fifos/listenBSD #
tcp        listen      0          -    root      \x00020ACE8194143a0000000000000000
- c - /usr/lib/saf/nlps_server #
```

If either step 9a or step 9b shows a problem, see “How to Set Up a SunOS 4.1 Local Printer” on page 288 for instructions on how to set up the port monitor and network listen services.

▼ How to Troubleshoot Incorrect Output

1. Become root or lp.

2. Make sure that the printer type is correct.

An incorrect printer type may cause incorrect output. For example, if you specify printer type PS and the pages print in reverse order, try printer type PSR. (These type names must be in upper case.) Also, an incorrect printer type may cause missing text, illegible text, or text with the wrong font. To determine the printer type, examine the `terminfo` database (see “Directory Structure of the `terminfo` Database” on page 84).

a. On the print server and print client, type `lpstat -p printer-name -l` and press Return.

This command lists the characteristics of the printer.

```
mars% lpstat -p luna -l
printer luna is idle. enabled since Wed Jan  2 18:20:22 PST 1994.
available.
    Content types: simple,postscript
    Printer types: PS
    Description:
    Users allowed:
        (all)
    Forms allowed:
        (none)
    Banner not required
    Character sets:
        (none)
    Default pitch:
    Default page size:
mars%
```

b. Consult the printer manufacturer’s documentation to determine the printer model.

- c. **If the printer type is not correct, change it with the Printer Manager's Modify Printer option, or type `lpadmin -p printer-name -T printer-type` and press Return.**

On the print client, the printer type should be `unknown`. On the print server, the printer type must match a `terminfo` entry that is defined to support the model of printer you have. If there is no `terminfo` entry for the type of printer you have, see "How to Adjust the `terminfo` Database for an Unsupported Printer" on page 345.

- 3. **If the banner page prints, but there is no output for the body of the document, check the file content types.**

File content types specified for a printer indicate the types of files the printer can print directly without filtering. An incorrect file content type causes filtering to be bypassed when in fact it may be needed.

- a. **Note the information on file content type that was supplied in the previous step by the `lpstat` command.**

On the print client, the file content type should be `any`, unless you have good reason to specify one or more explicit content types. If a content is specified on the client, filtering is done on the print client, rather than the print server. In addition, content types on the client must match the content types specified on the print server, which in turn must reflect the capabilities of the printer.

- b. **Consult your printer manufacturer's documentation to determine which types of files the printer can directly print.**

The names you use to refer to these types of files do not have to match the names used by the manufacturer. But the names you use must agree with the names used by the filters known to the LP print service.

- c. **If the file content type is not correct, change it with the Printer Manager's Modify Printer option, or**

type `lpadmin -p printer-name -I file-content-type(s)` and press Return.

Run this command on either the print client, or print server, or both, as needed. Try `-I any` on the print client, and `-I ""` on the print server. The latter specifies a null file content type list, which means an attempt should be made to filter all files, because the printer can directly print only files that exactly match its printer type. This combination is a good first choice when files are not printing. If it works, you may want to try specifying explicit content types on the print server to reduce unnecessary filtering. For a local PostScript printer, you should use

`postscript`, or `postscript, simple`— if in fact, the printer supports these types. Be aware that `PS` and `PSR` are not file content types; they are printer types.

If you omit `-I`, the file content list defaults to `simple`. If you use the `-I` option and want to specify file content types in addition to `simple`, `simple` must be included in the list.

When specifying multiple file content types, separate the names with commas. Or you can separate names with spaces and enclose the list in quotation marks. If you specify `any` as the file content type, no filtering will be done and only file types that can be printed directly by the printer should be sent to it.

4. Check that the print request does not bypass filtering needed to download fonts.

If a user submits a print request to a PostScript printer with the command `lp -T PS`, no filtering is done. Try submitting the request with the command `lp -T postscript` to force filtering, which may result in the downloading of non-resident fonts needed by the document.

5. Make sure the `stty` settings for the printer port are correct.

- a. Read the printer documentation to determine the correct `stty` settings for the printer port.**

Note – If a printer is connected by a parallel port, the baud setting is irrelevant.

b. To examine the current settings, type `stty -a < device-name` and press Return.

This command shows the current `stty` settings for the printer port.

```

mars# stty -a < /dev/term/a
speed 9600 baud;
rows = 0; columns = 0; ypixels = 0; xpixels = 0;
eucw 1:0:0:0, scrw 1:0:0:0
intr = ^c; quit = ^|; erase = ^?; kill = ^u;
eof = ^d; eol = <undef>; eol2 = <undef>; swtch = <undef>;
start = ^q; stop = ^s; susp = ^z; dsusp = ^y;
rprnt = ^r; flush = ^o; werase = ^w; lnext = ^v;
parenb -parodd cs7 -cstopb -hupcl cread -clocal -loblk -parext
-ignbrk brkint -ignpar -parmrk -inpck istrip -inlcr -igncr icrnl
-iuclc
ixon -ixany -ixoff imaxbel
isig icanon -xcase echo echoe echok -echonl -noflsh
-tostop echoctl -echoprt echoke -defecho -flusho -pendin iexten
opost -olcuc onlcr -ocrnl -onocr -onlret -ofill -ofdel tab3
mars#

```

Table 7-2 shows the default `stty` options used by the LP print service's standard printer interface program.

Table 7-2 Default `stty` Settings Used by the Standard Interface Program

Option	Meaning
9600	Set baud rate to 9600
cs8	Set 8-bit bytes
-cstopb	Send one stop bit per byte
-parity	Do not generate parity
ixon	Enable XON/XOFF (also known as START/STOP or DC1/DC3)
opost	Do "output post-processing" using the settings listed below
-olcuc	Do not map lowercase to uppercase
onlcr	Change line feed to carriage return/line feed
-ocrnl	Do not change carriage returns into line feeds
-onocr	Output carriage returns even at column 0

Table 7-2 Default stty Settings Used by the Standard Interface Program (Continued)

Option	Meaning
n10	No delay after line feeds
cr0	No delay after carriage returns
tab0	No delay after tabs
bs0	No delay after backspaces
vt0	No delay after vertical tabs
ff0	No delay after form feeds

c. To change the stty settings,
type `lpadmin -p printer-name -o "stty=options"` **and press Return.**
 Use Table 7-3 to choose stty options to correct various problems affecting print output.

Table 7-3 stty Options to Correct Print Output Problems

stty Values	Result	Possible Problem From Incorrect Setting
110, 300, 600, 1200, 1800, 2400, 4800, 9600, 19200, 38400	Sets baud rate to the specified value (enter only one baud rate)	Random characters and special characters may be printed and spacing may be inconsistent
oddp evenp -parity	Sets odd parity Sets even parity Sets no parity	Missing or incorrect characters appear randomly
-tabs	Sets no tabs	Text is jammed against right margin
tabs	Sets tabs every eight spaces	Text has no left margin, is run together, or is jammed together
-onlcr	Sets no carriage return at the beginning of line(s)	Incorrect double spacing
onlcr	Sets carriage return at beginning of line(s)	The print zigzags down the page

You can change more than one option setting by including the options in single quotation marks and separating them by spaces. For example, suppose the printer requires you to enable odd parity and set a 7-bit

character size. You would type a command like the following:

```
#lpadmin -p neptune -o "stty='parenb parodd cs7'"
```

The `stty` option `parenb` enables parity checking/generation, `parodd` sets odd parity generation, and `cs7` sets the character size to 7 bits.

6. **Type `lp -d printer-name filename` and press Return.**
Verify that the document prints correctly.

▼ How to Unhang the LP Commands

1. **Become root or lp.**
2. **Type `lpshut` and press Return.**
If this command hangs, press Control-c and proceed to the next step. If this command succeeds, skip to step 4.
3. **Type `ps -el | grep lp` and press Return.**
Use the process ID numbers (PID) from the first column in place of the *pid* variables in the next step.

```
# ps -el | grep lp
103 ?          0:00 lpNet
134 term/a    0:01 lpsched#
```

4. **Type `kill -15 pid1 pid2...` and press Return.**
This should stop the LP print service processes. If that is not the case, as a last resort go to step 5.

```
# kill -15 103 134
```

5. **Type `kill -9 pid1 pid2...` and press Return.**
All the `lp` processes are terminated.
6. **Type `rm /usr/spool/lp/SCHEDLOCK` and press Return.**
You have removed the `SCHEDLOCK` file so that you can restart the LP print service.

7. Type `/usr/lib/lp/lpsched` and press Return.

The LP print service is restarted. If you can't start the scheduler, turn to "How to Start or Stop the Print Service Scheduler" on page 120.

▼ **How to Troubleshoot an Idle (Hung) Printer**

This task includes a number of procedures to use when a printer appears idle but it should not be. It makes sense to try the procedures in order, but the order is not mandatory.

To check that the printer is ready to print:

1. Type `lpstat -p printer-name` and press Return.

The information displayed shows you if the printer is idle or active, enabled or disabled, or available or not accepting print requests. If everything looks all right, continue with other procedures in this section. If you cannot run the `lpstat` command, see "How to Unhang the LP Commands" on page 204.

2. If the printer is not available (not accepting requests),

type `accept printer-name` and press Return.

The printer begins to accept requests into its print queue.

3. If the printer is disabled, type `enable printer-name` and press Return.

This command re-enables the printer so that it will act on the requests in its queue.

To check for print filtering:

♦ Type `lpstat -o printer-name` and press Return.

See if the first waiting request is being filtered. If the output looks like the following example, the file is being filtered; the printer is not hung, it just is taking a while to process the request.

```
terra% lpstat -o luna
luna-10      fred          1261   Mar 12 17:34 being filtered
luna-11      iggy          1261   Mar 12 17:36 on terra
luna-12      jack          1261   Mar 12 17:39 on terra
terra%
```

To resume printing after a printer fault:

1. **Look for a message about a printer fault and try to correct the fault if there is one.**

Depending on how printer fault alerts have been specified, messages may be sent to root by email or written to a terminal on which you (root) are logged in.

2. **Type `enable printer-name` and press Return.**

If a request was blocked by a printer fault, this command will force a retry. If this command does not work, continue with other procedures in this section.

To send jobs to a remote printer when they back up in the local queue:

1. **On the print client, type `reject printer-name` and press Return.**

This command stops further queuing of print requests from the print client to the print server.

2. **Check that the print server and the network between the print client and the print server is up by typing `ping print-server-name` and press Return.**

3. **Type `more /var/lp/logs/lpNet` and press Return.**

The information displayed may help you pinpoint what is preventing the transmission of print requests from the print client to the print server.

4. **After you fix the problem, on the print client, type `accept printer-name` and press Return.**

This command allows new jobs to be queued.

5. **If necessary, on the print client, type `enable printer-name` and press Return.**

This command enables the printer you specify.

To free jobs from a print client that back up in the print server queue:

1. **On the print server, type `reject printer-name` and press Return.**

This command stops further queuing of print requests from any print client to the print server.

2. **Type `more /var/lp/logs/lpsched` and press Return.**

The information displayed may help you pinpoint what is preventing the print requests from the print client to the print server from being printed.

3. **After you fix the problem, on the print server, type `accept printer-name` and press Return.**
This allows new jobs to be queued.
4. **If necessary, on the print server, type `enable printer-name` and press Return.**

▼ How to Resolve Conflicting Status Messages

1. **Type `lpstat -p printer-name` and press Return.**
Check that the printer connected to the print server is enabled and is accepting requests. Users will see conflicting status messages when the print client is accepting requests, but the print server is rejecting requests.
2. **On the print server, type `lpstat -p -l printer-name` and press Return.**
This checks that the definition of the printer on the print client matches the definition of the printer on the print server. Look at the definitions of the print job components, like print filters, character sets, print wheels, and forms, to be sure they are the same on both the client and server systems so that local users can access printers on print server systems.

Part 3— Mail

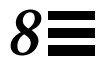
Part 3 has two chapters:

Chapter 8, “Understanding Mail Services,” describes mail service functions, the programs it uses, and how it works.

Chapter 9, “Setting Up and Administering Mail Services,” describes several typical mail scenarios and details the tasks for setting up mail for each scenario. It also provides information about how to troubleshoot a mail installation and how to administer mail services.

You can also refer to **Appendix C, “Customizing sendmail Configuration Files,”** explains the structure and syntax of the `sendmail` configuration file and how to customize it for more complex mail environments.

Understanding Mail Services



As a system administrator, you may need to expand an existing mail service or set up a new one. To help you with these tasks, this chapter describes the components of the mail service, defines mail services terminology, and describes several typical mail configurations. This chapter provides these sections:

- Components of the Mail Service
- Planning Your Mail System
- An Overview of the Mail Service
- How the Mail Service Works

This chapter provides conceptual information about mail services, and briefly outlines the tasks required for setting up each configuration. For step-by-step instructions, refer to other chapters. In particular, see Chapter 1, “Setting Up User Accounts and Groups,” for information about how to add users to an existing mail service, and Chapter 9, “Setting Up and Administering Mail Services,” for information about how to set up each of the typical mail configurations.

Components of the Mail Service

Mail services include many programs and daemons that interact with each other. This section introduces the programs and the terms and concepts related to administration of electronic mail.

Mail Services Programs and Files

Table 8-1 lists the mail services programs and files.

Table 8-1 Components of Mail Services Programs

Command	Description
/usr/bin/mailx	Mail program described in mailx(1)
/usr/bin/mail	Mailer that delivers mail to mailboxes
\$OPENWINHOME/bin/mailtool	Window-based interface to the sendmail program
/usr/lib/sendmail	Mail-routing program
/usr/lib/sendmail.mx	Mail-routing program linked with the domain name service resolver
/etc/mail/main.cf	Sample configuration file for main systems
/etc/mail/sendmail.subsidiary.cf	Sample configuration file for subsidiary systems
/etc/mail/sendmail.cf	Configuration file for mail routing
/etc/mail/aliases	Mail forwarding information
/etc/mail/sendmailvars	Table that stores macro and class definitions for lookup from sendmail.cf file
sendmailvars.org_dir	NIS+ version of sendmailvars table
/usr/bin/newaliases	Symbolic link to /usr/lib/sendmail
/usr/bin/mailq	Symbolic link to /usr/lib/sendmail
/usr/bin/mailstats	File used to store mail statistics generated by /etc/mail/sendmail.st (if present)

Table 8-1 Components of Mail Services Programs (Continued)

Command	Description
<code>/usr/bin/mconnect</code>	Connects to the mailer for address verification and debugging
<code>/usr/sbin/in.comsat</code>	Mail notification daemon
<code>/usr/sbin/syslogd</code>	Error message logger, used by <code>sendmail</code>

Mail services are provided by a combination of these programs, which interact as shown by the simplified diagram in Figure 8-1.

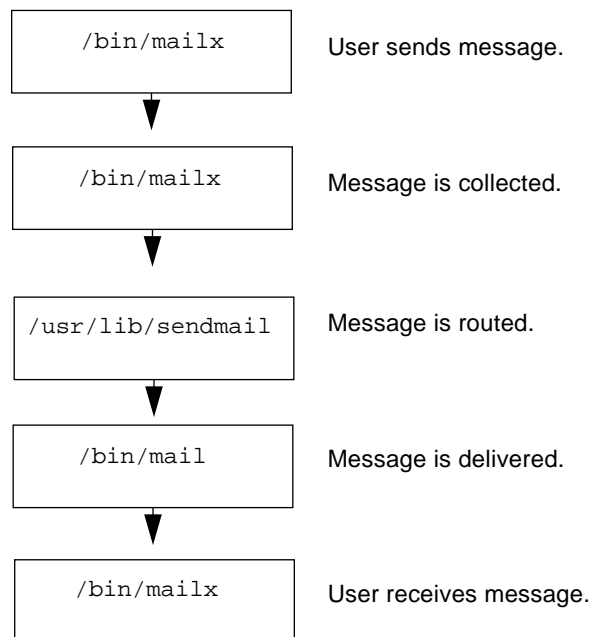


Figure 8-1 How Mail Programs Interact

Users send messages by using programs like `/bin/mailx`, or `mailtool`. See the reference manual pages for information about these programs.

The message is collected by the program that was used to generate it, and passed to the `sendmail` daemon. The `sendmail` daemon *parses* the addresses (divides them into identifiable segments) in the message, using information from the configuration file `/etc/mail/sendmail.cf` to determine network name syntax, aliases, forwarding information, and network topology. Using this information, `sendmail` determines the route a message must take to get to a recipient.

The `sendmail` daemon passes the message to the appropriate system. The `/bin/mail` program on the local system delivers the mail to the mailbox in the `/var/mail/username` directory of the recipient of the message.

The recipient is notified that mail has arrived, and retrieves it using the `/bin/mail`, `/bin/mailx`, `mailtool`, or similar programs.

Mail Services Terminology

This section defines the following terms and describes how they are used as part of the mail services:

- Systems in a mail configuration
 - Relay host
 - Gateway
 - Mail host
 - Mail server
 - Mail client
- User Agent (UA)
- Mail Transfer Agent (MTA)
- Mailers
- Domain Names
- Mail address
- Mailbox
- Aliases
- `sendmail` program
- Configuration file (`sendmail.cf`)
- Configuration table (`sendmailvars`)
- `.forward` files

Systems in a Mail Configuration

A mail configuration requires three elements, which can be combined on the same system or provided by separate systems:

- At least one mail server
- A mail host
- Mail clients

When you want users to communicate with networks outside your subnet, you must also have a relay host or a gateway.

Figure 8-2 shows a typical electronic mail configuration, using all three elements and a relay host. Each element is identified and described in the following sections.

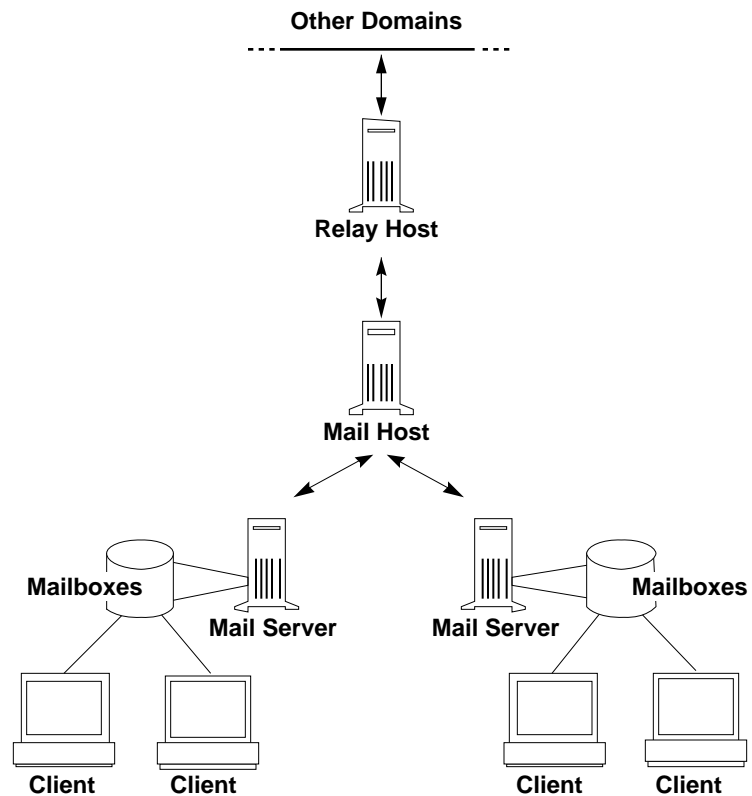


Figure 8-2 Typical Electronic Mail Configuration

Relay Host

A *relay host* is a system that runs at least one mail-related protocol, called a *mailer*. Each mailer specifies a policy and the mechanics to be used when delivering mail. The relay host handles mail with an address for which `sendmail` could not find a recipient in your domain. If a relay host exists, `sendmail` uses it for sending and receiving mail outside your domain.

The mailer on the sending relay host must be compatible with the mailer on the receiving system, as shown in Figure 8-3. You specify the mailer for your domain in the `sendmail.cf` file. The default `sendmail.cf` file defines several *mailer specifications*, like `smartuucp`, `ddn`, `ether`, and `uucp`. You can define others.

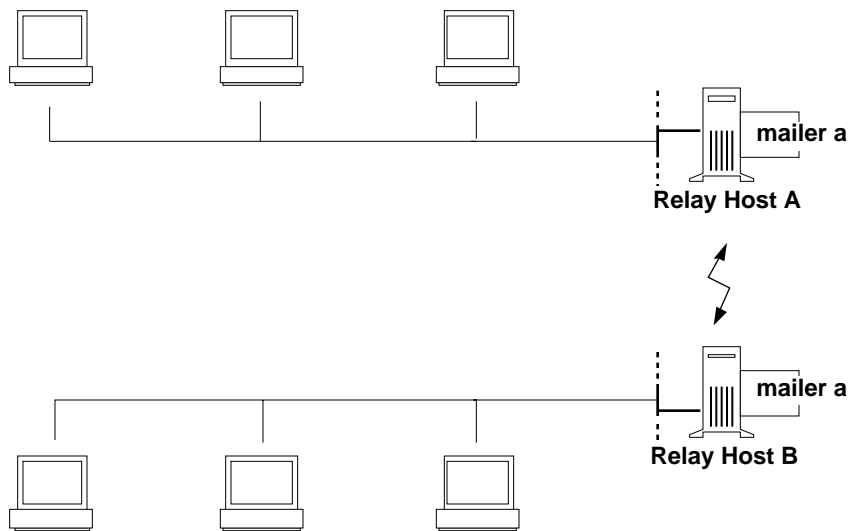


Figure 8-3 The Mailer on a Relay Host and the Receiving System

The relay host can be the same system as the mail host, or you can configure another system as the relay host. You can, in fact, configure more than one relay host for your domain if you have several connections outside of your site. If you have a `uucp` or Internet connection, configure the system with those connections as the relay host.

Gateway

A *gateway* is a system that handles connections between networks running different communications protocols, or between different subnets running the same communications protocols, as shown in Figure 8-4. A relay host and the system to which it is connected must use matching mailers. But, a gateway can handle connections to systems with unmatched mailers. You must customize the `sendmail.cf` file on the gateway system, which can be a difficult and time-consuming process.

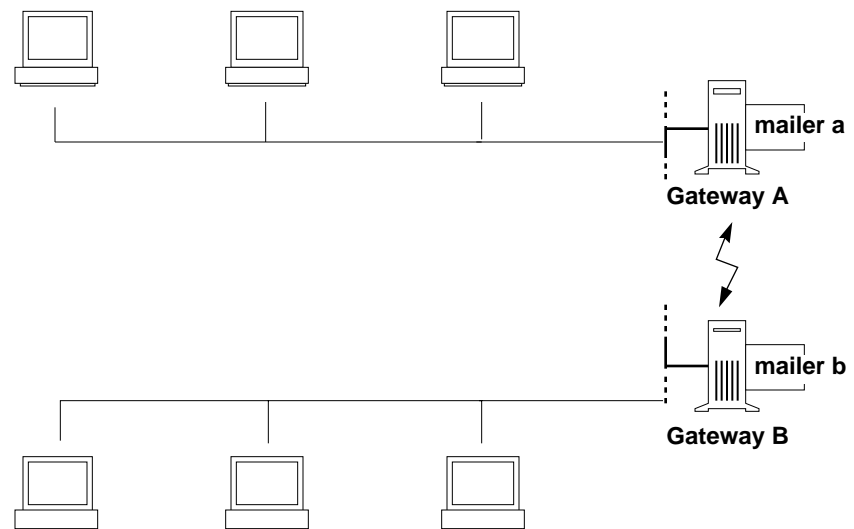


Figure 8-4 Gateway Between Different Communications Protocols

If you have to set up a gateway, you should find a gateway configuration file that is close to what you need, and modify it to fit your situation.

Mail Host

A *mail host* is a system that you designate as the main mail machine on your network. The mail host is the system to which other systems at the site forward mail that they cannot deliver. You designate a system as a mail host by adding

the word `mailhost` to the Internet Protocol (IP) address line in the system's `/etc/hosts` file. You must also use the `main.cf` file as the mail configuration file on the mail host system.

A good candidate for mail host is a system that is attached to an Ethernet and to phone lines, or a system configured as a router to the Internet. If you have a standalone system that is not networked but is in a time-sharing configuration, you can set up electronic mail on it. Treat the standalone system as the mail host of a one-system network. Similarly, if you have several systems on an Ethernet and none have phones, designate one as the mail host.

Mail Server

A *mail server* is any system that stores mailboxes in the `/var/mail` directory. The mail server routes all mail from a client. When a client sends mail, the mail server puts it in a queue for delivery. Once the mail is in the queue, the client can reboot or turn off the system without losing those mail messages. When the recipient gets mail from a client, the path in the "From" line of the message contains the name of the mail server. If the recipient responds, the response goes to the user's mailbox.

Mail is delivered to the system where the user's mailbox resides.

If the mail server is not the user's local system, users with NFS can mount the `/var/mail` directory by using the `/etc/vfstab` file, using the Automounter, or logging in to the server to read their mail.

Note - If you automount the `/var/mail` directory, you may have problems with mail on heterogeneous networks with SunOS 4.1 mail clients. Good candidates for mail servers are systems that provide a home directory for users, or that are backed up regularly.

Table 8-2 shows some sample statistics about the size of mail messages and mail traffic at a computer company with about 12,000 employees.

Note – The information in Table 8-2 is valid for ASCII messages only. With the advent of multimedia mail, which lets users transmit any type of data (not just ASCII text), the average size of an email message is likely to grow enormously. In the future, system administrators will need to allocate more spooling space for multimedia mailboxes.

Table 8-2 Sample Statistics for Mail Messages and Traffic

Statistic	Description
6,500 bytes	Average size of an email message
140 Kbytes	Amount of mail received by an average user in one day
15 Kbytes	Small mailbox size (user reads mail regularly and stores messages elsewhere)
40 Mbytes	Large mailbox size (user stores long-term mail in <code>/var/mail</code> mailbox)
18,000	Average number of messages per day sent outside the company
55,000	Average number of messages per day received from outside the company
2 Mbytes	Recommended spooling space to allocate for each user's mailbox, based on the figures in this table

Mail Client

A *mail client* is any system that receives mail on a mail server and does not have a local `/var/mail` directory.

You must make sure the mail client has the appropriate entry in the `/etc/vfstab` file and a mount point to mount the mailbox from the mail server.

User Agent

The *user agent* is the program that acts as the interface between the user and the `sendmail` program. The user agents supplied with the SunOS 5.x software system are `/usr/bin/mail`, `/usr/bin/mailx`, and `$/OPENWINHOME/bin/mailtool`.

Mail Transport Agent

The *mail transport agent* is responsible for actually receiving and delivering messages. The transport agent for the SunOS 5.x software is `sendmail`. The transport agent performs these functions:

- Accepts messages from the user agent
- Understands destination addresses
- Delivers mail originating on the local system to the proper mailboxes, if local, or to a delivery agent if not local
- Receives incoming mail from other delivery agents and delivers it to local users

Mailers

A *mailer* is a protocol that specifies the policy and mechanics used by `sendmail` when it delivers mail. You need to specify a mailer in the `sendmail.cf` file of a relay host or a gateway. The mailer for a relay host must match the mailer on the system outside of your domain. A gateway is a more complicated relay host (or you can think of a relay host as a simple gateway) and can communicate with more than one type of mailer.

The mailers provided with the SunOS 5.x system are:

- The `smartuucp` mailer (the default relay mailer) uses `uux` to deliver messages, but it formats headers with a domain-style address, and the `To:` and `CC:` lines are formatted by domain. For example, if `ignatz` in the `eng.acme.com` domain sends mail to `guy` at `auspex` using `smartuucp`, the headers look like this:

```
To: guy@auspex.com
From: ignatz@Eng.acme.COM
```

Use `smartuucp` for `uucp` mail to systems that can handle and resolve domain-style names. The sender also must be able to handle domain-style names and be able to receive replies from the Internet.

- The `uucp` mailer uses `uux` to deliver mail, but it uses an exclamation point address in the headers. For example, if `ignatz` in domain `acme.eng.acme.com` sends mail to `guy@auspex` using the `uucp` mailer, the headers look like this:

```
To: auspex!guy
From: acme!ignatz
```

Use `uucp` for `uucp` connections to systems that need a bang-style path.

- The `ddn` mailer uses the Simple Mail Transfer Protocol (SMTP) on port 25 to connect to the remote host. The `ddn` mailer inverts aliases and adds a domain name. For example, if `ignatz` in domain `eng.acme.com` sends mail to `paul@phoenix.stateu.edu`, the headers look like this:

```
To: paul@phoenix.stateu.edu
From: Iggy.Ignatz@Eng.acme.COM
```

If `ignatz` sends mail to `irving@sluggo` (both users in the `eng.acme.com` domain), the header looks like this:

```
To: Irving.Who@Eng.acme.Com
From: Iggy.Ignatz@Eng.acme.COM
```

Use `ddn` for sending mail outside of your domain, especially for mailers that you must reach through a relay.

- The `ether` mailer uses the SMTP on port 25 to connect to the remote host. The `ether` mailer does not invert aliases or append a domain name.

Use `ether` for systems in your `dns` domain that users can reach directly.

You can define other mailers by providing a mailer specification in the `sendmail.cf` file.

Domain Names

A *domain* is a directory structure for electronic mail addressing and network address naming. The domain address has this format:

```
mailbox@subdomain. . . .subdomain2.subdomain1.top-level-domain
```

The part of the address to the left of the @ sign is the local address, as shown in Figure 8-5.

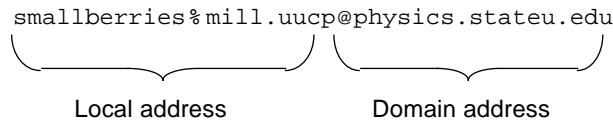


Figure 8-5 Domain Access Structure

The local address may contain information about the following:

- Routing using another mail transport (for example, bob::vmsvax@gateway or smallberries%mill.uucp@physics.uchicago.edu)
- An alias (for example, iggy.ignatz)
- A token that resolves to the name of a mailbox (for example, ignatz-->/var/mail/ignatz)

The receiving mailer is responsible for determining what the local part of the address means.

The part of the address to the right of the @ sign shows the domain address where the local address is located. A dot separates each part of the domain address. The domain can be an organization, a physical area, or a geographic region. In older forms the domain can show one or several computer systems.

Domain addresses are case-insensitive. It makes no difference whether you use uppercase, lowercase, or mixed case letters in the domain part of an address.

The order of domain information is hierarchical—the more local the address, the closer it is to the @ sign.

Note – British and New Zealand domains reverse this order. Most gateways automatically translate the reverse order of British and New Zealand domain names into the hierarchical order.

The larger the number of subdomains, the more detailed the information that is provided about the destination. Just as a subdirectory in a file-system hierarchy is considered to be inside the directory above, each subdomain in the mail address is considered to be inside the location to its right.

Table 8-3 shows the top-level domains in the United States.

Table 8-3 Top-level Domains in the United States

Domain	Description
Com	Commercial sites
Edu	Educational sites
Gov	Government installations
Mil	Military installations
Net	Networking organizations
Org	Nonprofit organizations

Table 8-4 shows the top-level domains for the United States and European countries. *!%@:: A Directory of Electronic Mail Addressing and Networks*, written by Donnalyn Frey and Rick Adams, contains a complete list of domain addresses, and is updated periodically.

Table 8-4 Top-Level Country Domains

Domain	Description
AT	Austria
BE	Belgium
CH	Switzerland
DE	Germany
DK	Denmark
ES	Spain
FI	Finland

Table 8-4 Top-Level Country Domains (Continued)

Domain	Description
FR	France
GR	Greece
IE	Ireland
IS	Iceland
IT	Italy
LU	Luxembourg
NL	The Netherlands
NO	Norway
PT	Portugal
SE	Sweden
TR	Turkey
UK	United Kingdom
US	United States
YU	Yugoslavia

For mail delivery, the network domain name and the mail domain name normally do not match. By default, the `sendmail` program strips off the first component from the network domain name to form the mail domain name. Mail is then delivered to the mailer for that mail domain. For example, if a NIS+ domain name were `bldg5.EBB.Eng.acme.COM`, its mail domain name would be `EBB.Eng.acme.COM`.

This default rule for determining the mail domain name restricts the number of components you can have in the network domain name. Fortunately, you can define the mail domain name in the `sendmail.cf` file. You can set the `m` variable (for mail domain name) using either a `D` macro definition or an `L` macro definition. The former is a simple assignment, while the latter uses a lookup table (`sendmailvars`) maintained by the naming service. The advantage of the lookup table is that you can change the mail domain name easily without having to edit the `sendmail.cf` file on each client.

Mail Address

The *mail address* contains the name of the recipient and the system to which the mail message is delivered.

When you administer a small mail system that does not use a naming service, addressing mail is easy: login names identify users uniquely.

When, however, you are administering a mail system that has more than one system with mailboxes, one or more domains, or when you have a `uucp` (or other) mail connection to the outside world, mail addressing becomes more complex. Mail addresses can be *route-based*, *route-independent*, or a mixture of the two.

Route-Based Addressing

Route-based addressing requires the sender of an email message to specify the local address (typically a user name) and its final destination, as well as the route that the message must take to reach its final destination. Route-based addresses are fairly common on `uucp` networks, and have this format:

```
host! path! user
```

Whenever you see an exclamation point as part of an email address, all (or some) of the route was specified by the sender. Route-based addresses are always read from left to right.

For example, an email address that looks like this:

```
venus!acme!sierra!hplabs!ucbvax!ignatz
```

reached user `ignatz` from the system named `venus` by going first to the address `acme`, then to `sierra`, then to `hplabs`, and finally to `ucbvax`. (Note that this is an example and not an actual route.) If any of the four mail handlers is out of commission, the message will be delayed or returned as undeliverable.

Route-Independent Addressing

Route-independent addressing requires the sender of an email message to specify the name of the recipient and the final destination address. Route-independent addresses usually indicate the use of a high-speed network like the Internet. In addition, newer `uucp` connections frequently use domain-style names. Route-independent addresses may have this format:

```
user@host.domain
```

Increased popularity of the domain hierarchical naming scheme for computers across the country is making route-independent addresses more common. In fact, the most common route-independent address omits the host name, and relies on the domain naming service to properly identify the final destination of the email message:

```
user@domain
```

Route-independent addresses are read by searching for the `@` sign, then reading the domain hierarchy from the right (the highest level) to the left (the most specific address to the right of the `@` sign).

Mailbox

A *mailbox* is a directory on a mail server that is the final destination for email messages. The name of the mailbox may be the user name, a group of users, or a place to put mail for someone with a specific function, like the postmaster. Mailboxes are in the `/var/mail/username` directory on either the user's local system or on a mail server. However, in either case, the mailbox is on the system to which the mail is delivered.

Mail should always be delivered to a local file system so that the user agent can pull mail from the mail spool and store it readily in the local mailbox. Do not use NFS-mounted file systems as the destination for a user's mailbox. NFS-mounted file systems cause problems with mail delivery and handling. Clients that NFS-mount `/var/mail` go into "Remote Mode" and arrange to have the server send and receive mail for them.

The `/etc/mail/aliases` file and naming services like NIS and NIS+ provide mechanisms for creating aliases for electronic mail addresses, so that users do not need to know the precise local name of a user's mailbox.

Some common naming conventions for special-purpose mailboxes are shown in Table 8-5.

Table 8-5 Conventions for the Format of Mailbox Names

Format	Description
<i>username</i>	User names are frequently the same as mailbox names.
<i>Firstname . Lastname</i> <i>Firstname_Lastname</i> <i>Firstinitial . Lastname</i> <i>Firstinitial_Lastname</i>	User names may be identified as full names with a dot (or an underscore) separating the first and last names, or by a first initial with a dot (or an underscore) separating the initial and the last name.
<code>postmaster</code>	Users can address questions and report problems with the mail system to the <code>postmaster</code> mailbox. Each site and domain should have a <code>postmaster</code> mailbox.
<code>MAILER-DAEMON</code>	<code>sendmail</code> automatically routes any mail addressed to the <code>MAILER-DAEMON</code> to the <code>postmaster</code> .
<code>x-interest</code>	Names with dashes are likely to be a distribution list or a mailing list. This format is commonly used for net mail groups.
<code>x-interest-request</code>	Names ending in <code>-request</code> are administrative addresses for distribution lists.
<code>owner-x-interest</code>	Names beginning with <code>owner-</code> are administrative addresses for distribution lists.
<i>local%domain</i>	The percent sign (%) marks a local address that is expanded when the message arrives at its destination. Most mail systems interpret mailbox names with % characters as full mail addresses. The % is replaced with an @, and the mail is redirected accordingly. Although many people use the % convention, it is not a formal standard. In the email community, it is referred to as the "% hack."

Aliases

An *alias* is an alternate name. For electronic mail, you can use aliases to assign additional names to a user, route mail to a particular system, or define mailing lists.

You need to create a mail alias for each user at your site. The alias points to where the mail is stored. Providing a mail alias is like providing a mail stop as part of the address for an individual at a large corporation. If you do not provide the mail stop, the mail is delivered to a central address. Extra effort is required to determine where, within the building, the mail is to be delivered, and the possibility of error increases. For example, if there are two people named Kevin Smith in the same building, the probability is high that each Kevin will receive mail intended for the other one.

Use domains and location-independent addresses as much as possible when you create alias files. To enhance portability and flexibility of alias files, make your alias entries as generic and system-independent as possible. For example, if you have a user named `ignatz` on system `mars`, in domain `Adm.acme.com`, create the alias as `ignatz` instead of `ignatz@Eng` or `ignatz@mars`. If user `ignatz` changes the name of his system, but remains within the engineering domain, you do not need to update any alias files to reflect the change in system name.

When creating aliases that include users outside of your domain, create the alias with the user name and the domain name. For example, if you have a user named `smallberries` on system `privet`, in domain `Mgmt.acme.com`, create the alias as `smallberries@Corp.acme.com`.

You can set an option in the `sendmail.cf` file to translate the email address to a fully qualified domain name when mail goes outside of the user's domain.

Uses for Aliases Files

You create mail aliases for global use in the NIS+ `mail_aliases` table, the NIS `aliases` map, or in local `/etc/mail/aliases` files if your site does not use a naming service. You can also create and administer mailing lists using the same alias files.

Depending on the configuration of your mail services, you may administer aliases by using the NIS or NIS+ naming service to maintain a global `aliases` database, or by updating all of the local `/etc/mail/aliases` files to keep them in sync.

Users can also create and use aliases. They can create aliases either in their local `.mailrc` file, which only they can use, or in their local `/etc/mail/aliases` file, which can be used by anyone. Users cannot create or administer NIS or NIS+ alias files.

Syntax of NIS+ Aliases

The NIS+ aliases table contains the names by which a system or person is known (except for private aliases listed in `.mailrc`). The `sendmail` program can use the NIS+ aliases table instead of the local `/etc/mail/aliases` files to determine mailing addresses. See the reference manual pages for `aliasadm(1M)` and `nsswitch.conf(4)` for more information.

Aliases in the NIS+ aliases table adhere to the following format:

<i>alias:</i>	<i>expansion</i>	<i>[options #</i>	<i>" comments"]</i>
---------------	------------------	--------------------	----------------------

Four columns are described in Table 8-6.

Table 8-6 Columns in the NIS+ Aliases Table

Column	Description
alias	The name of the alias
expansion	The value of the alias as it would appear in a <code>sendmail /etc/aliases</code> file
options	Reserved for future use
comments	Comments about an individual alias

The NIS+ aliases table should contain entries for all mail clients. You can list, create, modify, and delete entries in the NIS+ aliases table with the `aliasadm` command. Or you can use Administration Tool's Database Manager to administer NIS+ mail aliases.

If you are creating a new NIS+ aliases table, you must initialize the table before you create the entries. If the table exists, no initialization is needed. See "How to List the Contents of an NIS+ Aliases Table" on page 257 for information about how to create NIS+ alias tables.

When creating alias entries, enter one alias per line. You should only have one entry that contains the user's system name. For example, you could create the following entries for one user named `ignatz`:

```
ignatz: iggy.ignatz
iignatz: iggy.ignatz
iggyi: iggy.ignatz
iggy.ignatz: ignatz@mars
```

You can create an alias for local names or domains. For example, an alias entry for user `fred` who has a mailbox on the system `mars` and who is in the domain `Planets` could have this entry in the NIS+ aliases table:

```
fred: fred@Planets
```

To use the `aliasadm` command, you must be root, a member of the NIS+ group that owns the aliases table, or the person who created the table.

Syntax of NIS Aliases

Aliases in the NIS `aliases` map adhere to the following format:

```
name: name1, name2, ...
```

Syntax of .mailrc Aliases

Aliases in a `.mailrc` file adhere to the following format:

```
alias aliasname value value value ...
```

Syntax of /etc/mail/aliases Aliases

Distribution list formats in a local `/etc/mail/aliases` file adhere to the following format:

```
aliasname: value,value,value...
```

The aliases in the `/etc/mail/aliases` file are stored in text form. When you edit the `/etc/mail/aliases` file, run the `newaliases` program to recompile the database and make the aliases available in binary form to the `sendmail` program. Or you can use Administration Tool's Database Manager to administer the mail aliases stored in local `/etc` files.

`sendmail` Program

The SunOS 5.x operating system uses the `sendmail`¹ program as a mail router. `sendmail` is responsible for receiving and delivering electronic mail messages. It is an interface between mail reading programs like `mail`, `mailx`, and `mailtool`, and mail transport programs like `uucp`. The `sendmail` program controls email messages that users send, understands the recipients' addresses, chooses an appropriate delivery program, rewrites the addresses in a format that the delivery agent understands, reformats the mail headers as required, and finally passes the transformed message to the mail program for delivery.

Figure 8-6 shows how `sendmail` uses aliases. Programs that read mail, like `/usr/bin/mailx`, can have aliases of their own, which are expanded before the message reaches `sendmail`. Note that this behavior can be changed using `nsswitch.conf`. See the reference manual page for `nsswitch.conf(4)`.

1. See *sendmail*, by Bryan Costales with Eric Allman and Neil Rickert, published by O'Reilly & Associates, Inc., 1993, for a complete explanation of the `sendmail` program.

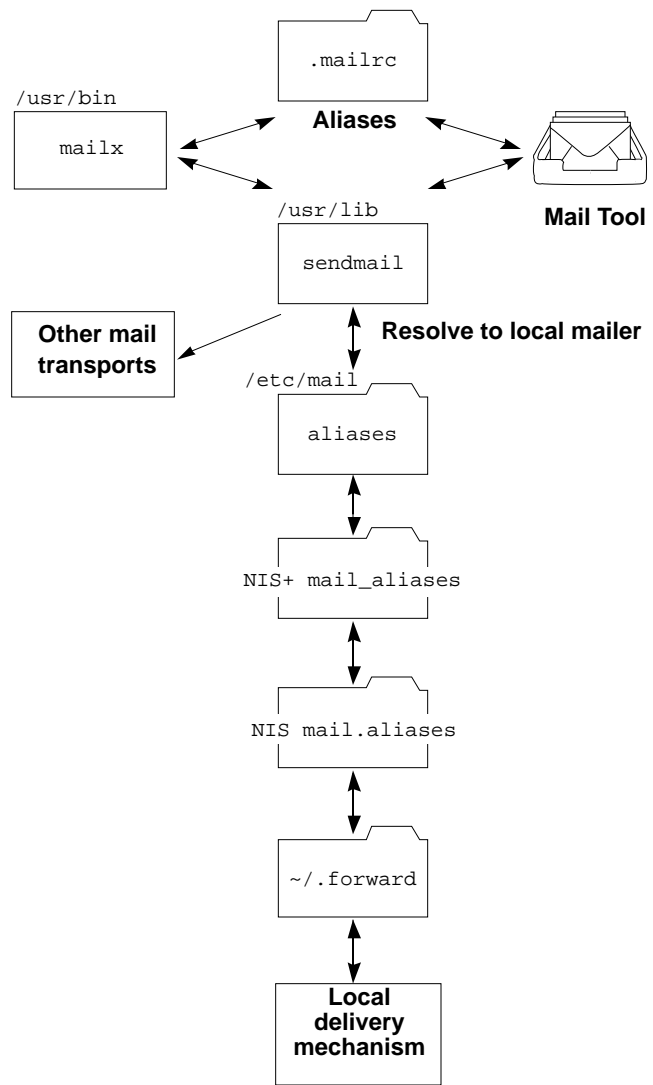


Figure 8-6 How sendmail Uses Aliases

sendmail *Configuration File*

A *configuration file* controls the way that `sendmail` performs its functions. The configuration file determines the choice of delivery agents, address rewriting rules, and the format of the mail header.

The `sendmail` program uses the information from the `/etc/mail/sendmail.cf` file to perform its functions. Each system has a default `sendmail.cf` file installed in the `/etc/mail` directory. You do not need to edit or change the default configuration file for mail servers or mail clients. The only systems that require a customized configuration file are mail hosts, relay hosts, and gateways.

The SunOS 5.x system provides two default configuration files in the `/etc/mail` directory:

- A configuration file, named `main.cf`, for the system (or systems) you designate as the mail host, a relay host, or a gateway
- A configuration file, named `sendmail.subsidiary.cf` (a duplicate copy of the default `sendmail.cf` file)

The configuration file you use on a system depends on the role the system plays in your mail service.

- For mail clients or mail servers, you do not need to do anything to set up or edit the default configuration file.
- To set up a mail host, relay host, or gateway, copy the `main.cf` file and rename it `sendmail.cf` (in the `/etc/mail` directory). Then edit the `sendmail.cf` file to set the relay mailer and relay host parameters needed for your mail configuration.

The following list describes some configuration parameters you may want to change, depending on the requirements of your site:

- Time values specify:
 - How often `sendmail` runs the queue. The interval is typically set to between 15 minutes and 1 hour
 - Read timeouts
 - How long a message remains undelivered in the queue before it is returned to the sender
- Delivery modes specify how quickly mail will be delivered.

- Load limiting prevents wasted time during loaded periods by not attempting to deliver large messages, messages to many recipients, and messages to sites that have been down for a long time.
- Log level specifies what kinds of problems are logged.
- File modes set:
 - User ID (setuid) for sendmail
 - Temporary file modes
 - `/etc/mail/aliases` permissions

sendmail *Configuration Table*

In response to two entries in the `sendmail.cf` file, the `sendmail` program can define macros and classes by looking up values in the `sendmailvars` configuration table. There are two such commands:

- Lines that begin with the `L` key letter are macro definitions, where the values assigned to the specified variable are obtained from the configuration table.
- Lines that begin with the `G` key letter are class definitions, where the values assigned to the specified variable are obtained from the configuration table.

The `L` command has the following syntax:

```
LXsearch_key
```

For example: `Lmmaildomain`

In this case, the search key `maildomain` is used to look up a value in the configuration table to assign to the variable `m`. Most often the single-letter variable name is uppercase, but for internal variables (like `m` for the mail domain name) it is lowercase.

The `G` command has the following syntax:

```
GCsearch_key
```

For example: `GVuucp-list`

In this case, the search key `uucp-list` is used to look up a value in the configuration table to assign to the variable `v`.

In both cases, matching of the search key is case sensitive.

Both commands have counterparts for defining macros or classes within the `sendmail.cf` file, rather than the lookup table. `D` is the counterpart of `L`; `C` is the counterpart of `G`.

If NIS+ is used to administer the network, a global version of the table, `sendmailvars.org_dir`, can be maintained. In addition to the NIS+ table or as an alternative, the table can be maintained in `/etc/mail/sendmailvars` files. The order in which these sources are searched by `sendmail` is controlled by the `sendmailvars` entry in the `/etc/nsswitch.conf` file. By default, the search order is `files nisplus`, which means `sendmail` attempts to look up information in the local table, before going to the NIS+ table.

Entries in an `/etc/mail/sendmailvars` file have the following format:

```
search_key [value1 value2 value3...]
```

The search key may be followed by a Tab or several spaces; values are separated by a single space.

The NIS+ `sendmailvars` table has two columns: a Key column and a Value column. The Value column can have one or more values, each separated by a space. For example:

Key Column	Value Column
maildomain	Eng.acme.COM
uucp-list	acmoon hugo comic

Most mail variables should be defined in the NIS+ table. However, in special cases, systems can override the global setting for a variable by including it in a their local `/etc/mail/sendmailvars` file. The following are such cases:

- A system that has a local `uucp` connection
- A gateway machine between two Internet domains

`.forward` *Files*

Users can create a `.forward` file in their home directories that `sendmail` uses to temporarily redirect mail or send mail to a custom set of programs without bothering a system administrator. When troubleshooting mail problems, particularly problems with mail not being delivered to the expected address, always check the user's home directory for a `.forward` file.

Planning Your Mail System

This section describes four basic types of mail configurations, and briefly outlines the tasks required to set up each configuration. You may find this section useful if you need to set up a new mail system or if you are expanding an existing one. The configurations start with the most basic case (mail completely local, no connection to the outside world) and increase in complexity to a two-domain configuration with a gateway.

To set up a mail system, regardless of its configuration, you need these elements:

- A `sendmail.cf` configuration file on each system
- Alias files with an alias for each user to point to the place where mail is stored
- A mailbox to store (or spool) mail files for each user
- A `postmaster` alias for the person who administers mail services

How you set up the configuration file and the alias file and where you put the mailboxes depend on the configuration you choose.

As system administrator, you should decide on a policy for updating aliases and for forwarding mail messages. You might set up an `aliases` mailbox as a place for users to send requests for mail forwarding and for changes to their default mail alias. If your system uses NIS or NIS+, you can administer forwarding rather than forcing users to manage it themselves.

A common mistake users make is to put a `.forward` file in the home directory of `hosta` that forwards mail to `user@hostb`. When the mail gets to `hostb`, `sendmail` looks up `user` in the NIS or NIS+ aliases and sends the message back to `user@hosta`, resulting in a loop, and more bounced mail.

Local Mail Only

The simplest mail configuration, shown in Figure 8-7, is one mail server with two or more workstations connected to it. Mail is completely local. One system is both the mail server (provides mail spooling for client mailboxes) and the mail host. Mail addresses are parsed using the `/etc/mail/aliases` files. No naming service is used.

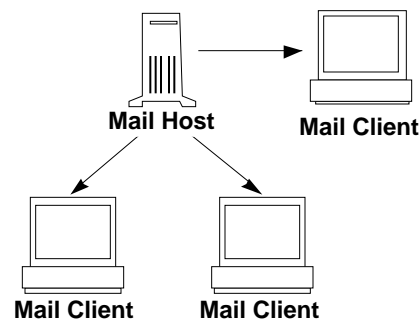


Figure 8-7 Local Mail Configuration

To set up this kind of a local mail configuration, assuming that the mail clients mount their mail files from `/var/mail` on the mail host, you need:

- The default `sendmail.cf` file in the `/etc/mail` directory on each system
- A server designated as the mail host (add `mailhost` to the `/etc/hosts` file on the mail host; then if you are not running NIS or NIS+, add the mail host IP address line to the `/etc/hosts` file of all mail clients)
- Matching `/etc/mail/aliases` files on any system that has a local mailbox
- Entries in each mail client's `/etc/vfstab` file to mount the `/var/mail` directory when mailboxes are located on the mail host

Local Mail and a uucp Connection

The most common mail configuration in a small network is shown in Figure 8-8. One system is the mail server, the mail host, and the relay host to the outside world. Mail is distributed using the `/etc/mail/aliases` files. No naming service is required.

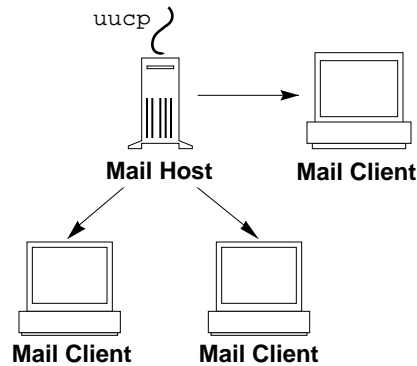


Figure 8-8 Local Mail Configuration With a uucp Connection

To set up this kind of a mail configuration, assuming that the mail clients mount their mail files from `/var/mail` on the mail host, you need:

- The `main.cf` file on the mail host (you must edit the file to select a major relay mailer)
- The default `sendmail.subsidiary.cf` file on each mail client system (no editing required)
- A server designated as the mail host (add `mailhost` to the `/etc/hosts` file on the mail host; if you are not running NIS or NIS+, add the mail host IP address line to the `/etc/hosts` file of all mail clients)
- Matching `/etc/mail/aliases` files on any system that has a local mailbox
- Entries in each mail client's `/etc/vfstab` file to mount the `/var/mail` directory when mailboxes are located on the mail host

One Domain, Two Networks, and a Router

The mail configuration shown in Figure 8-9 has one domain, two networks, and a router. In this configuration, the mail server, the mail host, and the relay host (or hosts) are likely to be different systems. To make the process of administering and distributing mail easier, a naming service is used.

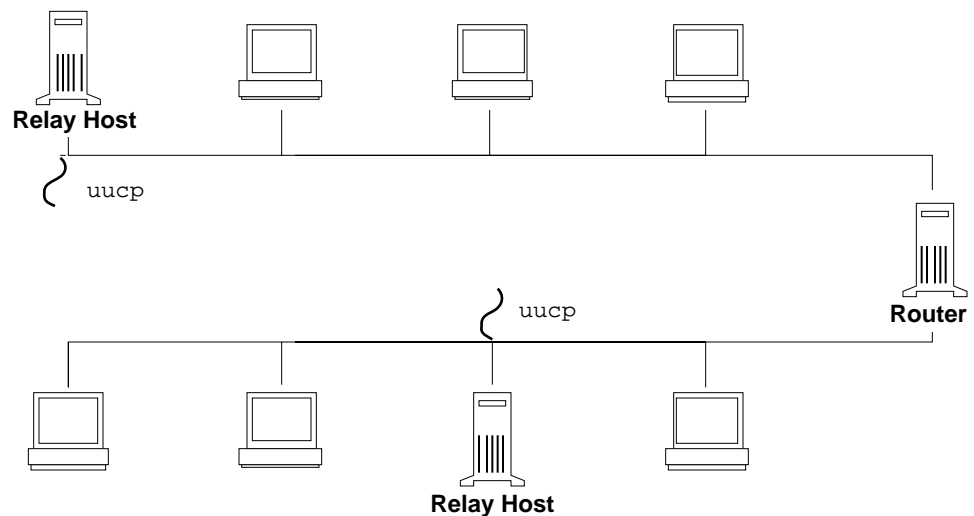


Figure 8-9 One Domain, Two Networks, With a Router and Multiple `uucp` Connections

To set up this kind of a mail configuration, assuming that the mail clients may have local or remote `/var/mail` files, you need everything specified in Table 8-7.

Table 8-7 Requirements for a One Domain, Two Network Mail Configuration

Category	Requirements
Relay host	The <code>main.cf</code> file on the relay hosts—the systems with <code>uucp</code> connections. You must edit the file to select a major relay connector. You may want to define a mail relay host that knows about all connections. Special rules added to the <code>sendmail.cf</code> file can help but are not mandatory.
Mail host	One system designated as the mail host (add mail host to the <code>/etc/hosts</code> file on the mail host system).

Table 8-7 Requirements for a One Domain, Two Network Mail Configuration (Continued)

Category	Requirements
Mail server	Adequate spooling space for client mailboxes.
Mail client	The <code>sendmail.subsidiary.cf</code> file on each mail client system (no editing required). Entries in each mail client's <code>/etc/vfstab</code> file to mount the <code>/var/mail</code> directory.
NIS+ tables	<code>mail_aliases.org_dir</code> tables for NIS+ with a mail alias entry for all users to point to where their mail is stored.

Two Domains and a Gateway

The mail configuration shown in Figure 8-10 has two domains and a gateway. In this configuration, the mail server, the mail host, and the relay host (or hosts) for each domain are likely to be different systems. To make the process of administering and distributing mail easier, a naming service is used.

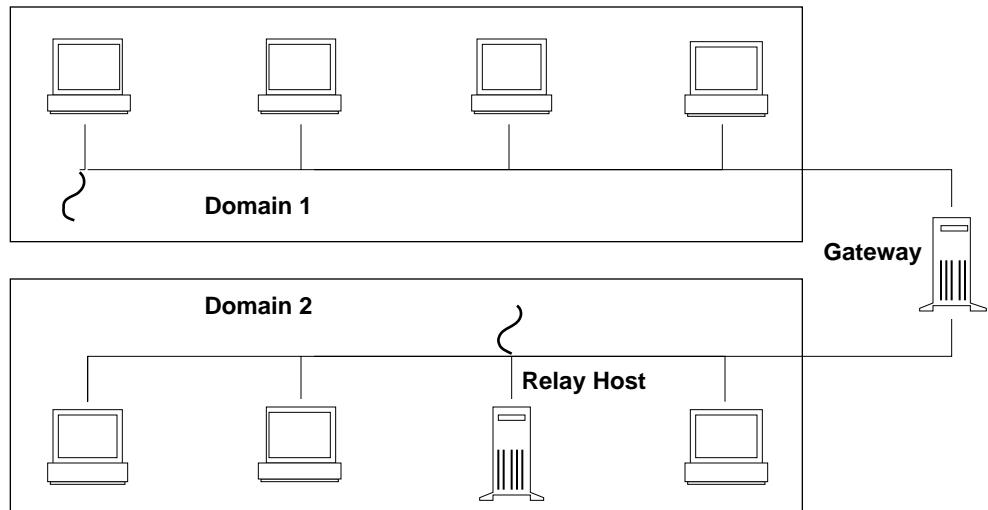


Figure 8-10 Two Domains and a Gateway

Table 8-8 lists the requirements for this mail configuration.

Table 8-8 Requirements for Mail Configurations With Two Domains and a Gateway

Category	Requirements
Gateway	Complex gateway systems usually need a customized <code>sendmail.cf</code> file with special rules added.
Relay host	<p>The <code>main.cf</code> file on the relay hosts—the systems with <code>uucp</code> connections. (You must edit the file to select a major relay mailer.)</p> <p>It might be useful to define a mail relay host with information about all connections. (Special rules added to the <code>sendmail.cf</code> file can help but are not mandatory.)</p> <p>NOTE: The <code>sendmail.subsidiary.cf</code> file has an entry (the CV class) that you can use to define “local” <code>uucp</code> connections (like the system at the left of Domain 1 in Figure 8-10). If you define such a local <code>uucp</code> connection, users must address mail using the format, <code>uucphost!remote-system!address</code>.</p>
Mail host	One system designated as the mail host (add <code>mailhost</code> to the <code>/etc/hosts</code> file on the mail host system).
Mail server	Adequate spooling space for client mailboxes.
Mail client	<p>The <code>sendmail.cf</code> file on each mail client system (no editing required).</p> <p>Entries in each mail client’s <code>/etc/vfstab</code> file to mount the <code>/var/mail</code> directory.</p>
NIS+ tables	<code>mail_aliases.org_dir</code> tables for NIS+ with a mail alias entry for each user to point to where the mail is stored.

An Overview of the Mail Service

This section describes the directory structure and files of the mail service and explains how the `sendmail` program and mail addressing work.

Structure of the Mail Service

Files for the mail service are located in three directories: `/bin`, `/etc/mail`, and `/usr/lib`. User’s mailboxes are located in the `/var/mail` directory.

Table 8-9 shows the contents of the `/bin` directory that are used for mail services.

Table 8-9 Contents of the `/bin` Directory Used for Mail Services

Name	Type	Description
<code>mail</code>	File	A user agent
<code>mailcompat</code>	File	A filter to store mail in SunOS 4.1 mailbox format
<code>mailq</code>	Link	Link to <code>/usr/lib/sendmail</code>
<code>mailstats</code>	File	A file used to store mail statistics generated by the <code>/etc/mail/sendmail.st</code> file (if present)
<code>mailx</code>	File	A user agent
<code>newaliases</code>	Link	Link to <code>/usr/lib/sendmail</code>

Table 8-10 shows the contents of the `/etc/mail` directory.

Table 8-10 Contents of the `/etc/mail` Directory

Name	Type	Description
<code>Mail.rc</code>	File	Default settings for the <code>mailtool</code> user agent
<code>aliases</code>	File	Mail forwarding information
<code>aliases.dir</code>	File	Binary form of mail forwarding information (created by running <code>newaliases</code>)
<code>aliases.pag</code>	File	Binary form of mail forwarding information (created by running <code>newaliases</code>)
<code>mailx.rc</code>	File	Default settings for the <code>mailx</code> user agent
<code>newaliases</code>	File	Command that creates the binary form of the <code>aliases</code> file
<code>main.cf</code>	File	Sample configuration file for main systems
<code>sendmail.cf</code>	File	Configuration file for mail routing
<code>sendmail.fc</code>	File	Frozen configuration file
<code>sendmail.hf</code>	File	Help file used by the <code>SMTP HELP</code> command

Table 8-10 Contents of the `/etc/mail` Directory (Continued)

Name	Type	Description
<code>sendmail.st</code>	File	The <code>sendmail</code> statistics file; if this file is present, <code>sendmail</code> logs the amount of traffic through each mailer
<code>sendmailvars</code>	File	Table that stores macro and class definitions for lookup from <code>sendmail.cf</code>
<code>sendmailvars.org_dir</code>	Table	NIS+ version of <code>sendmailvars</code> table
<code>sendmail.subsidiary.cf</code>	File	Sample configuration file for subsidiary systems

Table 8-11 shows the contents of the `/usr/lib` directory.

Table 8-11 Contents of the `/usr/lib` Directory

Name	Type	Description
<code>sendmail</code>	File	The routing program, also known as the mail transport agent
<code>sendmail.mx</code>	File	Mail-routing program linked with the domain name service

Spooling directories for delivered mail reside in the `/var/mail` directory, as shown in Table 8-12.

Table 8-12 Contents of the `/var/mail` Directory

Name	Type	Description
<code>mqueue</code>	Directory	Where undelivered mail is stored
<code>mailbox1</code> , <code>mailbox2</code>	File	Mailboxes for delivered mail

How the Mail Service Works

Figure 8-11 shows how `sendmail` interacts with the other programs in the mail system.

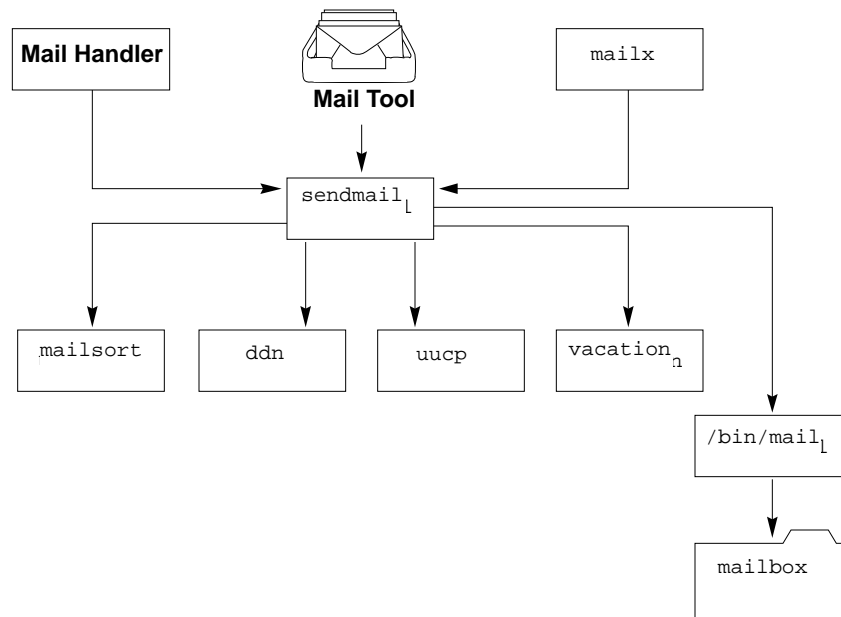


Figure 8-11 How the Mail Service Works

The user interacts with a mail-generating and sending program. When the mail is submitted, the mail-generating program calls `sendmail`, which routes the message to the correct mailers.

How `sendmail` Works

The `sendmail` program collects a message from a program like `mailx` or `mailtool`, edits the message header as required by the destination mailer, and calls appropriate mailers to do delivery or queuing for network transmission.

Note – The `sendmail` program never edits or changes the body of a message. Any changes that it makes to interpret email addresses are made only in the header of the message.

Argument Processing and Address Parsing

When `sendmail` receives input, it collects recipient names (either from the command line or from the SMTP protocol) and generates two files. One is an *envelope* that contains a list of recipients. The other file contains the header and the body of the message. The `sendmail` program expands aliases, including mailing lists, and validates as much as possible of the remote recipient. Then `sendmail` checks syntax and verifies local recipients. Detailed checking of host names is deferred until delivery. As local recipients are verified, messages are forwarded to them.

After parsing the recipient lists, `sendmail` appends each name to both the envelope and the header of the message. When a name is “aliased” or forwarded, it retains the old name in the list and sets a flag to tell the delivery phase to ignore this recipient. The lists are kept free from duplicates, preventing alias loops and duplicate messages delivered to the same recipient, which can occur if a recipient is in two different alias groups.

Note – Users may receive duplicate copies of the same message when alias lists contain email addresses for the same person using different syntax. The `sendmail` program cannot always match the email addresses as duplicates of one another.

Message Collection

The `sendmail` program then collects the message. The message should have a header at the beginning. The header and the body of the message must be separated by a blank line. No formatting requirements are imposed on the message body except that they must be lines of text. `sendmail` stores the header in memory, and stores the body of the message in a temporary file. To simplify the program interface, the message is collected, even if no names are valid. The message is returned to the sender with an error.

Note – With multimedia Mail Tool, users can transmit binary data. However, it must be encoded by Mail Tool. The `sendmail` program does not automatically encode binary data. Refer to the Mail Tool documentation for information about how to encode and decode binary data messages.

Message Delivery

For each unique mailer and host in the recipient list, `sendmail` calls the appropriate mailer. The mailer sends the message to all recipients on the host.

The `sendmail` program sends the message to the mailer using one of the same interfaces used to submit a message to `sendmail` (using the conventional UNIX argument vector/return status, communicating over a pair of UNIX pipes, or using SMTP over a TCP connection). Each copy of the message has a customized header.

Error Handling

When mail can't be delivered, the mailer catches and checks the status code, and a suitable error message is given as appropriate. The exit code must conform to a system standard. If a nonstandard exit code is used, `sendmail` transmits the message, `Services unavailable`.

Queueing for Retransmission

When the mailer returns a status that shows it might be able to handle the mail later (for example, the next host is down, or the phone is busy for `uucp`), `sendmail` queues it and tries again later.

Return to Sender

If errors occur during processing, `sendmail` returns the message to the sender for retransmission. The letter may be mailed back or written to the `dead.letter` file in the sender's home directory.

How Mail Addressing Works

Assuming that you are using the default rule set in the `sendmail.cf` file, the following examples show the route an email message takes, depending on the way it is addressed.

Mail within a domain addressed with only the user's login name goes to the `aliases` file on the mail host (or to the `aliases` database), and is sent to the address found in the database. In the example shown in Figure 8-12, mail addressed to user `ignatz` goes to the mail host, and is forwarded to the host named `mars`.

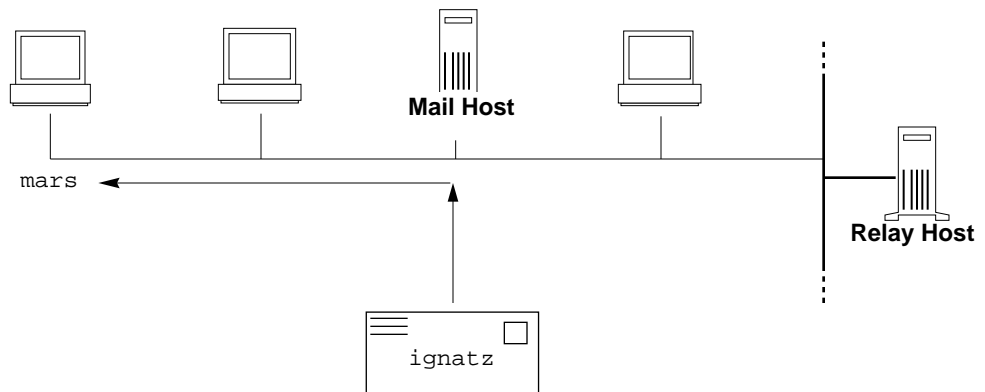


Figure 8-12 Delivery Path for Mail Addressed With User Name Only

Mail within a domain addressed with the user's login name and host name goes directly to the host system without any additional processing. In the example shown in Figure 8-13, mail addressed to user `ignatz` at the host named `mars` goes directly to the host named `mars`.

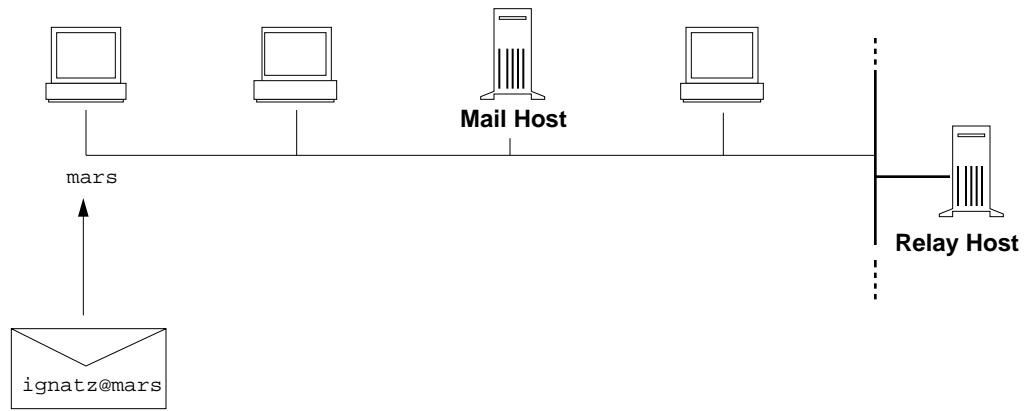


Figure 8-13 Delivery Path for Mail Addressed With User and Host Names

Mail within a domain addressed with the user's login name and domain name goes to the `aliases` file on the mail host (or to the `aliases` database). If the mail host has an alias, it redirects the message to the host system. In the example shown in Figure 8-14, mail addressed to user `ignatz@Lab` goes to the mail host and is then forwarded to the host named `mars`.

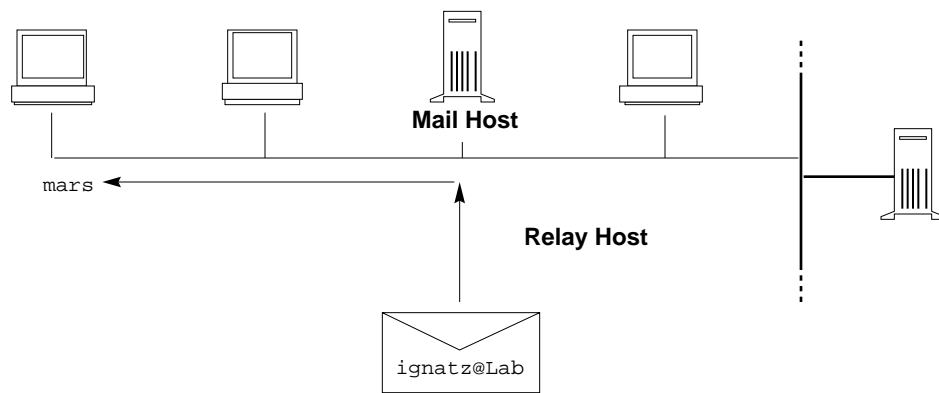


Figure 8-14 Delivery Path for Mail Addressed With User and Domain Names

Mail addressed with the user's name and a fully qualified domain name goes to the mail host, which sends it to the relay host. The relay host sends the message to the host system.

When the mail comes from within the recipient's domain, however, the mail host recognizes the domain name and does not send the message to the relay host. In the example shown in Figure 8-15, mail addressed to user `ignatz@Lab.Acme.com` from outside the engineering domain goes to the sender's mail host, the sender's relay host, and is then forwarded to the recipient's relay host, the recipient's mail host, and then to the host named `mars`.

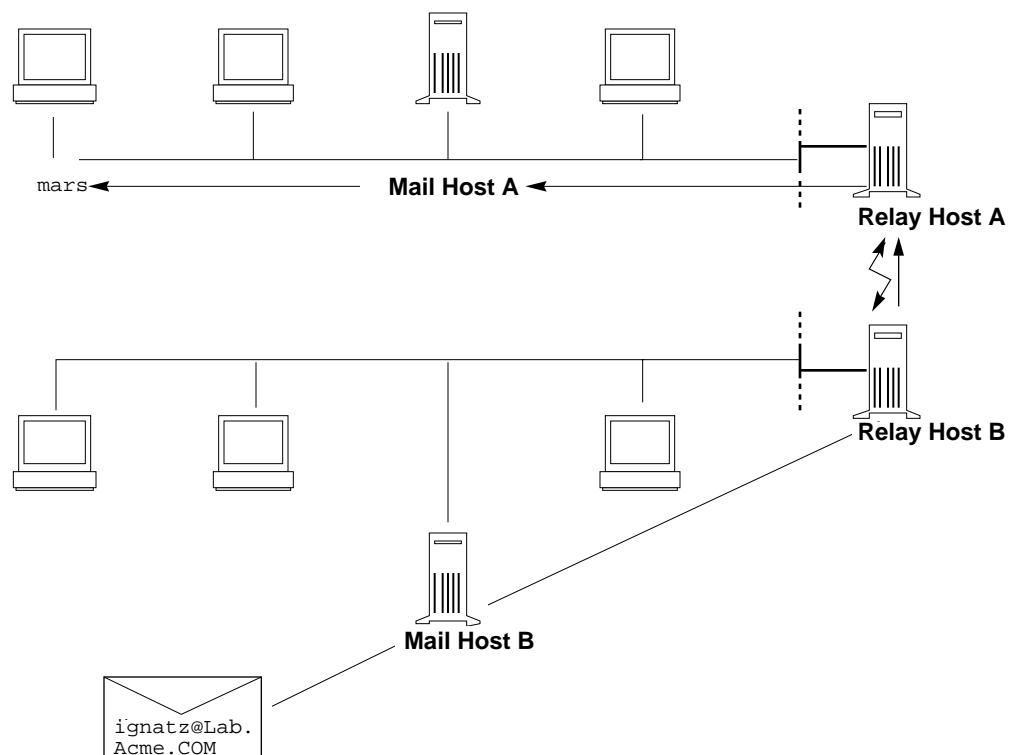


Figure 8-15 Delivery Path for Mail Addressed With User Name and Fully Qualified Domain Name

When a recipient responds to a message that was sent within a domain running neither NIS or NIS+, and with no `/etc/aliases` files on each system, using only the login name as an address, the sender's host name is appended to the address. If a recipient responds to such a message, the message is routed to the named system, then to the mail host. If the named host system is not available, the message is delayed until the host responds, or the message bounces after three days of trying. In the example shown in Figure 8-16, if `ignatz@mars` sent a copy to `fred`, and the recipient uses the information in the message header to reply, a message to `fred@mars` goes first to `mars`, then to the mail host, and finally to `fred@venus`, Fred's real mail destination.

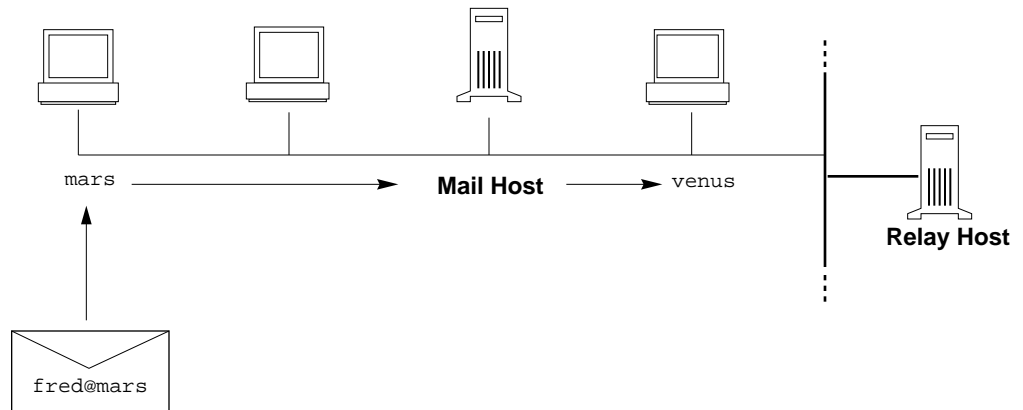


Figure 8-16 Delivery Path for Mail Addressed With User Name and Sender's Host Name

Setting Up and Administering Mail Services

9

This chapter describes how to set up and administer mail services.

Use the following table to find the page where the instructions for specific tasks.

<i>How to Set Up a Mail Server</i>	<i>page 254</i>
<i>How to Set Up a Mail Client</i>	<i>page 254</i>
<i>How to Set Up a Mail Host</i>	<i>page 256</i>
<i>How to Set Up a Relay Host</i>	<i>page 256</i>
<i>How to List the Contents of an NIS+ Aliases Table</i>	<i>page 257</i>
<i>How to Add Entries by Editing an NIS+ Aliases Table</i>	<i>page 259</i>
<i>How to Change Entries in an NIS+ Aliases Table</i>	<i>page 259</i>
<i>How to Delete Entries from a NIS+ Aliases Table</i>	<i>page 260</i>
<i>How to Set Up NIS Aliases Files</i>	<i>page 260</i>
<i>How to Set Up Local Mail Aliases Files</i>	<i>page 261</i>
<i>How to Set Up DNS Aliases Files</i>	<i>page 263</i>
<i>How to Set Up the Postmaster Alias</i>	<i>page 264</i>
<i>How to Test the Mail Configuration</i>	<i>page 265</i>
<i>How to Print the Queue</i>	<i>page 268</i>
<i>How to Force the Queue</i>	<i>page 268</i>
<i>How to Run the Old Mail Queue</i>	<i>page 269</i>

<i>How to Check Aliases</i>	<i>page 272</i>
<i>How to Test the sendmail Program</i>	<i>page 272</i>
<i>How to Verify Connections to Other Systems</i>	<i>page 273</i>

If you are not familiar with administering mail services, read Chapter 8, “Understanding Mail Services,” for an introduction to the terminology and structure of the mail services, and for descriptions of several mail services configurations.

Setting Up Mail Services

You can set up a mail service relatively easily if your site does not provide connections to electronic mail (email) services outside your company, or if your company is in a single domain.

Appendix C, “Customizing sendmail Configuration Files,” contains information about how to create more complicated configuration files.

Mail requires three types of configurations for local mail and a fourth for communication with networks outside of your domain. These configurations can be combined on the same system or provided by separate systems. There is also a fifth, optional, type of mail configuration called a *gateway*. You need to set up systems on your site to perform the functions described in Table 9-1.

Table 9-1 Mail Configurations

Configuration	Description
Mail server	You need to have at least one mail server. The mail server stores mailboxes in the <code>/var/mail</code> directory.
Mail host	You need at least one mail host. The mail host resolves difficult email addresses and reroutes mail within your domain.

Table 9-1 Mail Configurations (Continued)

Configuration	Description
Mail client	Mail clients are users who have mailboxes either locally or on a mail server.
Relay host	A relay host manages communication with networks outside of your domain.
Gateway	A gateway is a connection between differing communications networks. A relay host may also act as a gateway. You must add rules to the <code>sendmail.cf</code> file to set up a gateway. See Appendix C, “Customizing sendmail Configuration Files,” for information about adding rules. If you have to set up a gateway, you should find a gateway configuration file that is close to what you need, and modify it to fit your situation.

Before you begin to set up your mail service, choose the systems to act as mail servers, mail hosts, and relay hosts. You should also make a list of all the mail clients for which you will be providing service, and include the location of their mailboxes. This list will help you when you are ready to create mail aliases for your users. See Chapter 8, “Understanding Mail Services,” for more information about the function each of these systems provides. For your convenience, guidelines about which systems are good candidates for mail server, mail host, and relay host are repeated in the following sections.

To simplify the setup instructions, this chapter tells you what you need to do to set up individual mail servers, mail hosts, mail clients, and relay hosts. If a system in your mail services configuration is acting in more than one capacity, follow the appropriate instructions for each type of system. For example, if your mail host and mail server functions are on the same system, follow the directions for setting up that system as a mail host and then follow the directions for setting up the same system as a mail server.

Note – The following procedures for setting up a mail server and mail client apply when mailboxes are NFS-mounted. However, typically, mailboxes are maintained in locally mounted `/var/mail` directories, in which case, the following procedures are not needed.

Setting Up Mail Security

By default, security permissions on a `/var/mail` directory allow read, write, and execute access to the owner, members of groups to which the owner belongs, and all others. On Solaris 2.x mail servers, you can make a `/var/mail` directory more secure by changing its default permissions to allow only read and write access to anyone outside the owner's groups, as long as no SunOS 4.1.x mail clients are connected. For more information about changing directory permissions, refer to *Security, Performance, and Accounting Administration*.

▼ How to Set Up a Mail Server

The mail server routes all mail from a client. The only resource requirement for a mail server is that it has adequate spooling space for client mailboxes. See Chapter 8, "Understanding Mail Services," for recommendations about spooling space.

For this task, you will check the `/etc/dfs/dfstab` file to be sure the `/var` directory is exported.

1. Type `share` and press Return.

If the `/var` directory is not exported, continue with the next step.

2. Type `share -F nfs [-o options] /var/mail` and press Return.

Note - The `sendmail` program automatically creates mailboxes in the `/var/mail` directory the first time a message is delivered. You do not need to create individual mailboxes for your mail clients.

▼ How to Set Up a Mail Client

A mail client is a user of mail services, with a mailbox either on a local system or on a mail server, and a mail alias in the `/etc/mail/aliases` file that points to the location of the mailbox.

To set up a mail client with a mailbox on a mail server:

1. Become root on the mail client's system.

2. Create a `/var/mail` mount point on the mail client's system.

3. **Edit the `/etc/vfstab` file and add an entry for the `/var/mail` directory on the mail server, mounting it on the local `/var/mail` directory.**
The client's mailbox will be automatically mounted any time the system is rebooted.
4. **Type `mountall` to mount the mailbox.**
The client's mailbox is mounted.
5. **Use the Administration Tool to edit the `/etc/hosts` file and add an entry for the mail server.**
6. **Add the client to the `/etc/mail/aliases` file.**
See "Creating Mail Aliases" on page 257 for information about how to create mail aliases for different types of mail configurations.

Note - The `sendmail` program automatically creates mailboxes in the `/var/mail` directory the first time a message is delivered. You do not need to create individual mailboxes for your mail clients.

▼ How to Set Up a Mail Host

A mail host resolves email addresses and reroutes mail within your domain. A good candidate for a mail host is a system that connects your systems to the outside world or to a parent domain.

1. **Become root on the mail host system.**
2. **Use the Administration Tool to edit the `hosts` database.**
Add the word `mailhost` following the IP address and system name of the mail host system. The system is designated as a mail host.
3. **Use the Administration Tool to create an entry for the new mail host in the `hosts` database on the master NIS server.**
If you are not using NIS, you must create an entry in the `/etc/hosts` database for each system on the network.

The entry should use this format: *IP address mailhost_name mailhost*

4. **Type `cp /etc/mail/main.cf /etc/mail/sendmail.cf` and press Return.**
This copies and renames the `/etc/mail/main.cf` file.

5. Reboot the mail host and test your mail configuration.

See “Testing the Mail Configuration” on page 265 for information.

▼ How to Set Up a Relay Host

A relay host manages communication with networks outside of your domain that use the same relay mailer. The mailer on the sending relay host must match the mailer on the receiving system.

A good candidate for a relay host is a system attached to Ethernet and to phone lines, or a system configured as a router to the Internet. You may want to configure the mail host as the relay host, or configure another system as relay host. You may choose to configure more than one relay host for your domain. Each relay host you configure must use a mailer that matches the mailer on the connecting system. If you have `uucp` connections, you should configure the system (or systems) with `uucp` connections as the relay host.

1. Become root on the relay host system.

2. Type `cp /etc/mail/main.cf /etc/mail/sendmail.cf` and press **Return**.

This command copies and renames the `main.cf` file.

3. Edit the `/etc/mail/sendmail.cf` file and make the following changes:

a. Only if your relay mailer is not `uucp`, change the default entry `DMsmartuucp` to the entry that is appropriate for your relay mailer.

Available mailers are `smartuucp` (the default), `ddn`, `ether`, and `uucp`. If your relay mailer is `uucp`, you do not need to change this entry.

You can specify a different relay mailer for each relay host (if appropriate). You can define rule sets for other relay mailers in the `main.cf` file. See “Mailers” on page 220 for a description of each of the default relay mailers.

b. In the entry `DRddn-gateway`, replace `ddn-gateway` with the name of your relay host.

The `DR` entry defines the relay host.

c. In the entry `CRddn-gateway`, replace `ddn-gateway` with the name of your relay host.

The `CR` entry defines the class of the relay host. You can designate one or more hosts as a member of this class.

d. (Optional) Add a `Dmmail_domain` or `Lmmaildomain` entry to define the mail domain name to be used for mail delivery.

The `m` macro defines the mail domain name. If the macro is not defined, the naming service domain name is used with the first component stripped off. For example, `Ecd.East.Acme.COM` becomes `East.Acme.COM`. If you use the `L` command, `sendmail` will look up the name to use in the `sendmailvars` table, using `maildomain` as the search key.

e. Save the edits.

4. Reboot the mail host and test your mail configuration.

See “Testing the Mail Configuration” on page 265 for information.

Creating Mail Aliases

You can use the `aliasadm` command to create mail aliases for a user. Mail aliases must be unique within the domain. This section tells you how to use command lines to search the mail aliases table for aliases, and to create mail aliases for NIS+, NIS, DNS, or on the local system.

Or you can use the Administration Tool's Database Manager application to perform these tasks on the Aliases database.

▼ How to List the Contents of an NIS+ Aliases Table

To use the `aliasadm` command, you must be either root, a member of the NIS+ group that owns the aliases table, or the person who created the table.

To list the entire contents of the NIS+ aliases table:

◆ **Type `aliasadm -l` and press Return.**

This lists the contents of the aliases table in alphabetic order by alias.

Note – If you have a large aliases table, listing the entire contents can take some time. If you are searching for a specific entry, pipe the output through the `grep` command (`aliasadm -l | grep entry`) so that you can use the `grep` search capability to find specific entries.

To list individual entries in the NIS+ aliases table:

- ◆ **Type** `aliasadm -m alias` **and press Return.**
The alias entry is listed.

```
saturn# aliasadm -m ignatz
ignatz:ignatzsaturn@#Alias for Iggy Ignatz
```

Note - The `aliasadm -m` option matches only the complete alias name. It does not match partial strings. You cannot use metacharacters (like `*`, and `?`) with the `aliasadm -m` option. If you are interested in partial matches, try typing `aliasadm -l | grep partial-string` and pressing Return.

▼ **How to Add Aliases to a NIS+ Aliases Table from the Command Line**

If you are creating a completely new NIS+ aliases table, you first must initiate the NIS+ table.

To initiate an NIS+ table:

- ◆ **Type** `aliasadm -I` **and press Return.**

To add aliases to an NIS+ table from the command line:

1. **Compile a list of each of your mail clients, the locations of their mailboxes, and the names of the mail server systems.**
2. **Become root on any system.**
3. **For each alias, type** `aliasadm -a alias expanded_alias [options comments]` **and press Return.**
This adds the aliases to the NIS+ aliases table.
4. **Type** `aliasadm -m alias` **and press Return.**
This displays the entry you created.
5. **Check the entry to be sure it is correct.**

Example of Adding Aliases to NIS+ From the Command Line

```
# aliasadm -a iggy iggy.ignatz@saturn "Iggy Ignatz"
```

▼ How to Add Entries by Editing an NIS+ Aliases Table

If you are adding more than two or three aliases, you may want to edit the NIS+ table directly.

- 1. Compile a list of each of your mail clients, the locations of their mailboxes, and the names of the mail server systems.**
- 2. Become root on any system.**
- 3. Type `aliasadm -e` and press Return.**
The aliases table is displayed using the editor set with the `$EDITOR` environment variable. If the variable is not set, the `vi` editor is used.
- 4. Type each alias on a separate line, using these formats:**
 - a. Enter the aliases in any order, at any place in the table.**
The order is not important to the NIS+ aliases table. The `aliasadm -l` command sorts the list and displays them in alphabetical order.
 - b. Use the format `alias: expanded_alias# ["option" # "comments"]`**
If you leave the option column blank, enter an empty pair of quotation marks (" ") and then add the comments.
 - c. End each line by pressing Return.**
- 5. Check that the entries are correct.**
- 6. Save the changes.**

▼ How to Change Entries in an NIS+ Aliases Table

- 1. Become root on any system.**
- 2. Type `aliasadm -m alias` and press Return.**
The information for the alias is displayed.

3. **Type** `aliasadm -c alias expanded_alias [options comments]` **and press Return.**
The alias is changed using the new information you provide.
4. **Type** `aliasadm -m alias` **and press Return.**
The entry you created is displayed.
5. **Check the entry to be sure it is correct.**

▼ How to Delete Entries from a NIS+ Aliases Table

1. **Become root on any system.**
2. **Type** `aliasadm -d alias` **and press Return.**
The alias is deleted from the NIS+ aliases table.

▼ How to Set Up NIS Aliases Files

The `/etc/mail/aliases` file on an NIS master contains all names by which a system or person is known. The NIS master is searched if there is no match in the local `/etc/mail/aliases` files. The `sendmail` program uses the NIS master file to determine mailing addresses. See the reference manual page for `aliases(4)`.

The `/etc/mail/aliases` file on the NIS master should contain entries for all mail clients. You can either edit the file on each system or edit the file on one system and copy it to each of the other systems.

Aliases are in the following form:

```
name: name1, name2, ...
```

You can use aliases for local names or domains. For example, an alias entry for user `fred` who has a mailbox on the system `saturn` and who is in the domain `Planets` would have this entry in the `/etc/mail/aliases` file:

```
fred: fred@Planets
```

To set up NIS mail aliases files:

1. Compile a list of each of your mail clients, the locations of their mailboxes, and the names of the mail server systems.
2. Become root on the NIS master server.
3. Edit the `/etc/mail/aliases` file, and make the following entries:
 - a. Add an entry for each mail client.
 - b. Change the entry `Postmaster: root` to the mail address of the person who is designated as postmaster.
See “How to Set Up the Postmaster Alias” on page 264 for more information.
 - c. If you have created a mailbox for administration of a mail server, create an entry for `root: mailbox@mailserver`.
 - d. Save the changes.
4. Edit the `/etc/hosts` file on the NIS master server and create an entry for each mail server.
5. Type `cd /var/yp` and press Return.
6. Type `make` and press Return.
The changes in the `/etc/hosts` and `/etc/mail/aliases` files are propagated to NIS slave systems. It takes a few minutes, at most, for the aliases to take effect.

▼ **How to Set Up Local Mail Aliases Files**

The `/etc/mail/aliases` file on a local system contains all names by which a system or person is known. The `sendmail` program uses this file to determine mailing addresses. See the reference manual page for `aliases(4)`.

If your network is not running a naming service, the `/etc/mail/aliases` file of each system should contain entries for all mail clients. You can either edit the file on each system, or edit the file on one system and copy it to each of the other systems.

Aliases are of the form:

```
name: name1, name2, ...
```

You can create aliases for only local names—a current host name or no host name. For example, an alias entry for user `ignatz` who has a mailbox on the system `saturn` would have this entry in the `/etc/mail/aliases` file:

```
ignatz: ignatz@saturn
```

It is a good idea to create an administrative account for each mail server. You do this by assigning `root` a mailbox on the mail server and adding an entry to the `/etc/mail/aliases` file for `root`. For example, if the system `saturn` is a mailbox server, add the entry `root: sysadmin@saturn` to the `/etc/mail/aliases` file.

To set up local mail aliases files:

- 1. Compile a list of each of your mail clients and the locations of their mailboxes.**
- 2. Become root on the mail server.**
- 3. Edit the `/etc/mail/aliases` file, and make the following entries:**
 - a. Add an entry for each mail client.**
 - b. Change the entry `Postmaster: root` to the mail address of the person who is designated as postmaster.**
See “How to Set Up the Postmaster Alias” on page 264 for more information.
 - c. If you have created a mailbox for administration of a mail server, create an entry for `root: mailbox@mailserver`.**
 - d. Save the changes.**
- 4. Type `newaliases` and press Return.**
This creates an alias file in binary form that `sendmail` can use. The file is stored in the `/etc/mail/aliases.dir` and `/etc/mail/aliases.pag` files.

5. Copy the `/etc/mail/aliases`, the `/etc/mail/aliases.dir`, and `/etc/mail/aliases.pag` files to each of the other systems.

When you copy all three files, you do not need to run the `newaliases` command on each of the other systems.

You can copy the files by using the `rcp` or `rdist` commands, or by using a script that you create for this purpose. Remember that you must update all the `/etc/mail/aliases` files each time you add or remove a mail client.

▼ How to Set Up DNS Aliases Files

The DNS naming service does not use aliases for individuals. It does use aliases for hosts or domains, called *mail exchange (MX) records*. These *mail exchange records* are set in the `/etc/named.boot` file on the DNS servers for the domain or subdomain. You can specify host names, or domain names, or both in this file. Domain names can contain wildcards. For example, `*.acme.com` is an acceptable domain name. See *Name Services Administration Guide* for more information about administering DNS.

You must use the `sendmail.mx` program with the DNS naming service. When you use the `sendmail.mx` program, the `/${name}` rule, which creates fully qualified host names, is activated in the `sendmail.cf` file. You do not need to edit the configuration file to activate this rule.

To set up the `sendmail.mx` program:

1. Type `mv /usr/lib/sendmail /usr/lib/sendmail.nomx` and press Return.

This renames the current `sendmail` program.

2. Type `mv /usr/lib/sendmail.mx /usr/lib/sendmail` and press Return.

The `sendmail.mx` file gets host names directly from DNS.

3. Check for a `mailhost` entry.

Make sure there is an entry for `mailhost` in both the DNS `/etc/named.boot` file on the DNS server (or servers) and in the NIS database or NIS+ mail aliases table.

▼ How to Set Up the Postmaster Alias

Every system should be able to send mail to a `postmaster` mailbox. You can create an NIS or NIS+ alias for `postmaster`, or create one in each local `/etc/mail/aliases` file. Here is the default `/etc/mail/aliases` entry:

```
# Following alias is required by the mail protocol, RFC 822
# Set it to the address of a HUMAN who deals with this system's
# mail problems.
Postmaster: root
```

To create the `postmaster` alias, edit each system's `/etc/mail/aliases` file and change `root` to the mail address of the person who will act as postmaster.

You may want to create a separate mailbox for the postmaster to keep postmaster mail separate from personal mail. If you create a separate mailbox, use the mailbox address instead of the postmaster's mail address when you edit the `/etc/mail/aliases` files.

To create a separate mailbox for `postmaster`:

1. Create a user account for the person designated as `postmaster`, and put an asterisk (*) in the password field.
2. Type `mail -f postmaster` and press Return.
The mail program will be able to read and write to the mailbox name.

To add the `postmaster` mailbox to the aliases:

1. Become root and edit the `/etc/mail/aliases` file on each system.
If your network runs NIS or NIS+, edit the `/etc/mail/aliases` file.
2. Change the `postmaster` alias from `root` to
`Postmaster: postmastermailbox@postmasterhost` and save the changes.
3. On the postmaster's local system create an entry in the `/etc/mail/aliases` file that defines the name of the alias (`sysadmin`, for example), and includes the path to the local mailbox.
4. Type `newaliases` and press Return.

Or you could change the `Postmaster:` entry in the `aliases` file to a `postmaster: /usr/somewhere/somefile` entry.

Testing the Mail Configuration

When you have set up all the systems in your mail configuration, use the suggestions in this section to test the setup to be sure mail messages can be sent and received.

▼ How to Test the Mail Configuration

1. Reboot any system for which you have changed a configuration file.

2. Send test messages from each system by typing

```
/usr/lib/sendmail -v </dev/null names and press Return.
```

Specify a recipient's email address in place of the *names* variable.

This command sends a null message to the specified recipient, and displays messages while it runs.

3. Run the following tests:

a. Send mail to yourself or other people on the local system by addressing the message to a regular user name.

b. If you are on Ethernet, send mail to someone on another system.

Do this in three directions: from the main system to a subsidiary system, from a subsidiary system to the main system, and from a subsidiary system to another subsidiary system.

c. If you have a relay host, send mail to another domain from the mail host to ensure that the relay mailer and host are configured properly.

d. If you have set up a uucp connection on your phone line to another host, send mail to someone at that host, and have that person send mail back or call you when he or she receives the message.

e. Ask someone to send mail to you over the uucp connection.

The `sendmail` program cannot tell whether the message gets through, because it hands the message to `uucp` for delivery.

f. Send a message to `postmaster` on different systems and make sure that it comes to your postmaster's mailbox.

Administering the Mail Configuration

This section describes how to keep the mail service running smoothly.

Duties of the Postmaster

Your responsibilities as postmaster for administering mail include the following tasks:

- Check the mail queues to be sure mail is flowing in and out.
- Check any downed systems where mail is backing up. If the system is not needed, delete it from the mail service, or bring the system up to keep mail moving.
- Fix personal aliases, as requested.
- Administer alias databases as people move in and out of the domain.
- Set up temporary forwarding files.
- Contact owners of mailing lists and help them fix mailing list problems.
- Go through `postmaster` mail daily and look for problems, like broken `.forward` files and mail alias loops.
- Answer questions outside the company.
- Truncate log files periodically.

Mail Queue

Under high-load or temporary-failure conditions, `sendmail` puts a message into a job queue instead of delivering it immediately. The mail queue usually is processed automatically, but sometimes you may have to intervene. For example, if a major host is down for a period of time, the queue may become clogged. Although `sendmail` ought to recover gracefully when the host comes up, you may find performance unacceptably bad in the meantime.

Format of Queue Files

The `sendmail` program stores temporary queue files in the queue file `/var/spool/mqueue`. All such queue files have the form `xAA99999`, where `AA9999` is the ID for the file and `x` is the type. Table 9-2 shows the types of queue files.

Table 9-2 Types of Queue Files

Type	Description
d	Data file. The message body (excluding the header) is kept in this file.
l	Lock file. If this file is present, the job is currently being processed and running the job in the queue will not process it. For this reason, an extraneous lock file can cause a job to seem to disappear.
n	A file created when an ID is created. It is a separate file that ensures no mail can ever be destroyed because of a race condition. This file should not exist for more than a few milliseconds at any given time.
q	Queue control file. This file contains the information needed to process the job.
t	A temporary file. This file is an image of the <code>qf</code> file when it is being rebuilt. When the rebuild is complete, the file is renamed <code>qf</code> .
x	Transcript file that shows everything that happens during that session.

The `qf` file contains a series of lines each beginning with a code letter, as shown in Table 9-3.

Table 9-3 Codes for the `qf` File

Code	Description
P	The current message priority, which is used to order the queue. The higher the number, the lower the priority. The priority increases as the message sits in the queue. The initial priority depends on the message class and the size of the message.
T	The job creation or submission time in seconds, which is used to compute when the job times out.
D	The name of the data file.
M	A message line, which is printed by using <code>sendmail</code> with the <code>-bp</code> flag, and is generally used to store status information. It can contain any text.
S	The sender name.

Table 9-3 Codes for the `qf` File (Continued)

Code	Description
E	The error recipient name. Error messages are sent to this user instead of the sender. This line is optional.
H	A header definition. There may be any number of these lines. The order is important: it represents the order in the final message. The syntax is the same as header definitions in the configuration file.
R	A recipient name. There will be one line for each recipient. The recipient name will normally have complete aliases, but its aliases are redone when the job is processed. The recipient name must be at the end of the <code>qf</code> file.

See Appendix C, “Customizing sendmail Configuration Files,” for more information.

The queue is automatically run at the interval specified in the `sendmail.cf` file (the default is every hour). The queue is read and sorted, and then `sendmail` tries to process all jobs in order. The `sendmail` program first checks to see if a job is locked. If the job is locked, `sendmail` ignores the job; if the job is not locked, `sendmail` processes it.

If a major host goes down for several days, the queue may become prohibitively large, and `sendmail` will spend time sorting the queue. You can fix this by moving the queue to a temporary place and creating a new queue. You can run the old queue later when the host is returned to service.

▼ How to Print the Queue

You can print the contents of the queue by specifying the `-bp` flag to `sendmail`:

- ◆ **Type** `/usr/lib/sendmail -bp | more` **and press Return.**
A list of the queue IDs, the size of the message, the date the message entered the queue, message status, and the sender and recipients are displayed.

▼ How to Force the Queue

1. **Become root on the mail host.**

2. **Type** `ps -el | grep sendmail` **and press Return.**
Note the process identification (PID) for `sendmail`. You will use it in the next step.
3. **Type** `kill pid` **and press Return.**
This kills the old `sendmail` daemon to keep it from trying to process the old queue directory.
4. **Type** `cd /var/spool` **and press Return.**
5. **Type** `mv mqueue omqueue; mkdir mqueue` **and press Return.**
This moves the directory, `mqueue`, and all its contents to the directory, `omqueue`, and then creates a new empty `mqueue` directory.
6. **Type** `chmod 755 mqueue; chown daemon mqueue; chgrp daemon mqueue` **and press Return.**
These commands set the permissions of the directory to read/write/execute by owner, and read/execute by group and others; these commands also set the owner and group to `daemon`.
7. **Type** `/usr/lib/sendmail -bd -q1h` **and press Return.**
This starts a new `sendmail` daemon, with a queue runtime of one hour.

▼ How to Run the Old Mail Queue

To run the old mail queue:

1. **Type** `/usr/lib/sendmail -oQ/var/spool/omqueue -q` **and press Return.**
The `-oQ` flag specifies an alternate queue directory and the `-q` flag says to run every job in the queue. Use the `-v` flag if you want to see the verbose output displayed on the screen.
2. **When the queue is finally emptied, type** `rmdir /var/spool/omqueue` **and press Return.**
This removes the empty directory.

To run a subset of the mail queue:

- ♦ **Type** `/usr/lib/sendmail -R string` **and press Return.**
You can run a subset of the queue at any time with the `-Rstring` (run queue where any recipient name matches *string*) or with `-Mnnnn` (run just one message with queue ID *nnnnn*).

Example of Processing a Subset of the Queue

This example processes everything in the queue for recipient, wnj.

```
# /usr/lib/sendmail -Rwnj
```

System Log

The mail services log most errors using the `syslogd` program. The default is for `syslogd` to send messages to the `loghost`.

You can define a system called `loghost` in the `/etc/hosts` file to hold all logs for an entire NIS domain. The system log is supported by the `syslogd` program. You specify a `loghost` in `/etc/hosts`. If no `loghost` is specified, then error messages from `syslogd` are not reported.

Here is the default `/etc/syslog.conf` file:

Code Example 9-1 The `syslog.conf` File

```
#ident "%Z%M% %I% %E% SMI" /* SunOS 5.x */
#
# Copyright (c) 1994 by Sun Microsystems, Inc.
#
# syslog configuration file.
#
# This file is processed by m4 so be careful to quote (') names
# that match m4 reserved words. Also, within ifdef's, arguments
# containing commas must be quoted.
#
# Note: Have to exclude user from most lines so that user.alert
# and user.emerg are not included, because old sendmails
# will generate them for debugging information. If you
# have no 4.2BSD based systems doing network logging, you
# can remove all the special cases for "user" logging.
#
*.err;kern.debug;auth.notice;user.none /dev/console
*.err;kern.debug;daemon,auth.notice;mail.crit;user.none
/var/adm/messages

*.alert;kern.err;daemon.err;user.none operator
```

Code Example 9-1 The `syslog.conf` File (Continued)

```
*.alert;user.none          root

*.emerg;user.none         *

# if a non-loghost machine chooses to have authentication
# messages
# sent to the loghost machine, un-comment out the following line:
#auth.notice              ifdef('LOGHOST', /var/log/authlog,
@loghost)

mail.debug                ifdef('LOGHOST', /var/log/syslog,
@loghost)

#
# non-loghost machines will use the following lines to cause
# "user"
# log messages to be logged locally.
#
ifdef('LOGHOST', ,
user.err                  /dev/console
user.err                  /var/adm/messages
user.alert                `root, operator'
user.emerg                *
)
```

You can change the default configuration by editing the `/etc/syslog.conf` file.

Each line in the system log contains a time stamp, the name of the system that generated it, and a message. The `syslog` file can log a large amount of information.

The log is arranged as a succession of levels. At the lowest level, only unusual occurrences are logged. At the highest level, even the most mundane and uninteresting events are recorded. As a convention, log levels under 10 are considered “useful.” Log levels higher than 10 are usually used for debugging. See *File System Administration* for information about `loghost` and the `syslogd` program.

Troubleshooting Tips

This section provides some tips and tools that you can use for troubleshooting problems with the mail services.

▼ How to Check Aliases

To verify aliases and whether mail can be delivered to a given recipient:

- ◆ **Type** `/usr/lib/sendmail -v -bv recipient` and **press Return**. The command displays the aliases and identifies the final address as deliverable or not.

Here is an example of the output:

```

% /usr/lib/sendmail -v -bv shamira@raks
shamira... aliased to   mwong
mwong... aliased to   shamira@raks
shamira@raks... deliverable
%
```

You should take extra care to avoid loops and inconsistent databases when both local and domain-wide aliases are used. Be especially careful when you move a user from one system to another to avoid creating alias loops.

▼ How to Test the `sendmail` Program

1. **Type** `/usr/lib/sendmail -bt` and **press Return**. Information is displayed.
2. **At the last prompt (>) type a 0 (zero) and the mail address you want to test.**
3. **Type Control-d to end the session .**

See Appendix C, “Customizing `sendmail` Configuration Files,” for a complete description of the diagnostic information.

▼ How to Verify Connections to Other Systems

To verify connections to other systems, you can use the `mconnect` program to open connections to other `sendmail` systems over the network. The `mconnect` program runs interactively. You can issue various diagnostic commands. See the reference manual page for `mconnect(1)` for a complete description.

If you cannot use `mconnect` to connect to an SMTP port, check these conditions:

- Is the system load too high?
- Is the `sendmail` daemon running?
- Does the system have the appropriate `/etc/mail/sendmail.cf` file?
- Is port 25 (the port that `sendmail` uses) active?

Other Diagnostic Information

For other diagnostic information, check the following sources:

- Look at the `received` lines in the header of the message. These lines trace the route the message took as it was relayed. Note that in the `uucp` network, many sites do not update these lines, and in the Internet, the lines often get rearranged. To straighten them out, look at the date and time in each line. Do not forget to account for time zone differences.
- Look at the messages from `MAILER-DAEMON`. These typically report delivery problems.
- Check the system log that records delivery problems for your group of systems. The `sendmail` program always records what it is doing in the system log. You may want to modify the `crontab` file to run a shell script nightly that searches the log for `SYSERR` messages and mails any that it finds to the postmaster.
- Use the `mailstats` program to test mail types and determine the number of messages coming in and going out.

Setting Up Printer Services Using the Command-Line Interface



This appendix describes how to set up local printers, print servers, print clients, and network printers with the LP print service defaults using the command-line interface to the LP print service.

If you want to skip the background information that explains the concepts of printer setup, and proceed directly to step-by-step instructions, use the following table to find the page where the instructions for a specific task begin.

<i>How to Set Up a SunOS 5.x Local Printer</i>	<i>page 281</i>
<i>How to Set Up a SunOS 4.1 Local Printer</i>	<i>page 288</i>
<i>How to Set Up a SunOS 5.x Print Server</i>	<i>page 288</i>
<i>How to Set Up a SunOS 4.1 Print Server</i>	<i>page 292</i>
<i>How to Set Up SunOS 5.x Print Clients</i>	<i>page 292</i>
<i>How to Set Up SunOS 4.1 Print Clients</i>	<i>page 295</i>
<i>How to Add a Network Printer (Using the BSD Protocol)</i>	<i>page 296</i>
<i>How to Add a Network Printer (Using a Special Protocol)</i>	<i>page 298</i>

If you want background information related to the instructions given in this appendix, read the section, “About Setting Up Printing Using the LP Print Service Command-Line Interface” on page 276.

You can refer to the following chapters and appendixes in this manual for additional information about printing:

- See Chapter 3, “Setting Up Printers,” for information about setting up printing using Printer Manager, an Administration Tool application.
- See Chapter 4, “Routine Printer Administration,” for information about how to administer printing services once the printers have been set up.
- See Chapter 5, “Managing Character Sets, Filters, Forms, and Fonts,” for tasks specific to different types of printers.
- See Chapter 6, “Setting Printing Policies,” for tasks to set printing policy, such as whether banner pages are required.
- See Chapter 7, “Troubleshooting Printing Problems,” if you have trouble setting up or administering printing.
- See Appendix B, “Understanding and Customizing the LP Print Service,” for a description of the LP print service and how to customize various aspects of it.

About Setting Up Printing Using the LP Print Service Command-Line Interface

Setting up printing services using the LP print service command-line interface is complicated and error-prone. The Printer Manager offers an easier and more reliable solution. Through printer manager, you just supply the information and let the tool run the required commands.

You should use the procedures described in this chapter only when you have special needs that require the command-line approach, like writing scripts to perform batch setup.

Regardless of the method, the decisions you make and information you need to supply during setup are similar, whether you use the Printer Manager or command lines. Consequently, most of the conceptual and reference information you may need is described in Chapter 3, “Setting Up Printers,” and it is not repeated in this appendix. Cross-references from the margin notes for the tasks point you to the information, as needed. Background information that applies only to using the command-line interface is presented before the instructions in this chapter.

Print Server Configuration

In addition to defining the characteristics of the printer to the LP print service, you must perform the following tasks for a SunOS 5.x system (with its printer) to act as a print server, thereby providing printing services over the network:

- Configure the port monitor.
- Register the network listen service.
- Identify the print clients.

Configuring the Port Monitor

For print clients to access a print server running SunOS 5.x system software, the port monitor on the print server must accept service requests and notify the LP print service of such requests. In addition, the port monitor on the SunOS 5.x print clients must be running to receive message from the server. See *Peripherals Administration* for a complete discussion of port monitors and the Service Access Facility.

Registering the Network Listen Service With the Port Monitor

The LP print service uses a connection-oriented protocol to establish connections for incoming requests from remote systems. When the port monitor is configured, the following listen services are registered:

- Service 0
- listenS5
- listenBSD

These services “listen” for print requests from print clients, or confirmations from the server. When a communication is detected, the service hands over the process to the `lpNet` daemon.

Identifying Print Clients to the Print Server

By default the print client type is BSD. If you want a SunOS 5.x print server to support a SunOS 4.1 print client, you must identify the client’s type as BSD, which is then stored along with the client’s name in the `/etc/lp/Systems` file on the server.

Setting Up a Network Printer

A *network printer* is a hardware device that provides printing services to print clients without being connected to a print server. It has its own system name and IP address, and it is connected directly to the network.

The procedures in “How to Add a Network Printer (Using the BSD Protocol)” on page 296 describe the steps needed to add a network printer to your network. Two possible variations are covered:

- The network printer uses the BSD protocol. The print client is set up as if it were printing to a print server that uses the BSD protocol.

Although the system administrator can choose the IP address and system name of the network printer, the administrator may not be able to change the printer name.

- The network printer uses a special protocol. The print client must use a vendor-supplied program to send print jobs to this printer.

The procedures to set up the vendor-supplied program can vary. The vendor might supply an SVR4 interface program for you to use. If so, this program will call the command to use to send the job to the printer. If not, you will need to modify the standard interface script to call the vendor-supplied program.

For the second case, you should set up a system as a print server. You can use `/dev/null` as the printer port, since the vendor-supplied program will actually perform the communication to the printer using the network interface. This is done by editing the per-printer copy of the standard interface to call the vendor-supplied program, and is described in the second procedure.

Other print clients can be set up to either print to the print server (acting as a server to the network printer) or directly to the network printer.

Select the procedure that identifies your network printer.

Printer Configuration Information for LP Print Service Commands

This section describes the configuration information you may need to understand when you are using the LP print service command-line interface, rather than the Printer Manager, to set up a printer on the network:

- The filter names for your printer
- The universal address version of the print server's Internet Protocol (IP) address required for printing between systems

SunOS 5.x system software provides a default set of PostScript filters, which the Printer Manager automatically installs. Some of the TranScript filters used with SunOS 4.1 have SunOS 5.x equivalents, and some do not. Table A-1 lists the default PostScript filters and identifies the TranScript filters, when applicable.

Table A-1 PostScript Filters

Filter	Action	TranScript Equivalent
download	Download fonts	
dpost	ditroff to PostScript	psdit
postdaisy	daisy to PostScript	
postdmd	dmd to PostScript	
postio	Serial interface for PostScript printer	pscomm
postior	Communicate with printer	
postmd	Matrix gray scales to PostScript	
postplot	plot to PostScript	psplot
postprint	simple to PostScript	enscript
postreverse	Reverse or select pages	psrev
posttek	TEK4014 to PostScript	ps4014

SunOS 5.x does not provide the following filters:

- TEX
- oscat (NeWSprint opost)
- Enscript

The `postreverse`, `postprint`, `postio`, and `dpost` filters are provided in place of `Enscript`.

The Printer Manager installs the default filters. If you have printing needs not met by these filters, see Appendix B, “Understanding and Customizing the LP Print Service,” for more information about filters, including how to write a new filter.

Universal Address for the Print Server

The *universal address* for the print server is required only if the printer is connected to a print server on a network. As part of configuring the network listen process to listen for print requests from other systems, the universal address—the Internet Protocol address (IP address) in hexadecimal form—of the print server must be registered with the LP print service. The Printer Manager does this for you when you add a printer or add a client.

You can obtain such information using the `lpsystem -A` command. The universal address has four parts, as shown in Figure A-1. The third part, the IP Address, consists of zeros. The zeros represent a special IP address that refers to the local host. The last part, RFU, means Reserved for Future Use, and could be used for other families of addresses (for example, Open Systems Interface) in the future.

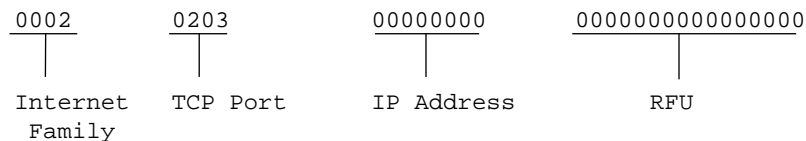


Figure A-1 Parts of the Universal Address

Setting Up Printing Services with the Command-Line Interface

This section contains instructions to set up:

- Local printers
- Print servers
- Print clients

The instructions in this section cover basic setup tasks, using the LP print service defaults with the command-line interface.

As an alternative, see Chapter 3, “Setting Up Printers,” for instructions on how to set up printing services using the Printer Manager, a graphic Administration Tool application.

▼ How to Set Up a SunOS 5.x Local Printer

Prerequisite

- System installed with SunOS 5.x system software

Information You Need

- System’s superuser password
- Device name (typically, `/dev/term/a`, `/dev/term/b`, or `/dev/bpp0`)
- Unique printer name
- Printer type
- Printer file content type
- Printer description

- 1. Connect the printer to the system and turn on the power to the printer.**
See the printer vendor’s manual and your system’s installation manual for information about switch settings and cabling requirements. You usually connect printer cables to a serial port, but in some cases you can use a parallel port.
- 2. On the system to which the printer is connected, log in as, or become, root.**
- 3. Type `chown lp /dev/address` and press Return.**
The `lp` user now owns the port device to which the printer is connected. For a serial port, `address` is usually `term/a` or `term/b`. For a parallel port, `address` is usually `bpp0`.
- 4. Type `chmod 600 /dev/address` and press Return.**
Now only `lp` can access the printer port device.

Note – You can combine steps 5, 6, and 7 by entering the `lpadmin` command once with the `-p`, `-T`, and `-I` options used together in the command.

- 5. Type `lpadmin -p printer-name -v /dev/address` and press Return.**
Give the printer a name unique on your network and associate that name with the port. The printer is registered with the LP print service.

6. Type `lpadmin -p printer-name -T printer-types` and press Return.

The printer type is used to extract information from the `terminfo` database. For a local PostScript printer, use uppercase `PS` for PostScript or uppercase `PSR` for PostScript Reverse (to reverse the order of the pages). You must specify printer type. If you do not, it defaults to `unknown`. You can specify more than one printer type; see `lpadmin(1M)`.

Note – If you specify printer type `PS` and pages print in reverse order, try printer type `PSR`. If you specify more than one printer type, you must specify `simple` as the content type.

7. Type `lpadmin -p printer-name -I file-content-types` and press Return.

Specify the file content types that the printer can print directly, which are added to the printer type (as specified with the `-T` option in step 6). Table A-2 lists some common file content types.

Table A-2 Common File Content Types for Local Printers

Types	Description
<code>cif</code>	Output of BSD <code>cifplot</code>
<code>daisy</code>	Daisy wheel printer
<code>dmd</code>	DMD
<code>fortran</code>	ASA carriage control format
<code>otroff</code>	Cat typesetter instructions generated by BSD or pre-System V <code>troff</code>
<code>plot</code>	Plotting instructions from Tektronix displays and devices
<code>postscript</code>	PostScript language
<code>raster</code>	Raster bitmap format for Varian raster devices
<code>simple</code>	ASCII file
<code>tex</code>	DVI format files
<code>troff</code>	Device-independent output from <code>troff</code>

For a local PostScript printer, you should use `postscript`, or `postscript, simple`, if in fact, the printer supports these types.

Note – PS and PSR are not file content types; they are printer types.

If you omit `-I`, the file content list defaults to `simple`. You can specify one or more file content types. Separate the names with commas. Or, you can use spaces to separate names, enclosing the word `allow:` and the list of users in quotation marks. If you use the `-I` option, `simple` is no longer the default, but must be included in the list to be valid. Use `any` as the file content type to bypass filtering. In this case, only file types that can be printed directly by the printer should be sent to it. Use `-I ""` to specify a null file content type list, which means an attempt should be made to filter all files, because the printer can directly print only files that exactly match its printer type.

8. Initialize the system's print filters.

You need to set up the print filters only once, when you install a printer (or if you reinstall the operating system). This step is not repeated for additional printers, unless they require different filters, which you must create (see “Creating a New Print Filter” on page 332). Print filters are used to convert the content type of a file to a content type acceptable to a printer. The system-supplied filters support only PostScript printers.

a. Type `cd /etc/lp/fd` and press Return.

Change to the directory containing the print filter descriptor files, so you do not have to give their full path to the `lpfilter` command.

b. Install the filters.

If you have a PostScript printer, you must install the filters for PostScript (`postscript`) content type; installing the filters for other content types is optional.

The following examples show several ways to install the filters using the `lpfilter -f filter-name -F filter-name .fd` command.

You can create a script that installs all the filters described in `/etc/lp/fd` by typing:

```
# sh
# for f in `ls | sed 's/.fd$//`
> do
> lpfilter -f $f -F $f.fd
> done
#
```

Or you could create a script that installs the required PostScript filters by typing:

```
# sh
# for f in download dpost postio postior postprint postreverse\
postdaisy postdmd posttek postplot postmd
> do
> lpfilter -f $f -F $f.fd
> done
#
```

Or you can type the command line for each filter:

```
# lpfilter -f download -F download.fd
# lpfilter -f dpost -F dpost.fd
# lpfilter -f postio -F postio.fd
# lpfilter -f postior -F postior.fd
# lpfilter -f postprint -F postprint.fd
# lpfilter -f postreverse -F postreverse.fd
# lpfilter -f postdaisy -F postdaisy.fd
# lpfilter -f postdmd -F postdmd.fd
# lpfilter -f posttek -F posttek.fd
# lpfilter -f postplot -F postplot.fd
# lpfilter -f postmd -F postmd.fd
```

The following table lists the filter names to use for other common file content types.

Table A-3 Print Filters for Common Non-PS File Content Types

File Content Type	Type Name	Filter Name
Plotter	plot	postplot
Daisy Wheel	daisy	postdaisy
ASCII	simple	postprint
TEK 4014	tek	posttek
DMD	dmd	postdmd
Matrix	md	postmd

The `lpfilter` command installs the filter description in the filter table, `/etc/lp/filter.table`. It uses the filter descriptor file (*filter-name*.fd from the `/etc/lp/fd` directory) to determine the characteristics of the filter and to find out where the filter program is located. PostScript filters are located in the `/usr/lib/lp/postscript` directory. You may locate other types of filters you add to the system in a different directory under `/usr/lib/lp`.

9. Type `accept printer-name` and press Return.

The printer is now ready to begin accepting (queuing) print requests.

10. Type `enable printer-name` and press Return.

The printer is now ready to process print requests in the print queue.

11. (Optional) Type `lpadmin -p printer-name -D "comment"` and press Return.

Attach a description giving the user helpful information, like where the printer is located. The *comment* is displayed as part of the printer status.

12. (Optional) Type `lpadmin -d printer-name` and press Return.

The printer you specify is established as the default printer for the system. Define a default printer even if there is only one printer configured for a system.

13. Type `lpstat -t` and press Return.

Check the messages displayed to verify that the printer is accepted and enabled.

Example of How to Set Up a PostScript Printer

This example is based on a network of five different systems. You have one PostScript printer on the network. You decide to designate `terra` as the print server, because it can support printing for all five systems.

You establish the following characteristics for this printer:

- Printer name: `luna`
- Printer type: PS
- File content types: `postscript,simple` (from Table A-2 on page 282)
- Device Name: `/dev/term/b` (determined by the port to which the printer is connected)

```
terra% su
# lpstat -r
scheduler is running
# chown lp /dev/term/b
# chmod 600 /dev/term/b
# lpadmin -p luna -v /dev/term/b
# lpadmin -p luna -T PS
# lpadmin -p luna -I postscript,simple
# cd /etc/lp/fd
# lpfilter -f download -F download.fd
# lpfilter -f dpost -F dpost.fd
# lpfilter -f postio -F postio.fd
# lpfilter -f postior -F postior.fd
# lpfilter -f postprint -F postprint.fd
# lpfilter -f postreverse -F postreverse.fd
# accept luna
destination "luna" now accepting requests
# enable luna
printer "luna" now enabled
# lpadmin -p luna -D "PostScript Laser printer in Building 5, Room 262"
# lpadmin -d luna
# lpstat -t
scheduler is running
system default destination: luna
device for luna: /dev/term/b
luna accepting requests since Mon Mar  4 14:37:55 PST 1994
printer luna is idle. enabled since Mon Mar  4 14:37:59 PST 1994. available.
# lp -d luna /etc/passwd
request id is luna-1 (1 file)
#
```

▼ How to Set Up a SunOS 4.1 Local Printer

Prerequisite

- System installed with SunOS 4.1 system software

Information You Need

- System's superuser password
 - Device name (typically, `/dev/term/a` or `/dev/term/b`)
 - Unique printer name
- ◆ **Set up the printer according to the instructions in the SunOS 4.1 system administration book, *System and Network Administration (800-3805-10)*.**
The interoperation of SunOS 4.1 and SunOS 5.x printing systems means that a printer on a SunOS 4.1 system can serve SunOS 5.x print clients.

▼ How to Set Up a SunOS 5.x Print Server

Prerequisites

- System running SunOS 5.x system software with a local printer connected to it (see "How to Set Up a SunOS 5.x Local Printer" on page 281)
- Network that enables communication between systems

Information You Need

- Server's superuser password
 - Names of printer, server, and clients
- 1. If not already done, set up the local printer.**
See "How to Set Up a SunOS 5.x Local Printer" on page 281 for instructions.
 - 2. Log in as, or become, root on the server system.**
 - 3. Configure the port monitor to accept service requests.**

Note – Perform step 3 *only once*, the first time you configure a print server system. Once registered, the processes and daemons are automatically restarted by the Service Access Controller when the system is rebooted.

- a. Type** `sacadm -a -p tcp -t listen \`
`-c "/usr/lib/saf/listen tcp" -v `nlsadmin -V` -n 9999`
and press Return.

The network listen service starts to listen for incoming TCP/IP requests. The options to the `sacadm` command are listed in Table A-4.

Table A-4 The `sacadm` Command Options

Option	Description
-a	Adds the -p port
-t	Identifies the type of service
-c	Tells which command to use to start the port monitor
-v	Identifies the version of the network listen process
-n	Specifies the number of times the Service Access Controller will restart the process, if it dies

- b. Type** `sacadm -l` **and press Return.**

When the network listen service is starting, the following information is displayed:

```
# sacadm -l
PMTAG      PMTYPE      FLGS RCNT STATUS      COMMAND
tcp        listen      -   9999 ENABLED    /usr/lib/saf/listen tcp #
```

Note – It may take several minutes before the network listen service is enabled.

4. Register the network listen service with the port monitor.

Note – Perform step 4 *only once*, the first time you configure a print server system. Once registered, the processes and daemons are automatically restarted by the Service Access Controller when the system is rebooted.

- a. Type `lpsystem -A` and press Return.
The system's universal address is displayed.

```
# lpsystem -A
00020203000000000000000000000000
```

You use the universal address, or a modified version of it, in step 4b. For the modified version, replace port number 0203 with 0ACE (the first digit is a zero). See the section “Universal Address for the Print Server” on page 280 for more.

Note – You must type the characters `\x` at the beginning of the universal (or modified universal) address in the next step exactly as shown. In addition, the address must be enclosed in single quotation marks so the backslash is not stripped off.

- b. Type `pmadm -a -p tcp -s type -i root -m `nlscmd` -v `nlsadmin -V`` and press Return.
This command configures the port monitor to listen for requests from the LP print service. Use the following table to determine which values to use for different service types as input to the command. Repeat this step for each of the three service types.

Table A-5 Variable Input to the `pmadm` Command Options

<i>type</i>	<i>nlscmd</i>
Value	Value
lp	<code>nlsadmin -o /var/spool/lp/fifos/listenS5</code>
lpd	<code>nlsadmin -o /var/spool/lp/fifos/listenBSD -A '\xaddress'</code>
0	<code>nlsadmin -c /usr/lib/saf/nlps_server -A '\xmodified_address'</code>

- c. Type `cat /var/saf/tcp/log` and press Return.
Examine the messages displayed to make sure that the services are enabled and initialized.

The following example illustrates step 4. The print server is `terra`. All three types of network listen services are registered.

```
# lpsystem -A
000202030000000000000000000000000000
# pmadm -a -p tcp -s lp -i root -m 'nlsadmin -o \
/var/spool/lp/fifos/listenS5' -v 'nlsadmin -V'
# pmadm -a -p tcp -s lpd -i root -m 'nlsadmin -o \
/var/spool/lp/fifos/listenBSD -A \
'\x000202030000000000000000000000000000' -v 'nlsadmin -V'
# pmadm -a -p tcp -s 0 -i root -m 'nlsadmin -c \
/usr/lib/saf/nlps_server -A \
'\x00020ACE0000000000000000000000000000' -v 'nlsadmin -V'
terra# cat /var/saf/tcp/log
10/28/91 10:22:51; 178; @(#)listen:listen.c      1.19.9.1
10/28/91 10:22:51; 178; Listener port monitor tag: tcp
10/28/91 10:22:51; 178; Starting state: ENABLED
10/28/91 10:22:51; 178; Service 0: fd 6 addr \
\x00020ACE0000000000000000000000000000
10/28/91 10:22:51; 178; Service lpd: fd 7 addr
\x000202030000000000000000000000000000
10/28/91 10:22:52; 178; Net opened, 2 addresses bound, 56 fds free
10/28/91 10:22:52; 178; Initialization Complete
#
```

5. (Optional) Perform additional setup steps, depending on the type of printer, and the printing policies you want to set for your site.

For example, you may want to create a class of printers and include the printer in that class. Or, you may want to set up the printer to use a certain form. See Chapter 5, “Managing Character Sets, Filters, Forms, and Fonts,” and Chapter 6, “Setting Printing Policies,” for more information.

The next task, after you have set up the print server, is usually to set up print clients. It is hard to tell if you set up the print server correctly until you set up a print client and try to use the printer on the print server. See “How to Set Up SunOS 5.x Print Clients” on page 292 or “How to Set Up SunOS 4.1 Print Clients” on page 295.

▼ How to Set Up a SunOS 4.1 Print Server

Prerequisites

- Working printer connected to a system running SunOS 4.1 system software
- Network that enables communication between systems

Information You Need

- Server's superuser password
- Names of printer, server, and clients

- ♦ **Set up the print server according to the instructions in the SunOS 4.1 system administration book *System and Network Administration* (800-3805-10).**

The interoperation of SunOS 4.1 and SunOS 5.x printing means that an existing 4.1 print server can be used by a SunOS 5.x print client by simply setting up the SunOS 5.x print client.

▼ How to Set Up SunOS 5.x Print Clients

Prerequisites

- System running SunOS 5.x or SunOS 4.1 system software that has been set up as a print server and that has a local printer connected
 - See "How to Set Up a SunOS 5.x Print Server" on page 288
 - See "How to Set Up a SunOS 4.1 Print Server" on page 292
- Systems running SunOS 5.x system software to become print clients
- Network that enables communication between systems

Information You Need

- Server's superuser password
- Clients' superuser password
- Names of printer, server, and clients
- Printer type
- Printer file content type

On each SunOS 5.x print client:

1. **Become root.**

Note – The next step is the key step that differs when the print client is using a SunOS 5.x print server versus a SunOS 4.1 print server.

2. Type `lpsystem -t type server-system-name` and press Return.

This step identifies the print server system and specifies its type as BSD (for SunOS 4.1) or System V (for SunOS 5.x). BSD is the default.

Note – You can combine steps 6, 7, and 8 by entering the `lpadmin` command once with the `-p`, `-s`, `-T`, and `-I` options used together in the command.

3. Type `lpadmin -p printer-name -s server-system-name` and press Return.

This step identifies the printer on the server system. If you want users on the print client to be able to refer to the printer by a name that is not the same as on the server, enter `-p printer-name1 -s server-system-name! printer-name2`.

4. (Optional) Type `lpadmin -p printer-name -T unknown` and press Return.

Specify a printer type of `unknown`. The actual printer type has to be specified only when the printer is set up as a local printer. If you omit this step, a printer type of `unknown` results by default.

5. (Optional) Type `lpadmin -p printer-name -I any` and press Return.

You should specify the file content type as `any`. This causes files to be filtered on the print server according to the file contents set up for the printer on the server.

6. Type `accept printer-name` and press Return.

The LP print system now accepts print requests.

7. Type `enable printer-name` and press Return.

The printer is enabled and can process print requests.

8. (Optional) Type `lpadmin -p printer-name -D "comment"` and press Return.

Attach a description giving the user helpful information, like where the printer is located. The *comment* is displayed as part of the printer status.

9. (Optional) Type `lpadmin -d printer-name` and press Return.

The printer you specify is configured as the default printer for the client system.

10. Type `lpstat -t` and press Return.

Check the messages that are displayed to verify that the printer is accepted and enabled.

11. Type `lp -d printer-name filename` and press Return.

The file you chose is sent to the printer. If the file did not print correctly, or did not print on the printer you expected, see Chapter 7, "Troubleshooting Printing Problems," for help.

Examples of Setting Up Print Clients

Following are two examples of how to set up a SunOS 5.x print client: one for a print client using a SunOS 5.x print server and one for a client using a SunOS 4.1 server.

Setting Up SunOS 5.x Print Clients for a SunOS 5.x Print Server

This example shows the steps for setting up the SunOS 5.x print client `mars` to print on `luna`, which is connected to the SunOS 5.x print server `terra`. Beginning with the `lpsystem` command, you would perform the same steps on other print clients like `venus`, `mercury`, and `jupiter` to let them become print clients of `terra`. You must tell each client system about the print server and the characteristics of its printer.

```
mars% su
# lpsystem -t s5 terra
"terra" has been added.
# lpadmin -p luna -s terra -T unknown -I any
# accept luna
destination "luna" now accepting requests
# enable luna
printer "luna" now enabled
# lpadmin -p luna -D "PostScript Laser printer in Building 5, Room 262"
# lpadmin -d luna
# lpstat -t
scheduler is running
system default destination: luna
system for luna: terra
luna accepting requests since Mon Mar  4 15:15:21 PST 1994
printer luna is idle. enabled since Mon Mar  4 15:15:26 PST 1994. available.
# lp /etc/passwd
request id is luna-23
```

Setting Up SunOS 5.x Print Clients for a SunOS 4.1 Print Server

The only differences when setting up a print client for a SunOS 4.1 print server instead of a SunOS 5.x print server are that you:

- Do not use the `lpssystem` command on the print server to identify the print clients
- Use `-t bsd` instead of `-t s5` in the `lpssystem` command

▼ **How to Set Up SunOS 4.1 Print Clients**

Refer to the SunOS 4.1 system administration book, *System and Network Administration* (800-3805-10), for details. Only a summary of the steps for setting up a SunOS 4.1 print client is given here.

Prerequisites

- System running SunOS 5.x or SunOS 4.1 system software that has been set up as a print server and that has a local printer connected. (See “How to Set Up a SunOS 5.x Print Server” on page 288, or “How to Set Up a SunOS 4.1 Print Server” on page 292.)
- Systems running SunOS 4.1 system software to become print clients
- Network that enables communication between systems

Information You Need

- Print server’s superuser password
- Print client’s superuser password
- Names of printer, server, and clients

On each SunOS 4.1 print client:

- 1. Log in as root on the client system.**
- 2. Add the remote printer to `/etc/printcap`.**
- 3. Create the spooling directory for that printer and set correct permissions/ownership.**
- 4. Send a file to the printer to verify that the setup works.**
If there are problems, see Chapter 7, “Troubleshooting Printing Problems,” for help.

Example of Setting Up a SunOS 4.1 Client for a SunOS 5.x Print Server

Suppose you have a SunOS 4.1 system named `neptune`. You want to let `neptune` print files on the printer `luna` on the SunOS 5.x system `terra`.

```
neptune% su
# vi /etc/printcap
luna|terra:\
    :lp=:rm=terra:rp=luna:br#9600:rw:\
    :lf=/var/spool/lpd/luna/log:\
    :sd=/var/spool/lpd/luna:
# cd /var/spool/lpd
# mkdir luna
# chmod 744 luna
# chgrp daemon luna
# lpr -Pluna /etc/passwd
```

Adding a Network Printer

Prerequisites

- System running SunOS 5.x system software
- Network communication between system and network printer

Information You Need

- Print client's superuser password
- Name of network printer
- IP address and system name of network printer

▼ How to Add a Network Printer (Using the BSD Protocol)

- 1. Connect the printer to the network and turn on the power to the printer.**
See the printer vendor's manual for information about configuring the IP address, setting hardware switches and cabling requirements.
- 2. Become root on the print client.**
- 3. Type `lpsystem -t bsd system-name` and press Return.**

4. Use the `lpadmin` command to add the printer and specify printer and content types. If the network printer comes with a preset printer name, use step a. Otherwise, use step b.
 - a. Type `lpadmin -p printer-name -s system-name -T unknown -I any` and press Return.
 - b. Type
`lpadmin -p local-printer-name -s system-name!remote-printer-name -T unknown -I any` and press Return.
If the network printer has a preset- printer name, specify a local printer name (*local-printer-name*) and the *system-name* with the network printer's hard coded *printer-name*. Be careful when using the exclamation point (!) because some shells treat this as a special character.
5. Type `accept printer-name` and press Return.
The LP print system now accepts print requests.
6. Type `enable printer-name` and press Return.
The printer is enabled and can process print requests.
7. (Optional) Type `lpadmin -p printer-name -D "comment"` and press Return.
Attach a description giving the user helpful information, like where the printer is located. The *comment* is displayed as part of the printer status.
8. (Optional) Type `lpadmin -d printer-name` and press Return.
The printer you specify is configured as the default printer for the client system.
9. Type `lpstat -t` and press Return.
Check the messages that are displayed to verify that the printer is accepted and enabled.
10. Type `lp -d printer-name filename` and press Return.
The file you chose is sent to the printer. If the file did not print correctly, or did not print on the printer you expected, see Chapter 7, "Troubleshooting Printing Problems," for help.

▼ How to Add a Network Printer (Using a Special Protocol)

1. **Connect the printer to the network and turn on the power to the printer.** See the printer vendor's manual for information about the IP address, hardware switches, and cabling requirements.
2. **On the system that will act as a print server, log in as, or become, root.**
3. **Use the `lpadmin` command to configure the printer. If you are using a vendor-supplied interface script, use step a to specify the name of the script with the `-i` option. If you are modifying the standard interface script, follow steps b-d.**

a. **Type `lpadmin -p printer-name -v /dev/null -T unknown -I any -i vendor-supplied interface` and press Return.**

Provide the printer name, associate that name with the port, and identify the printer type and file content type.

b. **Type `lpadmin -p printer-name -v /dev/null -T unknown -I any` and press Return.**

Provide the printer name, associate that name with the port, and identify the printer type and file content type.

c. **Modify the standard printer interface script to include the name of the vendor-supplied printing program. Three lines in this file are assigned to the `FILTER` variable:**

```
FILTER="/usr/lib/lp/postscript/postio"
FILTER="/usr/lib/lp/postscript/postreverse | \
/usr/lib/lp/postscript/postio"
FILTER="${LPCAT} 0"
```

d. **Edit the printer interface script, `/etc/lp/interface/printer-name`, to change these lines so that they specify the vendor-supplied program instead of the standard programs.**

The vendor-supplied program can be placed in the `/etc/lp/interfaces` directory, and it can be called from the interface script using that path name.

The standard interface will use the value of the `FILTER` variable as the program to run to send print jobs to the device. This variable is used further down in the standard interface program:

```
0<${file} eval ${FILTER} 2>&1 1>&3
```

-
4. **Type** `accept printer-name` **and press Return.**
The printer is now ready to begin accepting (queuing) print requests.
 5. **Type** `enable printer-name` **and press Return.**
The printer is now ready to process print requests in the print queue.
 6. **(Optional) Type** `lpadmin -p printer-name -D "comment"` **and press Return.**
Attach a description giving the user helpful information, like where the printer is located. The *comment* is displayed as part of the printer status.
 7. **(Optional) Type** `lpadmin -d printer-name` **and press Return.**
The printer you specify is established as the default printer for the system. Define a default printer even if there is only one printer configured for a system.
 8. **Type** `lpstat -t` **and press Return.**
Check the messages displayed to verify that the printer is accepting print jobs and enabled.

≡ A

Understanding and Customizing the LP Print Service



This appendix provides a detailed description of the LP print service: what it is, how it works, and how to customize various aspects of it.

If you want to skip the background information that explains the concepts of customizing the LP print service, and proceed directly to step-by-step instructions, use the following table to find the page where the instructions for a specific task begin.

<i>How to Adjust the Printer Port Characteristics</i>	<i>page 344</i>
<i>How to Adjust the terminfo Database for an Unsupported Printer</i>	<i>page 345</i>
<i>How to Provide a Custom Printer Interface Program</i>	<i>page 348</i>
<i>How to Provide a New Print Filter</i>	<i>page 349</i>
<i>How to Provide a New Form</i>	<i>page 353</i>

If you want background information about the LP print service and how it works, read the sections “An Overview of the LP Print Service” on page 302 and “Understanding How the Print Service Works” on page 319.

If you want background information related to the instructions given in this appendix, read the section, “About Customizing the LP Print Service” on page 327.

An Overview of the LP Print Service

The *LP print service* is a set of software utilities that allows users to print files while they continue to work. Originally, it was called the LP spooler. (Spool is an acronym for system peripheral operation off-line. LP stood for line printer, but its meaning now includes many other types of printers, like laser printers.)

The print service consists of the LP print service software, any print filters (programs that process data before printing) you may provide, and the hardware (the printer, workstation, and network connections).

The following section describes the functions provided by the LP print service. Then, the section “The Structure of the LP Print Service” on page 308 describes the directory structure and commands.

Functions Provided by the LP Print Service

The LP print service performs the following functions:

- Administers files and schedules local print requests
- Schedules network requests
- Filters files (if necessary) so that they print properly
- Starts programs that interface with the printers
- Tracks the status of jobs
- Tracks forms mounted on the printer
- Tracks print wheels currently mounted
- Delivers alerts to mount new forms or different print wheels
- Delivers alerts about printing problems

Administering Files and Scheduling Local Print Requests

The LP print service has a scheduler daemon called `lp sched`. The scheduler daemon updates the LP system files with information about printer setup and configuration.

The `lp sched` daemon also schedules all local print requests, as shown in Figure B-1, regardless of whether the requests are issued by users from an application or from the command line. In addition, the scheduler tracks the status of printers and filters. When a printer finishes printing a request, the scheduler schedules the next request, if there is one in the queue.

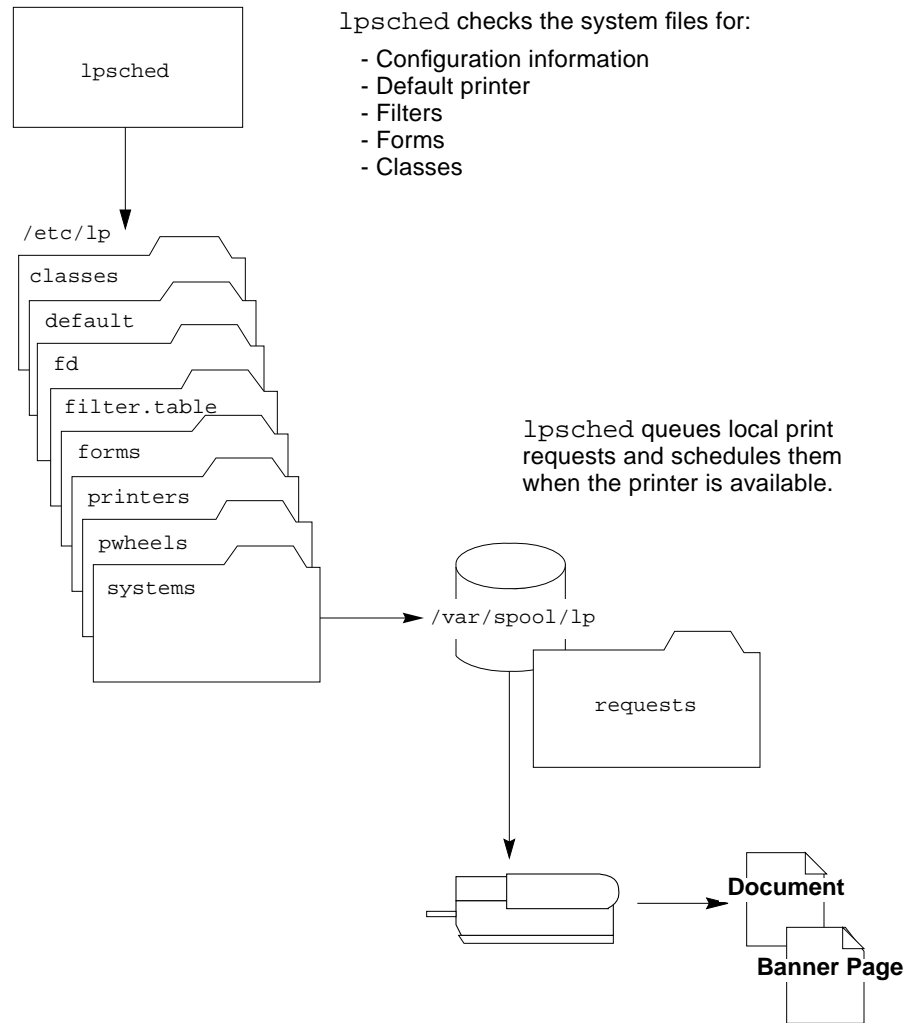


Figure B-1 The lpsched Schedules Local Print Requests

Each print client and print server must have *only* one LP scheduler running. The scheduler is started when a system is booted (or enters run level 2) by the control script `/etc/rc2.d/S80lp`. Without rebooting the systems, you can

stop the scheduler with the `/usr/lib/lp/lpshut` command and restart the scheduler with the `lpsched` command. The scheduler for each system manages requests issued to the system by the `lp` commands.

Scheduling Network Print Requests

Each print client and print server must have one or more `lpNet` daemons. The `lpNet` daemon schedules network print requests. The `lpNet` daemon is started when a system is booted. If you stop and restart the scheduler (using the `lpshut` and `lpsched` commands), the `lpNet` daemon is also stopped and restarted.

The `lpNet` daemon needs a configured port monitor and registered listen services to handle incoming network requests on each print server running SunOS 5.x system software.

Filtering Print Files

Print filters are programs on the print server that convert the content of a to-be-printed file from one format to another. The LP print service uses filters to:

- Convert a file from one data format to another so that it can be printed properly on a specific type of printer
- Handle the special modes of printing that users may request using the `-y` option to the `lp` command, for example, two-sided printing, landscape printing, draft- or letter-quality printing
- Detect printer faults and notify the LP print service of them so that the print service can deliver alerts

Not every print filter can perform all these tasks. However, because each task is printer-specific, the task can be implemented separately.

A print filter can be as simple or as complex as needed. SunOS 5.x system software provides print filters in the `/usr/lib/lp/postscript` directory that cover most PostScript printing situations—where the destination printer requires the data to be in PostScript format. You have to create and add filters to the system for non-PostScript printers.

A set of *print filter descriptor files* are provided in the `/etc/lp/fd` directory. These descriptor files describe the characteristics of the filter (for example, fast or slow filter), and point to the filter program (for example, `/usr/lib/lp/postscript/postdaisy`).

Starting the Printer Interface Program

The LP print service interacts with other parts of the operating system. It uses a standard printer interface program to:

- Initialize the printer port, if necessary. The standard printer interface program uses the `stty` command to initialize the printer port.
- Initialize the printer. The standard printer interface program uses the `terminfo` database and the `TERM` shell variable to find the appropriate control sequences.
- Print a banner page, if necessary.
- Print the correct number of copies specified by the print request.

The LP print service uses the standard interface program (found in the `/usr/lib/lp/model` directory) unless you specify a different one. You can create custom interface programs, but you must be careful that the custom program does not terminate the connection to the printer or interfere with proper printer initialization.

Tracking the Status of Print Jobs

The `lpsched` daemon on both the print server and print client keeps a log of each print request that it processes and notes any errors that occurred during the printing process. This log is kept in the `/var/lp/logs/lpsched` file. Every night, the `lp` cron job renames `/var/lp/logs/lpsched` to a new file `lpsched.n` and starts a new log file. If errors occur or jobs disappear from the print queue, you can use the log files to determine what `lpsched` has done with a printing job.

Tracking Forms

The LP print service helps you (the system administrator) track which forms are mounted on each printer, and notifies you when it cannot find the description of how to print on the form. You are responsible for creating form

descriptions and mounting and unmounting the paper form in each printer, either as part of setting up a printer or in response to alerts from the LP print service.

Users can specify the form on which they want a job to print. You (root) can mount a specific form and then tell the LP print service that the form is available and on which printer it is mounted. Or users can submit print requests specifying a particular form, and whether or not the form is mounted. When the LP print service receives the request, it sends an alert message to the system administrator (root) requesting that the form be mounted.

Tracking Print Wheels

The procedure for tracking print wheels is similar to the procedure for tracking forms. Some printers (usually letter-quality printers) have removable print heads, like daisy wheels or print balls, that provide a particular font or character set. A user can request a named character set. If that character set is not available, the LP print service notifies the system administrator (root) of the request. The job is stored in the print queue until the print wheel is changed.

Receiving Printing Problem Alerts

The LP print service performs sophisticated error checking. If a printing problem occurs, alerts are sent to the originator of a print request, or to the system administrator, depending on the nature of the problem and what is required to fix it. Users are notified when a print request cannot be completed. If users request it, they are notified by email when a job is successfully completed. Administrators are alerted to problems with printers, and to requests for filters, forms, or character sets.

For problems that require an administrator's attention, the LP print service default is to write an alert message to the system administrator's console window (that is, to the terminal on which root is logged in).

As the system administrator, you can change the alert policy to receive alert messages via:

- Electronic mail
- Program of your choice

Or, you can choose to receive no alerts when printing problems occur.

The Structure of the LP Print Service

This section explains the structure and directory hierarchy for the LP print service. The many files of the LP print service are distributed among seven directories, as shown in Table B-1.

Table B-1 Directories for the LP Print Service

Directory	Contents
/usr/bin	The <code>lp</code> , <code>lpstat</code> , <code>enable</code> , and <code>disable</code> commands
/etc/lp	A hierarchy of LP configuration files
/usr/share/lib	The <code>terminfo</code> database directory
/usr/sbin	The LP commands
/usr/lib/lp	The LP daemons; directories for binary files and PostScript filters; and the <code>model</code> directory (which contains the standard printer interface program)
/var/lp/logs	The logs for LP activities
lpNet	Messages from lpNet
lpsched.n	Messages from lpsched
requests.n	Information about completed print requests
/var/spool/lp	The spooling directory where files are queued for printing

User Commands

The `/usr/bin` directory contains the `lp` and `lpstat` commands, with which users submit and monitor print requests. The directory also contains the `enable` and `disable` commands, with which printers are enabled and disabled.

A user can customize a print request by using options to the `lp` command: specifying forms, character sets, filters, titles, banners, and so forth. Table B-2 summarizes the frequently used options for the `lp` command. You can use these options individually or combine them in any order on the command line. When combining options, use a space between each option and repeat the dash (-). For example, the following command specifies a destination printer, requests email notification, and prints six copies of a file.

```
% lp -d printer-name -m -n6 filename
```

Table B-2 Summary of Frequently Used `lp` Command Options

Option	Description
<code>-d</code>	Destination. Specifies a destination printer by name.
<code>-m</code>	Mail. Sends email to the user that submitted the print request when the file has printed successfully.
<code>-n</code>	Number. Specifies the number of copies to be printed.
<code>-t</code>	Title. Specifies a title for a print request (printed only on the banner page).
<code>-o nobanner</code>	Option. Suppresses printing of the banner page for an individual request.
<code>-c</code>	Copy. Copies the file before printing.
<code>-w</code>	Write. Writes a message to root's terminal when the file has printed successfully.

See `lp(1)` for a complete list of options.

LP Configuration Files

The scheduler stores configuration information in LP configuration files located in the `/etc/lp` directory, as described in Table B-3. These configuration files serve the function of the `/etc/printcap` file in SunOS 4.1.

Note – You can check the contents of these files, but you should not edit them directly. Instead, use the LP administrative commands, described in “LP Administrative Commands” on page 313, provide input for the configuration files in the `/etc/lp` directory. The `lpsched` daemon administers and updates the configuration files. Instead, use the administrative commands any time you need to update any configuration file.

The following table describes the contents of the `/etc/lp` directory.

Table B-3 Contents of the `/etc/lp` Directory

File	Type	Description
<code>Systems</code>	ASCII file	Names of systems defined using the <code>lpssystem</code> command. Includes every remote system with which the local system can exchange print requests.
<code>classes</code>	Directory	Files identifying classes provided by the <code>lpadmin -c</code> command.
<code>default</code>	ASCII file	Name of the default destination provided by the <code>lpadmin -d</code> command.
<code>fd</code>	Directory	Description of existing filters.
<code>filter.table</code>	File	Print filter lookup table.
<code>forms</code>	Directory	Location to put files for each form. Initially, this directory is empty.
<code>interfaces</code>	Directory	Printer interface program files.
<code>logs</code>	Link to <code>/var/lp/logs</code>	Log files of printing activities.
<code>model</code>	Link to <code>/usr/lib/lp/model</code>	The standard printer interface program.
<code>printers</code>	Directory	Directories for each (remote or local) printer. Each directory contains configuration information and alert files for an individual printer.
<code>pwheels</code>	Directory	Print wheel or cartridge files.

The `printers` directory has a subdirectory for each printer (local or remote) known to the system. The following example shows the `printers` subdirectories of `printers sparcl` and `luna`.

```
%ls -l /etc/lp/printers
drwxrwxr-x 2 lp lp 512 Jan 23 23:53 luna
drwxrwxr-x 2 lp lp 512 Jan 11 17:50 sparcl
```

Within each of the printer-specific directories, the following files can describe the printer:

- `alert.sh` - Shell to execute in response to alerts
- `alert.vars` - Alert variables
- `configuration` - Configuration file
- `users.deny` - List of users to whom printer access is denied
- `comment` - Printer description

The configuration file for the printer `luna`, `/etc/lp/printers/luna/configuration`, would typically appear as follows:

```
Banner: on: Always
Content types: PS
Device: /dev/term/b
Interface: /usr/lib/lp/model/standard
Printer type: PS
Modules: default
```

Printer Definitions

The `/usr/share/lib` directory contains the `terminfo` database directory, which contains definitions for many types of terminals and printers. The LP print service uses information in the `terminfo` database to initialize a printer, to establish a selected page size, character pitch, line pitch, and character set, as well as to communicate the sequence of codes to a printer.

Each printer is identified in the `terminfo` database with a short name. See “Directory Structure of the `terminfo` Database” on page 84 for a description of the structure of the `terminfo` database. If necessary, you can add entries to the `terminfo` database, but it is a tedious and time-consuming process. See “Adjusting the `terminfo` Database for an Unsupported Printer” on page 327.

Daemons and LP Internal Files

The `/usr/lib/lp` directory contains daemons and files used by the LP print service, as described in Table B-4.

Table B-4 Contents of the `/usr/lib/lp` Directory

File	Type	Description
<code>bin</code>	Directory	Contains files for generating printing alerts, slow filters, and queue management programs.
<code>lpNet</code>	Daemon	Controls LP requests for network printing.
<code>lpdata</code>	ELF executable file	Lists LP print service configuration information (interactive command).
<code>lpsched</code>	Daemon	Manages scheduling of LP print requests.
<code>model</code>	Directory	Contains the standard printer interface program.
<code>postscript</code>	Directory	Contains all PostScript filter programs provided by the SunOS 5.x LP print service. These filters come with descriptor files in the <code>/etc/lp/fd</code> directory that tell the LP print service the characteristics of the filters and where to locate them.

LP Administrative Commands

The commands used to set up and administer the LP print service are in the `/usr/sbin` directory, as shown in Table B-5.

Table B-5 The `lp` Commands in the `/usr/sbin` Directory

Command	Purpose
<code>accept</code>	Accepts print requests into the printer's queue.
<code>lpadmin</code>	Defines printer names, printer types, file content types, print classes, printer devices, and printer comments; removes printers or print classes; specifies fault recovery, interface programs (either custom or standard), printing options, banner/no banner; mounts forms; mounts print wheels or cartridges; defines <code>allow</code> and <code>deny</code> user lists.
<code>lpfilter</code>	Adds, changes, deletes, and lists filters.
<code>lpforms</code>	Adds, changes, deletes, and lists forms.
<code>lpmove</code>	Moves queued print requests from one printer to another.
<code>lpshut</code>	Halts the LP print service. (<code>lp sched</code> , which starts the LP print service, is in the <code>/usr/lib/lp</code> directory.)
<code>lpssystem</code>	Registers print servers and print clients with the LP print service.
<code>lpusers</code>	Sets queue priorities for users.
<code>reject</code>	Rejects print requests into the printer's queue

Log Files

The LP print service maintains two sets of log files:

- A list of current requests that are in the print queue (`/var/spool/lp`)
- An ongoing history of print requests (`/var/lp/logs/requests`)

Print Queue Logs

The scheduler for each system keeps a log of print requests in the directories `/var/spool/lp/tmp/system` and `/var/spool/lp/requests/system`. Each print request has two files (one in each directory) that contain information about the request. The information in the `/var/spool/lp/requests/system`

directory can be accessed only by root or lp. The information in the `/var/spool/lp/tmp/system` can be accessed only by the user who submitted the request, root, or lp.

The following example shows the contents of the `/var/spool/lp/tmp/terra` directory:

```
terra% ls /var/spool/lp/tmp/terra
20-0 21-0
terra% cat 21-0
C 1
D slw2
F /etc/default/login
P 20
t simple
U tamiro
s 0x1000
```

These files remain in their directories only as long as the print request is in the queue. Once the request is finished, the information in the files is combined and appended to the file `/var/lp/logs/requests`, which is described in the next section.

Use the information in the `/var/spool/lp` logs if you need to track the status of a print request that is currently in the queue.

History Logs

The LP print service records a history of printing services in three log files: `lpNet`, `lpsched`, and `requests`. These log files are located in the `/var/lp/logs` directory. You can use the information in these logs to diagnose and troubleshoot printing problems. Here is an example of the contents of the `/var/lp/logs` directory:

```
# cd /var/lp/logs
# ls
lpNet      lpsched.1   requests    requests.2
lpsched    lpsched.2   requests.1
#
```

The files with the `.1` and `.2` suffixes are copies of the previous day's logs. Each day, the `lp` cron job cleans out the `lpsched` and `requests` log files and keeps copies for two days. See "How to Monitor and Clean Out Log Files" on page 122 for suggestions on modifying the cron job for cleaning out the `requests` log.

The two most important log files for troubleshooting are:

- The `lpNet` log, which contains information about network printing
- The `lpsched` log, which contains information about local printing requests

The `requests` log contains information about print requests that are completed and no longer in the print queue. Once a request is finished printing, the information in the `/var/spool/lp` log files is combined and appended to the `/var/lp/logs/requests` log.

The `requests` log has a simple structure, so that you can extract data using common UNIX shell commands. Requests are listed in the order they are printed, and are separated by lines showing their request IDs. Each line below the separator line is marked with a single letter that identifies the kind of information contained in that line. Each letter is separated from the data by a single space.

Here is an example of the contents of a `requests` log:

```
# pwd
/var/lp/logs
# tail requests.2
= slw2-20, uid 200, gid 200, size 5123, Mon Nov 18 01:24:01 EST 1994
z slw2
C 1
D slw2
F /etc/motd
P 20
t simple
U irving
s 0x0100
#
```

Table B-6 shows the letter codes in the LP `requests` log.

Table B-6 Letter Codes in the LP `requests` Log

Letter	Content of Line
=	The separator line. It contains the following items: request ID, the user ID (UID) and group IDs (GIDs) of the user, the total number of bytes in the original (unfiltered) file size, and the time when the request was queued.
C	The number of copies printed.
D	The printer or class destination or the word <code>any</code> .
F	The name of the file printed. The line is repeated for each file printed; files were printed in the order shown.
f	The name of the form used.
H	One of three types of special handling: <code>resume</code> , <code>hold</code> , and <code>immediate</code> . The only useful value found in this line will be <code>immediate</code> .
N	The type of alert used when the print request was successfully completed. The type is the letter <code>M</code> if the user was notified by email or <code>W</code> if the user was notified by a message to the terminal.
O	The printer-dependent <code>-o</code> options (for example, <code>nobanner</code>).
P	The priority of the print request.
p	The list of pages printed.
r	A single-letter line that is included if the user asked for “raw” processing of the files (the <code>-r</code> option of the <code>lp</code> command).
S	The character set, print wheel, or cartridge used.
s	The outcome of the request, shown as a combination of individual bits expressed in hexadecimal form. Several bits are used internally by the print service. The bits and what they mean are describe in Table B-7.
T	The title placed on the banner page.
t	The type of content found in the files.
U	The name of the user who submitted the print request.

Table B-6 Letter Codes in the LP requests Log (Continued)

Letter	Content of Line
x	The slow filter used for the print request.
Y	The list of special modes for the print filters used to print the request.
z	The printer used for the request. This printer differs from the destination (the D line) if the request was queued for any printer or a class of printers, or if the request was moved to another destination.

Table B-7 Outcome Codes in the LP Requests Log

Outcome Codes	Description
0x0001	The request was held pending resume.
0x0002	Slow filtering is running.
0x0004	Slow filtering finished successfully.
0x0008	The request is on the printer.
0x0010	Printing finished successfully.
0x0020	The request was held pending user change.
0x0040	The request was canceled.
0x0080	The request will print next.
0x0100	The request failed filtering or printing.
0x0200	The request is in transit to a remote printer.
0x0400	The user will notified.
0x0800	A notification is running.
0x1000	A remote system has accepted the request.
0x2000	The administrator placed a hold on the request.
0x4000	The printer had to change filters.
0x8000	The request is temporarily stopped.

Spooling Directories

Files queued for printing are stored in the `/var/spool/lp` directory until they are printed, which may be only seconds. Table B-8 shows the contents of the `/var/spool/lp` directory.

Table B-8 Contents of the `/var/spool/lp` Directory

File	Type	Description
SCHEDLOCK	File	Lock file for the scheduler. Check for this file if the scheduler dies and will not restart.
admins	Directory	Link to <code>/etc/lp</code> .
bin	Directory	Link to <code>/usr/lib/lp/bin</code> .
fifos	Directory	Pipes that convey networked print requests to and from the lpNet daemon.
logs	Link	Link to <code>../lp/logs</code> where completed print requests are logged.
model	Link	Link to <code>/usr/lib/lp/model</code> .
requests	Directory	Directory that contains subdirectories for each configured printer where print requests are logged until printed. Users cannot access this log.
system	Directory	A print status file for the system.
temp	Link	Link to <code>/var/spool/lp/tmp/printer-name</code> , which contains the spooled requests.
tmp	Directory	Directory for each configured printer where print requests are logged until printed. Changes to existing print requests are also recorded in this log.

Understanding How the Print Service Works

This section describes what happens to a local print request from the time it is submitted by the user until it is printed successfully. It also explains how printing works between SunOS 4.1 and SunOS 5.x systems.

Local Printing

Figure B-2 shows what happens when a user submits a request to print a PostScript file on a *local* printer, which is a printer connected to the user's system. All processing happens on the local system; no network printing software is used.

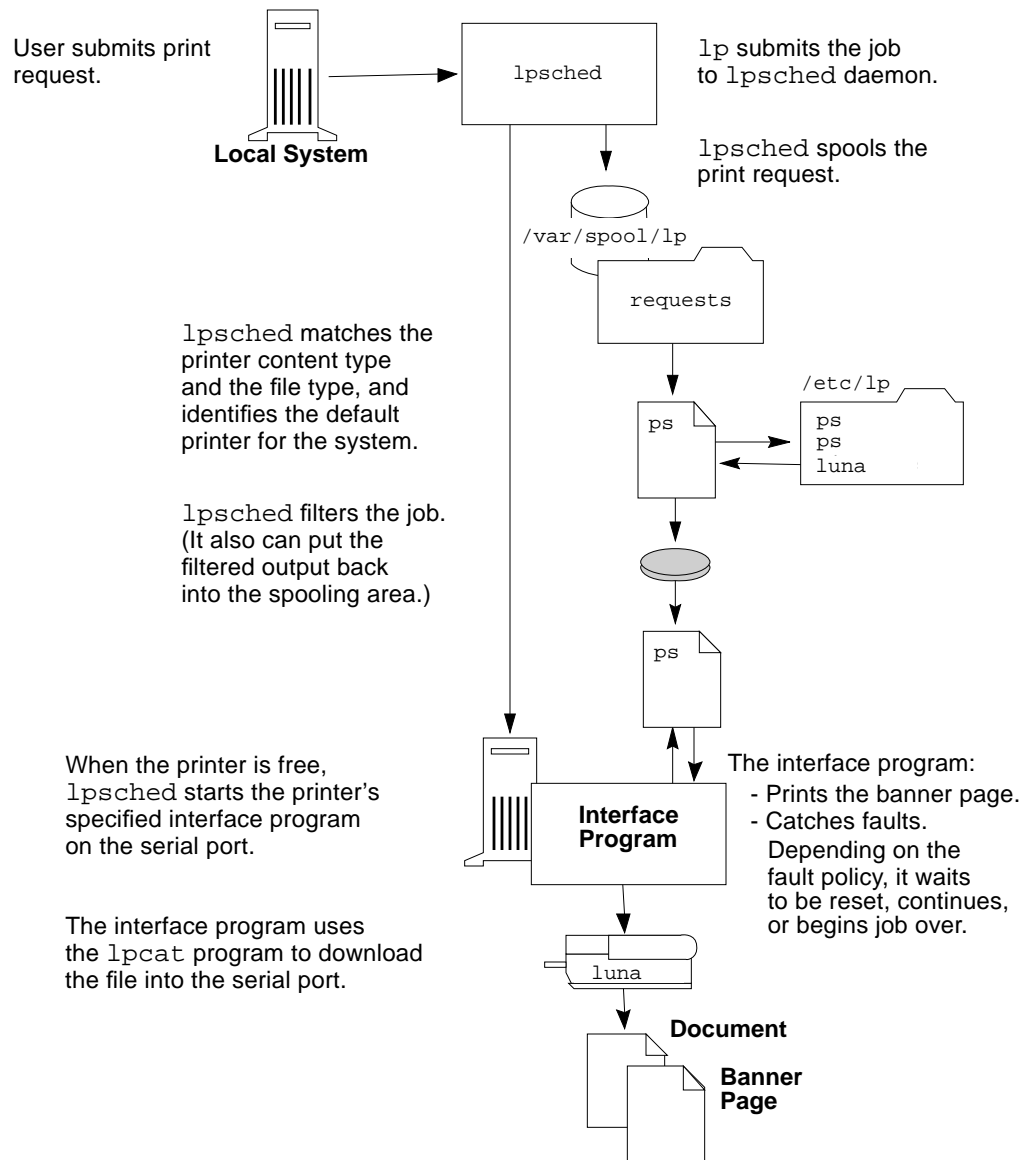


Figure B-2 The Local Printing Process

Remote Printing

Figure B-3 shows what happens when a user on a SunOS 5.x print client submits a print request to a SunOS 4.1 print server. The `lpsched` daemon handles the local part of the print request, and the `lpNet` daemon and its child process handle the network communication between the two systems.

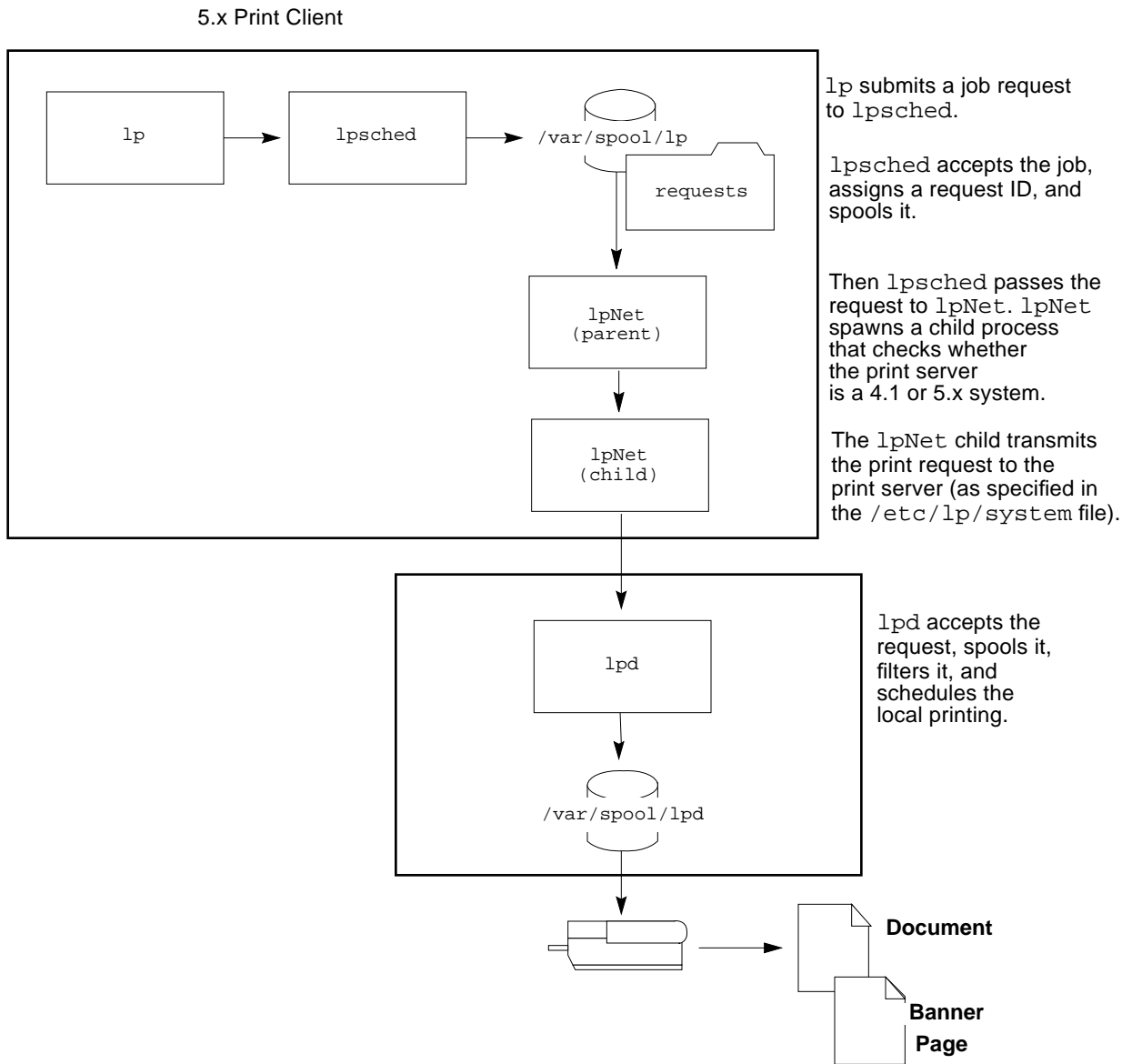


Figure B-3 Network Printing Between a SunOS 5.x Print Client and a SunOS 4.1 Print Server

Figure B-4 shows a SunOS 4.1 print client submitting a print request to a SunOS 5.x print server. The `lpd` daemon handles the local part of the print request and the connection to the print server. The Service Access Facility network listen process on the server waits for network printing requests and sends them to the `lpNet` daemon. The `lpNet` daemon and its child processes hand the request over to the `lpsched` daemon, which processes the request on the print server

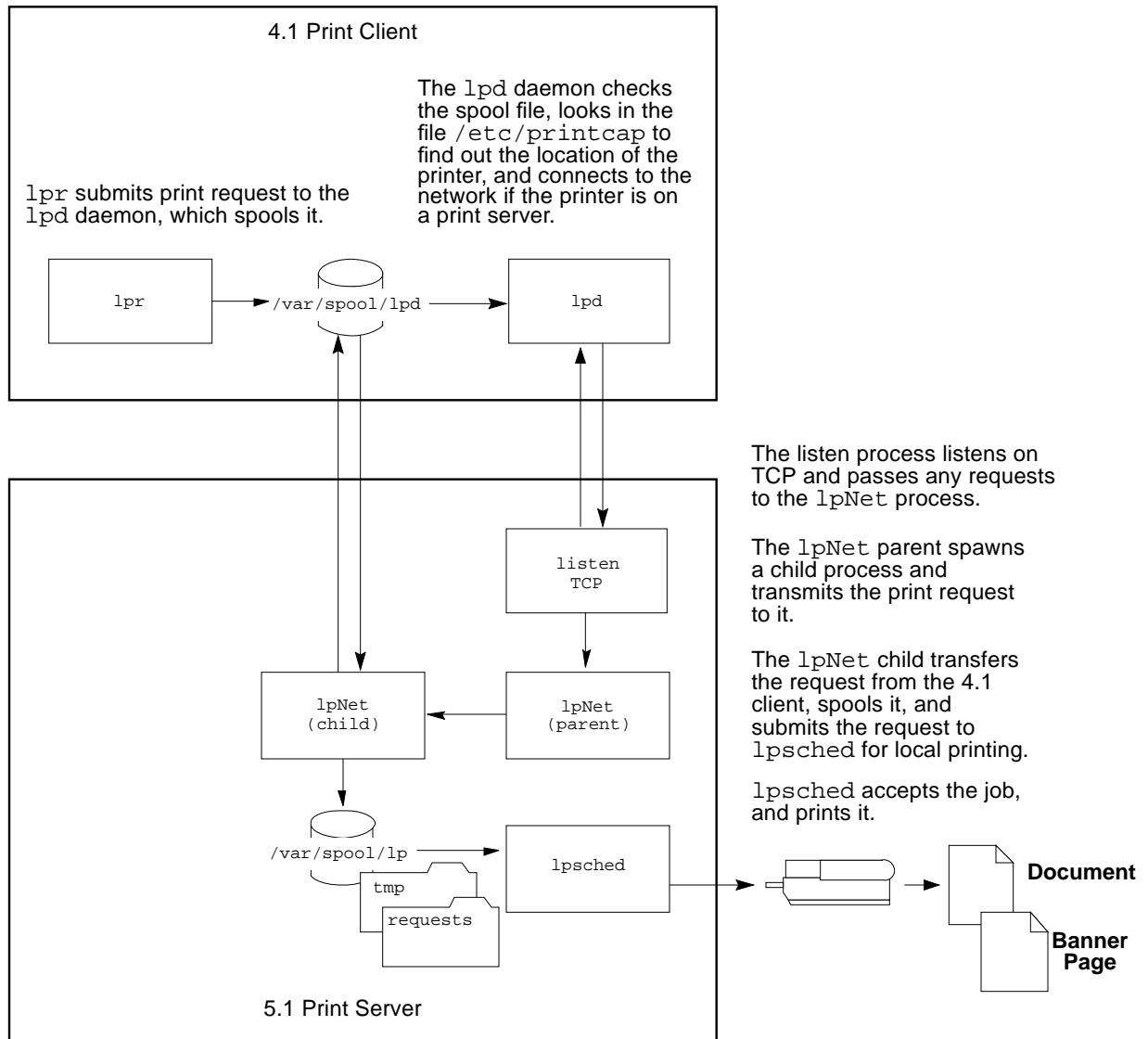


Figure B-4 Network Printing Between a SunOS 4.1 Print Client and a SunOS 5.x Print Server

Figure B-5 shows what happens when a user of a SunOS 5.x print client submits a print request to a SunOS 5.x print server. The `lpsched` daemon on the print client handles the local part of each print request. Then `lpsched` passes the request to the `lpNet` daemon on the print client, which spawns a child process that communicates with the print server. The Service Access Facility network listen service on the print server monitors network printing requests and sends them to the `lpNet` daemon on the print server. The `lpNet` daemon and its child processes send the request to the `lpsched` daemon on the print server, which processes the print request.

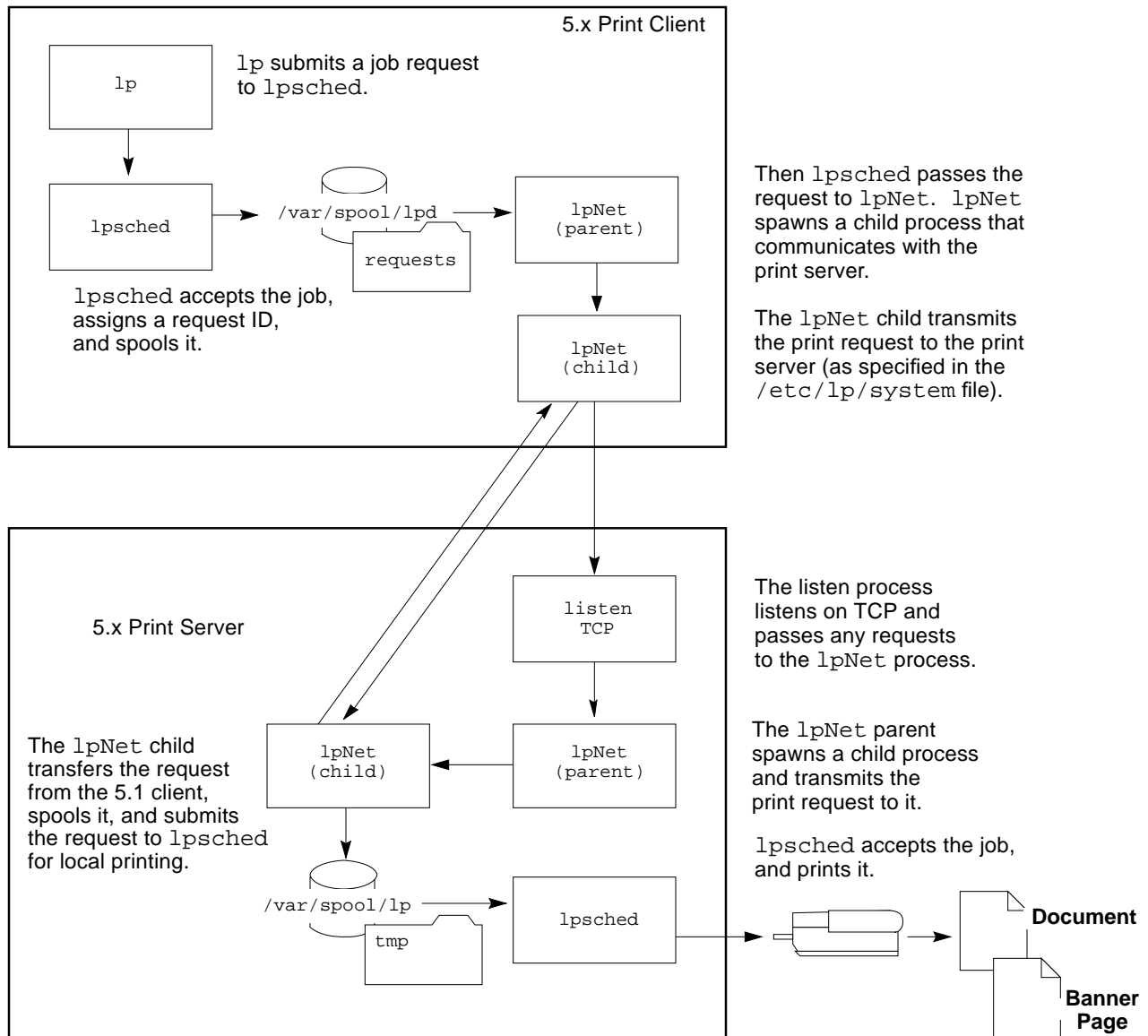


Figure B-5 Network Printing Between a SunOS 5.x Print Client and a SunOS 5.x Print Server

About Customizing the LP Print Service

Although the LP print service is designed to be flexible enough to handle most printers and printing needs, it does not handle every possible situation. You may have a printing request that is not accommodated by the standard features of the LP print service. Or you may have a printer that does not quite fit into the way the LP print service handles printers.

You can customize the LP print service in the following ways:

- Adjust the printer port characteristics
- Adjust the `terminfo` database
- Customize the printer interface program
- Create a print filter
- Define a form

Adjusting Printer Port Characteristics

The printer port characteristics set by the LP print service must be compatible with the printer communication settings.

If the default printer port settings provided by the LP print service do not work with a printer, refer to the printer manual to find out what settings the printer requires from the LP print service. See Table B-13 for a list of the `stty` settings most frequently used by the LP print service. See the manual page for the `stty(1)` command for a complete list of options.

Adjusting the `terminfo` Database for an Unsupported Printer

The LP print service uses an interface program and the `terminfo` database to initialize each printer and establish a selected page size, character pitch, line pitch, and character set.

Each printer is identified in the `terminfo` database with a short name. The name required by the `terminfo` database is identical to the kind of name used to set the `TERM` shell variable. This name is also the printer type you specify when setting up a printer. For example, the entries for different types of PostScript printers are in `/usr/share/lib/terminfo/P`. The default entries provided with the SunOS 5.x system are `PS` (for PostScript) and `PSR` (for

PostScript Reverse). For more information about finding entries in the `terminfo` database, see “Printer Entries in the `terminfo` Database on page 84.

If you cannot find a `terminfo` entry for your printer, you still may be able to use the printer with the LP print service, without automatic selection of page size, pitch, and character sets. However, you may have trouble keeping the printer set in the correct modes for each print request.

If there is no `terminfo` entry for your type of printer, you can either customize the interface program used with the printer or add an entry to the `terminfo` database.

The `terminfo` database contains hundreds of items defined for each terminal or printer. The LP print service, however, uses fewer than 50 of these items. Table B-14 on page 346 lists the required items.

Customizing the Printer Interface Program

Any printer interface program performs the following tasks:

- It initializes the printer port, if necessary. The standard printer interface program uses the `stty` command to initialize the printer port.
- It initializes the printer hardware. The standard printer interface program gets the control sequences from the `terminfo` database and the `TERM` shell variable.
- It prints a banner page, if necessary.
- It prints the number of copies specified by the print request.



Caution – If you have a printer interface program from a release of UNIX System V prior to Release 3.2, it will probably work with the SunOS 5.x LP print service. However, several `-o` options have been standardized in the SunOS 5.x print service and will be passed to every printer interface program. These options may interfere with similarly named options used by the old interface.

The LP print service, not a printer interface program, is responsible for opening the printer port. The printer port is given to the printer interface program as standard output, and the printer is identified as the “controlling terminal” for the printer interface program so that a “hang-up” of the port will cause a SIGHUP signal to be sent to the printer interface program.

If you have a printer that is not supported by the standard printer interface program, you can furnish your own printer interface program. You can copy the standard program, and then tell the LP print service to use it for a specified printer. But first, you need to understand what is in the standard program. The following section describes the standard program.



Caution – A customized printer interface program must not terminate the connection to the printer or “uninitialize” the printer in any way.

The Standard Printer Interface Program

The standard (model) printer interface program, `/usr/lib/lp/model/standard`, is used by the LP print service to set the printing defaults shown in Table B-9.

Table B-9 Default Printer Port Characteristics

Characteristic	Default Setting
Default filter	None
Character pitch	None
Line pitch	None
Page width	None
Page length	None
Character set	None
stty options	9600 cs8 -cstopb -parenb -parodd ixon -ixany opost -olcuc onlcr -ocrnl -onocr -onlret -ofill nl0 cr0 tab0 bs0 vt0 ff0
Exit code	0

Customizing `stty` Modes

If you need to change the terminal characteristics, like baud rate or output options, look for the section of the standard printer interface program that begins with the following comment:

```
## Initialize the printer port
```

Exit Codes

When printing is complete, your interface program should exit with a code that shows the status of the print job. The exit code is the last entry in the printer interface program.

Table B-10 shows the exit codes and how they are interpreted by the LP print service.

Table B-10 Printer Interface Program Exit Codes

Code	Meaning to the LP Print Service
0	The print request has been successfully completed. If a printer fault occurred, it has been cleared.
1 to 127	A problem was encountered when printing a request (for example, too many non-printable characters, or the request exceeds the printer capabilities). The LP print service notifies the person who submitted the request that there was an error when printing it. This error will not affect future print requests. If a printer fault has occurred, it has been cleared.
128	This code is reserved for internal use by the LP print service. Interface programs must not exit with this code.
129	A printer fault was encountered when printing the request. This fault will affect future print requests. If the fault recovery for the printer directs the LP print service to wait for the administrator to fix the problem, the LP print service disables the printer. If the fault recovery is to continue printing, the LP print service will not disable the printer, but it will try printing again in a few minutes.
>129	These codes are reserved for internal use by the LP print service. Interface programs must not exit with codes in this range.

If the program exits with a code of 129, the system administrator (root) is alerted of a printer fault. Unfortunately, if the printer interface program exits, the LP print service must also reprint the request from the beginning, after the fault has been cleared. If you do not want the entire request to be reprinted,

you can have the interface program send a fault message to the LP print service, but wait for the fault to clear. When the fault clears, the interface program can resume printing the file. When printing is finished, the printer interface program can give a zero exit code, just as if the fault had never occurred. An added advantage to this approach is that the interface program can detect when the fault is cleared automatically, so that the administrator does not need to re-enable the printer.

Fault Messages

You can use the `lp.tell` program to send fault messages to the LP print service. This program is referenced by the `LPTELL` shell variable in the standard printer interface code. The program takes standard input and sends it to the LP print service, where it is put into the message that alerts the administrator to the printer fault. If its standard input is empty, `lp.tell` does not initiate an alert. For an example of how the `lp.tell` program is used, examine the standard printer interface code immediately after the following comment:

```
# Here's where we set up the $LPTELL program to capture fault
messages
```

If you use the special exit code 129 or the `lp.tell` program, the printer interface program does not need to disable the printer itself. The interface program can disable the printer directly, but doing so will override the fault-alerting mechanism. Alerts are sent only if the LP print service detects that the printer has a fault, and the special exit code and the `lp.tell` program are its main detection tools.

If the LP print service has to interrupt the printing of a file at any time, it kills the interface program with a signal `TERM` (trap number 15, see `kill(1)` and `signal(3B)`). If the printer interface program dies from receipt of any other signal, the LP print service assumes that future print requests will not be affected, and continues to use the printer. The LP print service notifies the person who submitted the request that the request has not been finished successfully.

When the interface is first invoked, the signals `HUP`, `INT`, `QUIT`, and `PIPE` (trap numbers 1, 2, 3, and 13) are ignored. The standard interface changes this so that these signals are trapped at appropriate times. The standard interface interprets receipt of these signals as warnings that the printer has a problem; when it receives a signal, it issues a fault alert.

Using a Customized Printer Interface Program

You can create a customized printer interface program and use it in place of the standard printer interface program on the print server. To do so, you have to use the `lpadmin` command to register the program with the LP print service for a specific printer.

Creating a New Print Filter

A filter is used by the LP print service each time it has to print a type of file that is not acceptable to a printer. Creating a new print filter is not easy, and usually requires extensive experimentation. A print filter can be as simple or as complex as needed. Filters contain input types, output types, and complex options that provide a language to process command-line arguments within the filter.

The SunOS 5.x print service provides filter programs in the `/usr/lib/lp/postscript` directory. These filters cover most PostScript printing situations—where the destination printer requires the data to be in PostScript format. A set of filter descriptor files are provided in the `/etc/lp/fd` directory. These files describe the characteristics of the filters (for example, fast or slow filter), and point to the filter programs (for example, `/usr/lib/lp/postscript/postdaisy`).

If you have non-PostScript printers, you have to create and add print filters as required. First, you need to understand what print filters are and the requirements that must be met by a filter program.

Understanding Print Filters

This section describes different types of filters, how they convert file content, handle special modes, and detect printer faults.

Types of Filters

There are two types of print filters: fast filters and slow filters.

Fast filters do not require much processing time to prepare a file for printing. In addition, they must have access to the printer when they run. A print filter that detects printer faults must be a fast filter. Any filter that uses the `PRINTER` keyword as a filter option must be installed as a fast filter.

Slow filters require a lot of processing time to prepare a file for printing. In addition, slow filters do not require access to the printer when they run. Slow filters are run in the background so that they do not tie up a printer, allowing other files that do not need slow filtering to be printed. Slow filters that are specified by users (using the `-y` option to the `lp` command) must be run on the system where the print request was issued. The LP print service cannot pass values for filters to remote systems. It can, however, match a file content type to a content type on a remote system. If you want to activate special filter modes on a remote system, you must do so by specifying content types (instead of slow filters) that tell the LP print service to match input types and output types.

Converting Files

The LP print service uses print filters to convert files from one content type to another. You can specify the file content types for each printer. When a user submits a file, specifying its content type, the LP print service finds a printer that can print files of that content type. Because many applications can generate files for various printers, this is often sufficient. However, some applications may generate files that cannot be printed on any available printers.

Each time the LP print service receives a request to print a type of file that is in a format that cannot be accepted directly by a printer, the LP print service tries to match the content type of the print request with the content type of the available (or specified) printer. If there is a match, the file can be sent directly to the printer without filtering. If no match is found, or if the content type specifies that a filter be used, the LP print service tries to match the content type of the file with the input content type of available filters, and match the output type of the filter with the content type of the printer. When an appropriate filter is found, the print request is passed through the filter.

Handling Special Printing Modes

A print filter handles special modes and requests to print specific pages. A special printing mode is needed to print any characteristics of print requests that require a customized filter. Filters handle the following characteristics:

- Printer type
- Character pitch
- Line pitch
- Page length

- Page width
- Pages to print
- Character set
- Form name
- Number of copies

The LP print service provides default settings for these characteristics; however, a print filter may handle some characteristics more efficiently. For example, some printers can handle multiple copies more efficiently than the LP print service, in which case, you may want to provide a filter for multiple-copy page control.

Detecting Printer Faults

Each printer has its own way of detecting printer faults and transmitting fault signals to the LP print service. The LP print service only checks for hang-ups (loss of carrier) and excessive delays in printing.

Some printers provide good fault coverage and can send a message describing the reason for a fault. Other printers indicate a fault by using signals other than the signals indicating loss of carrier signal or shut off of data flow. A filter is required to interpret this additional printer fault information.

A filter can also put a print request on hold, wait for a printer fault to clear, and then resume printing. In this way, the print request that was interrupted does not need to be reprinted in its entirety. Only a filter that knows the control sequences used by a printer can determine where to break a file into pages. Consequently, only such a filter can find the place in the file where printing should start again when a fault is cleared.

When a print filter generates messages, those messages are handled by the LP print service, and alerts are sent to the system administrator, if alerts are enabled. See “Setting Printer Fault Alerts” on page 156.

Requirements for a Print Filter Program

A print filter can be simple or complex, but it has to meet the following requirements:

- The filter should get the contents of a file from its standard input and send the converted file to the standard output.

- A program cannot be used as a filter if it references external files. You may be tempted to use a program like `troff`, `nroff`, or a similar word processing program as a filter. The LP print service does not recognize references to other files, known as *include files*, from a filter program. Because `troff` and `nroff` allow include files, they may fail when used as filters. If the program needs other files to complete its processing, it should not be used as a filter.
- The filter should not depend on files that normally would not be accessible to a user. If a filter fails when run directly by a user, it will fail when run by the LP print service.
- A slow filter can send messages about errors in the file to standard error. A fast filter should not. Error messages from a slow filter are collected and sent to the user who submitted the print request.
- If a slow filter dies because it received a signal, the print request is stopped and the user who submitted the request is notified. Likewise, if a slow filter exits with a non-zero exit code, the print request is stopped and the user is notified. The exit codes from fast filters are treated differently.

If you want the filter to detect printer faults, the filter must also meet the following requirements:

- If possible, the filter should wait for a fault to be cleared before exiting. Additionally, it should continue to print at the top of the page where printing stopped after the fault clears. If the administrator does not want this contingency followed, the LP print service will stop the filter before alerting the administrator.
- The filter should send printer fault messages to its standard error as soon as the fault is recognized. It does not have to exit, but rather it can wait for the fault to be cleared.
- The filter should not send messages about errors in the file to standard error. These messages should be included in the standard output, where they can be read by the user.
- The filter should exit with a zero exit code if the file is finished printing (even if errors in the file have prevented it from being printed correctly).
- The filter should exit with a non-zero exit code only if a printer fault has prevented it from finishing a print request.
- When added to the filter table, the filter must be added as a fast filter.

Defining a Print Filter

When defining a new print filter, in addition to writing a filter program, you must define the characteristics of its use to the LP print service:

- Name of the filter program to run
- The types of input it accepts
- The types of output it produces
- The types of printers to which it can send jobs
- The names of specific printers to which it can send jobs
- The type of filter (either fast or slow)
- Options

You can type the characteristics as direct input to the `lpfilter` command. You also can create a file that specifies the filter's characteristics, and use the file name as input to the `lpfilter` command. Such a file is called a *filter descriptor file* and should be located in the `/etc/lp/fd` directory. These files are not the filters themselves, but rather point to the filters.

Whether you store the information in a file, or enter it directly on the command line, use the following format:

```
Command: command-pathname [options]
Input types: input-type-list
Output types: output-type-list
Printer types: printer-type-list
Printers: printer-list
Filter type: fast or slow
Options: template-list
```

The information can be arranged in any order, and not all the information is required. When you do not specify values, those shown in Table B-11 are assigned by default. They are not very useful, which is why you should specify explicit values.

Table B-11 Default Values for `lpfilter` Arguments

Item	Default
Input types	any
Output type	any

Table B-11 Default Values for `lpfilter` Arguments (Continued)

Item	Default
Printer types	any
Printers	any
Filter type	slow

The filter characteristics are defined as follows:

- **Command** – The full path of the filter program. If there are any fixed options that the program always needs, include them here.
- **Input types** – The list of file content types that the print filter can process. The LP print service does limit the number of input types, but most filters can accept only one type. Several file types may be similar enough that the filter can deal with them. You can use whatever names you like, using a maximum of 14 alphanumeric characters and dashes. Do not use underscores as part of the input type name.
Because the LP print service uses these names to match a filter to a file type, follow a consistent naming convention. For example, if more than one filter can accept the same input type, use the same name for that input type when you specify it for each filter. Inform your users of these names so they know how to identify the file type when submitting a file for printing.
- **Output types** – The list of file types that the filter can produce as output. For each input type, the filter produces a single output type. The output type may vary, however, from job to job. The name of the output type is restricted to 14 alphanumeric characters and dashes.
The output type names should either match the types of available (local or remote) printers, or match the input types handled by other filters. The LP print service groups filters in a shell pipeline if it finds that several passes by different filters are needed to convert a file. It is unlikely that you will need this level of sophistication, but the LP print service allows it. Try to find a set of filters that takes as input types all the different files the users may want printed, and that converts those files directly into file types the printer can handle.
- **Printer types** – The list of printer types into which the print filter can convert files. For most printers and filters, you can leave this part of the filter definition blank, because it is identical to the list of output types. But it can be different. For example, you could have a printer with a single printer

type for purposes of initialization, but which can recognize several different file content types. Essentially, this printer has an internal filter that converts the various file types into one that it can handle. Thus, a filter may produce one of several output types that match the file types that the printer can handle. The print filter should be marked as working with that printer type. As another example, you may have two different models of printers that are listed as accepting the same file types. Due to slight differences in manufacture, however, one printer deviates in the results it produces. You label the printers as being of different printer types, say A and B, where B is the one that deviates. You create a filter that adjusts files to account for the deviation produced by printers of type B. Because this filter is needed only for those printer types, you would list it as working only on type B printers.

- **Printers** – Usually, a print filter should be able to work with all printers that accept its output, so you can usually skip this part of the filter definition. You may, however, have some printers that are or inappropriate for the output that the filter produces. For example, you may want to dedicate one printer for fast turnaround, only sending files that require no filtering to that printer. Other printers of identical type may be used for files that need extensive filtering before they can be printed.
- **Filter type** – The LP print service recognizes fast and slow filters. Fast filters incur little overhead when preparing a file for printing, and they must have access to the printer when they run. A filter that is supposed to detect printer faults must be a fast filter. A filter that uses the `PRINTER` keyword as a filter option must be a fast filter. Slow filters incur lots of overhead in preparing a file, and do not require access to a printer to run. The LP print service runs slow filters in the background, without tying up a printer. Files that do not need slow filtering can move ahead in the print queue, and printers are not left idle while a slow filter works on one file.

Slow filters that are invoked by printing modes (using the `-y` option of the `lp` command), must be run on the system from which the print request originated. The LP print service cannot pass values for modes to print servers. It can, however, match a file content type (specified after the `-T` option of the `lp` command) to a content type on a print server. Therefore, if you want to activate special modes on a print server, you must specify content types that permit the LP print service to match input types and output types.

- Options – Options specify how different types of information are converted into command line arguments to the filter command. This information may include specifications from a user (with the print request), the printer definition, and the specifications implemented by any filters used to process the request.

Defining Print Filter Options With Templates

There are 13 sources of information, each of which is represented by a *keyword*. Each option is defined in a *template*. A template is a statement in a filter definition that defines an option to be passed to the filter command, based on the value of one of the characteristics of the filter. A template has the following format:

The options specified in a filter definition may include none, all, or any subset of the 13 keywords. In addition, a single keyword may be defined more than once, if multiple definitions are required for a complete filter definition. See Table B-12 for a description of the 13 keywords.

A print filter definition can include more than one template. Multiple templates are entered on a single line and separated with commas, or they are entered on separate lines, preceded by the `Options:` prefix.

The format of a template is as follows:

keyword pattern = replacement

The *keyword* identifies the type of option being registered for a particular characteristic of the filter.

The pattern is specific option for the keyword. The replacement is what happens when the keyword has the noted value.

For an example of how an option is defined for a particular filter, suppose you want to have the print service scheduler assign print requests to filters following this criteria:

- If the type of `OUTPUT` to be produced by the filter is `impress`, then pass the `-I` option to the filter.
- If the type of `OUTPUT` to be produced by the filter is `postscript`, then pass the `-P` option to the filter.

To specify these criteria, provide the following templates as options to the `lpfilter` command:

```
Options: OUTPUT impress=-I, OUTPUT postscript=-P
```

If the `Options:` line becomes too long, put each template on a separate line, as follows:

```
Options: OUTPUT impress=-I
Options: OUTPUT postscript=-P
```

In both templates, the *keyword* is defined as `OUTPUT`. In the first template, the pattern is `impress` and the value of the *replacement* is `-I`. In the second template, the value of *pattern* is `postscript` and the value of *replacement* is `-P`.

The 13 keywords shown in Table B-12 are available for defining `Options` in a print filter definition.

Table B-12 Print Filter Options Keywords

Characteristic	Keyword	Possible Patterns	Example
Content type (input)	INPUT	content-type	troff
Content type (output)	OUTPUT	content-type	postscript, impress
Printer type	TERM	printer-type	att495
Printer name	PRINTER	printer-name	lp1
Character pitch	CPI	scaled-decimal	10
Line pitch	LPI	scaled-decimal	6
Page length	LENGTH	scaled-decimal	66
Page width	WIDTH	scaled-decimal	80
Pages to print	PAGES	page-list	1-5,13-20
Character set	CHARSET	character-set	finnish

Table B-12 Print Filter Options Keywords (Continued)

Characteristic	Keyword	Possible Patterns	Example
Form name	FORM	form-name	invoice2
Number of copies	COPIES	integer	3
Special modes	MODES	mode	landscape

To find out which values to supply for each type of template (that is, for the *pattern* and *replacement* arguments for each keyword), consider the following:

- The values for the `INPUT` and `OUTPUT` templates come from the file content type that needs to be converted by the filter, and the output type that has to be produced by the filter, respectively.
- The value for the `TERM` template is the printer type.
- The value for the `PRINTER` template is the name of the printer that will be used to print the final output.
- The values for the `CPI`, `LPI`, `LENGTH`, and `WIDTH` templates come from the user's print request, the form being used, or the default values for the printer.
- The value for the `PAGES` template is a list of pages that should be printed. Typically, it is a list of page ranges separated by commas. Each page range consists of a pair of numbers separated by a dash, or a single number. (For example, `1-5,6,8,10` indicates pages 1 through 5, plus pages 6, 8, and 10). However, whatever value was given in the `-P` option to a print request is passed unchanged.
- The value for the `CHARSET` template is the name of the character set to be used.
- The value for the `FORM` template is the name of the form requested by the `-f` option of the `lp` command (the command used to submit a print request).
- The value of the `COPIES` template is the number of copies of the file to print. If the filter uses this template, the LP print service will reduce to one the number of copies of the filtered file it prints, since this "single copy" includes the multiple copies produced by the filter.

- The value of the `MODES` template comes from the `-y` option of the `lp` command. Because a user can specify several `-y` options, there may be several values for the `MODES` template. The values will be applied in the left-to-right order given by the user.

The *replacement* part of a template shows how the value of a template should be given to the filter program. It is typically a literal option, sometimes with the placeholder asterisk (*) included to show where the value goes. The *pattern* and *replacement* also can use the regular expression syntax of `ed(1)` for more complex conversion of user input options into filter options. All regular expression syntax of `ed(1)` is supported, including the `\(. . . \)` and `\n` constructions, which can be used to extract portions of the *pattern* for copying into the *replacement*, and the `&`, which can be used to copy the entire *pattern* into the *replacement*.

Note – If a comma or an equals sign (=) is included in a *pattern* or a *replacement*, precede it with a backslash (\). A backslash in front of any of these characters is removed when the *pattern* or *replacement* is used.

Creating a New Printer Form

When you want to provide a new form, you must define its characteristics by entering information about nine required characteristics (page length, page width, and so on) as input to the `lpforms` command. The LP print service uses this information for two purposes:

- To initialize the printer so that printing is done properly on the form
- To send reminders to the system administrator about how to handle the form

Before running the `lpforms` command, gather the following information about the new form:

- *Page length* – The length of the form, or the length of each page in a multipage form. This information can be the number of lines, or the size in inches or centimeters.
- *Page width* – The width of the form, in characters, inches, or centimeters.

- *Number of pages* – The number of pages in a multipage form. The LP print service uses this number with a print filter (if available) to restrict the alignment pattern to a length of one form. See the description of alignment pattern below. If no filter is available, the LP print service does not truncate the output.
- *Line pitch* – A measurement of how close lines appear on the form. This is also called *leading*. It is the distance between two lines, from baseline to baseline, measured by either lines per inch or lines per centimeter.
- *Character pitch* – A measurement of how close together characters appear on the form. It is the distance between characters, measured by either characters per inch or characters per centimeter.
- *Character set choice* – The character set, print wheel, or font cartridge that should be used when this form is used. Users may choose a different character set for their own print requests when using this form, or you can insist that only one character set be used.
- *Ribbon color* – If the form should always be printed using a certain color ribbon, the LP print service can give a mount alert message indicating which color to use.
- *Comment* – Any remarks that might help users understand the form. For example, the remarks could indicate the name of the form, its revision, its purpose, or restrictions on its use.
- *Alignment pattern* – A sample file that the LP print service uses to fill one blank form. When mounting the form, you can print this pattern on the form to align it properly. You can also define a content type for this pattern so that the print service knows how to print it.

Note – The LP print service does not try to mask sensitive information in the alignment pattern. If you do not want sensitive information printed on sample forms—for example when you align checks—then you should mask the appropriate data. The LP print service keeps the alignment pattern stored in a safe place, where only those logged in as `root` or `lp` can read it.

When you have gathered the information for the form, you enter it as input to the `lpforms` command. You should record this information first in a separate file so you can edit it before entering it with `lpforms`. You can then use the file as input instead of typing each piece of information separately after a prompt.

Instructions for Customizing the LP Print Service

This section includes instructions for the following tasks:

- Adjusting the printer port characteristics
- Adjusting the `terminfo` database for an unsupported printer
- Providing a custom printer interface program
- Providing a new print filter
- Providing a new form

For many tasks, you will find an example of user input and system output after the instructions.

▼ How to Adjust the Printer Port Characteristics

1. On the system to which the printer is connected, become root or lp.
2. Type `lpadmin -p printer-name -o "stty=options"` and press Return. You can change more than one `stty` option setting by enclosing the options in single quotation marks and separating the options by spaces. The following table shows the default settings used by the LP print service. Change them as needed.

Table B-13 `stty` Default Settings Used by the LP Print Service

Option	Meaning
9600	Set baud to 9600
cs8	Set 8-bit bytes
-cstopb	Send one stop bit per byte
-parity	Do not generate parity
ixon	Enable XON/XOFF (also known as START/STOP or DC1/DC3)
opost	Do "output post-processing" using the settings listed below:
-olcuc	Do not map lowercase to uppercase
onlcr	Change line feed to carriage return/line feed
-ocrnl	Do not change carriage returns into line feeds
-onocr	Output carriage returns even at column 0
n10	No delay after line feeds

Table B-13 `stty` Default Settings Used by the LP Print Service (Continued)

Option	Meaning
<code>cr0</code>	No delay after carriage returns
<code>tab0</code>	No delay after tabs
<code>bs0</code>	No delay after backspaces
<code>vt0</code>	No delay after vertical tabs
<code>ff0</code>	No delay after form feeds

Example of Adjusting the Printer Port Characteristics

For example, suppose your printer requires you to enable parity, set it to odd, and set a 7-bit character size. You would type the following command:

```
terra% su
Password:
# lpadmin -p luna -o "stty='parenb parodd cs7'"
```

The `stty` option `parenb` enables parity checking/generation, `parodd` sets odd parity generation, and `cs7` sets the character size to 7 bits.

For example, suppose your printer requires you to enable hardware flow control. You would type the following command:

```
terra% su
Password:
# lpadmin -p oak -o "stty='parenb parodd cs7'"
```

The `stty` option `parenb` enables parity checking/generation, `parodd` sets odd parity generation, and `cs7` sets the character size to 7 bits.

▼ How to Adjust the `terminfo` Database for an Unsupported Printer

1. Determine a correct `TERM` name for the printer.

a. On the system to which the printer is connected, type `cd /usr/share/lib/terminfo` and press Return.

b. Type `ls -l *` and press Return.

The file names shown in the listing are all valid `TERM` variables. These are the printer types you can specify when setting up a printer. Use them as a guide for picking a name for the printer. You should make sure none of the existing entries will support the new printer (by trying to set up the printer with an entry for a similar printer, if there are any).

2. Construct a database entry for a new printer.

The following table shows the items you must define in the `terminfo` entry to add a new printer to the LP print service. For more details about the structure of the `terminfo` database, see `terminfo(4)`.

Table B-14 `terminfo` Items Required to Add an Unsupported Printer

Item	Meaning
Booleans:	
<code>cpix</code>	Changing character pitch changes resolution
<code>daisy</code>	Printer needs operator to change character set
<code>lpix</code>	Changing line pitch changes resolution
Numbers:	
<code>bufsx</code>	Number of bytes buffered before printing
<code>cols</code>	Number of columns in a line
<code>cps</code>	Average print rate in characters per second
<code>it</code>	Tabs initially every <i>n</i> spaces
<code>lines</code>	Number of lines on a page
<code>orc</code>	Horizontal resolution, in units per character
<code>orhi</code>	Horizontal resolution, in units per inch
<code>orl</code>	Vertical resolution, in units per line
<code>orvi</code>	Vertical resolution, in units per inch
Strings:	
<code>chr</code>	Change horizontal resolution
<code>cpi</code>	Change number of characters per inch

Table B-14 terminfo Items Required to Add an Unsupported Printer (Continued)

Item	Meaning
cr	Carriage return
csnm	List of character set names
cudl	Down one line
cud	Move carriage down <i>n</i> lines
cuf	Move carriage right <i>n</i> columns
cvr	Change vertical resolution
ff	Page eject
hpa	Horizontal position absolute
ht	Tab to next 8-space Tab stop
if	Name of initialization file
iprogr	Path name of initializing program
is1	Printer initialization string
is2	Printer initialization string
is3	Printer initialization string
lpi	Change number of lines per inch
mgc	Clear all margins (top, bottom, and sides)
rep	Repeat a character <i>n</i> times
rwidm	Disable double-wide printing
scs	Select character set
scsd	Start definition of a character set
slines	Set page length to <i>n</i> lines per page
smgl	Set left margin at current column
smglp	Set left margin
smgr	Set right margin at current column
Strings:	
smgrp	Set right margin
smglr	Set both left and right margins

Table B-14 terminfo Items Required to Add an Unsupported Printer (Continued)

Item	Meaning
msgt	Set top margin at current line
smgtp	Set top margin
smgb	Set bottom margin at current line
smgbp	Set bottom margin
smgtb	Set both top and bottom margins
swidm	Enable double-wide printing
vpa	Vertical position absolute

3. Type `tic filename` and press Return.

When you have created a file containing the new entry, you need to compile it into the database using the `tic(1M)` program.

4. Stop the LP print service by typing `lpshut` and press Return.

5. Restart the LP print service by typing `lpsched` and press Return.

The new information is registered with the LP print service.

▼ How to Provide a Custom Printer Interface Program

To customize the standard interface program:

1. On the print server, become root or lp.

2. Type `cp /var/spool/lp/model/standard custom-filename` and press Return.

This copies the standard printer interface program.

3. Change the copy of the standard printer interface program to fit your particular needs.

Refer to the description of the program in the section “The Standard Printer Interface Program” on page 329 to determine what you need to change.

4. Type `lpadmin -p printer-name -i custom-filename` and press Return to tell the LP print service to use the custom program.

You have to tell the LP print service to use the custom printer interface program; otherwise, the standard printer interface program is used. The custom printer interface program is registered with the LP print service, and will be used by that printer when users submit print requests.

To use a customized interface program for another printer:

- ◆ Type `lpadmin -p printer-name1 -e printer-name2` and press Return.

After you have created a custom printer interface program, you can use it for other printers. The printer interface program from *printer-name2* is registered with the LP print service for *printer-name1*. Note that the program is still registered with the LP print service for *printer-name2*.

▼ How to Provide a New Print Filter

1. On the print server or print client, become root or lp.

2. Write a print filter program.

It would be difficult to give you a step-by-step procedure for this. Read the information in the section “Creating a New Print Filter” on page 332 for guidelines. The filter program is a binary executable. By convention, filter programs for PostScript printers are located in the `/usr/lib/lp/postscript` directory. You should put programs you create under `/usr/lib/lp` in a directory of your choosing.

3. Create a print filter definition.

You should save the printer definition in a text file. By convention, filter definitions are located in the `/etc/lp/fd` directory and are identified with the `.fd` suffix. The definition tells the LP print service about the filter, what program to run, what kind of conversion it does, on so on. The filter definition has the following format:

```
Command: command-pathname [options]
Input types: input-type-list
Output types: output-type-list
Printer types: printer-type-list
Printers: printer-list
Filter type: fast or slow
Options: template-list
```

4. Add the print filter to the LP print service, using the definition from a file, or enter it directly at the command line.

This step registers the new filter with the LP print service.

a. To use a file, type `lpfilter -f filter-name -F filename.fd` and press Return.

The filter definition from the file named is added to the `/etc/lp/filter.table` file using the filter name you specify.

```
terra# lpfilter -f daisytroff -F /etc/lp/fd/daisytroff.fd
```

b. To enter the definition, type `lpfilter -f filter-name - filter-definition` and press Control-d.

The filter definition you type is added to the `/etc/lp/filter.table` file using the filter name you specify.

Examples of Print Filter Definitions

Print Filter to Convert N37 or Nlp to simple

To provide the following filter definition for a filter called `col`, you would type:

```
Input types: N37, Nlp, simple
Output types: simple
Command: /usr/bin/col
Options: TERM 450 = -b, MODES expand = -x
Options: INPUT simple = -p -f
```

Note – If you provide more than one definition (that is, more than one line) for any filter characteristic other than `Options`, only the second definition will be used by the print service.

After you register this definition with the print service by entering it as input to the `lpfilter` command, users' print requests will be handled as follows:

- When a user enters this command:

```
% lp -y expand report.dec10
```

The filter command is run with the following arguments:

```
/usr/bin/col -x -p -f
```

- When a user enters this command:

```
% lp -T N37 -y expand report.dec10
```

The filter command is run with the following arguments:

```
/usr/bin/col -x
```

- When the default printer is not of type 450 and a user enters this command:

```
% lp -y expand -T 450 report.dec10
```

The filter command is run with the following arguments:

```
/usr/bin/col -b -x
```

Print Filter to Convert From troff to PostScript

In this example, the filter program is called `/usr/lib/lp/postscript/dpost`. It takes one input type, `troff`, produces a `postscript` output, and works with any printer of type `PS` (for PostScript). You have decided that users need to give just the abbreviations `port` or `land` when they ask for the paper orientation to be portrait mode or landscape mode, respectively. Because these options are not intrinsic to the LP print service, users must specify them using the `-y` option to the `lp` command.

The print filter definition would look like this:

```
Input types: troff
Output types: postscript
Printer types: PS
Filter type: slow
Command: /usr/lib/lp/postscript/dpost
Options: LENGTH * = -l*
Options: MODES port = -pp, MODES land = -pl
```

A user submitting a troff file type for printing on a PostScript printer (type PS), with requests for landscape orientation and a page length of 60 lines, would type the following command:

```
% lp -T troff -o length=60 -y land -d printer-name filename
```

Then the dpost filter would be invoked by the LP print service to convert the file as follows:

```
/usr/lib/lp/postscript/dpost -l60 -pl printer-name filename
```

Add the following option template to the previous example:

```
Options: MODES group \=([1-9]) = -n\1
```

This template converts a MODES option of this form:

```
-y group=number
```

Into filter options:

```
-number
```


So, if a user gives the following command:

```
% lp -y group=4
```

The `dpost` command includes the following options:

```
-n4
```

For additional examples, become root and run the command:

```
# lpfilter -f all -l
```

This command will list all of the currently installed print filters.

▼ How to Provide a New Form

1. Determine the definition and name to use for the new form.

The form name can be anything you choose, as long as it does not contain more than 14 alphanumeric characters and underscores. The information must be in the following format:

```
Page length: scaled number  
Page width: scaled number  
Number of pages: integer  
Line pitch: scaled number  
Character pitch: scaled number  
Character set choice: character-set-name [,mandatory]  
Ribbon color: ribbon-color  
Comment:  
informal notes about the form  
Alignment pattern: [content-type] alignment pattern
```

The optional phrase [*,mandatory*] means that the user cannot override the character set choice in the form. The *content-type* can be given, although this is optional, with an alignment pattern. If this attribute is given, the print service uses it to determine, as necessary, how to filter and print the file.

With two exceptions, the information may appear in any order. The exceptions are the `Alignment` pattern (which must always be last), and the `comment` (which must always follow the line with the `Comment :` prompt). If the comment contains a line beginning with a key phrase (like `Page length`, `Page width`, and so on), precede that line with a `>` character so the key phrase is not at the beginning of the line. The initial `>` character is stripped from the comment and is not displayed.

Not all of the information must be given. When you do not specify values for the items listed in Table B-15, the default values are assigned.

Table B-15 Default Form Values

Item	Default
Page length	66 lines
Page width	80 columns
Number of pages	1
Line pitch	6 lines per inch
Character pitch	10 characters per inch
Character set choice	Any
Ribbon color	Any
Comment	(No default)
Alignment pattern	(No default)

- 2. On the print server or print client, become root or lp.**
- 3. (Optional) Create a file containing the new form definition.**
- 4. Add the form to the LP print service, using the definition from a file, or enter it directly in the command line.**
 - To use the definition from a file, type `lpforms -f form-name -F filename` and press Return.
The form is registered with the LP print service. The form definition from the file named is added to the `/etc/lp/forms/form-name` directory under the file named `description`.

```
terra# lpforms -f medical -F /etc/lp/forms/medical.fmd
```

- ◆ **To enter the definition directly, type `lpforms -f form-name -
form-definition` and press **Control-d**.**

The form is registered with the LP print service. The form definition you type is added to the `/etc/lp/forms/form-name` directory under the file named `description`.

5. **Type `lpadmin -p printer-name -f allow:form-list` and press **Return**.**
The printer now has access to the specified forms. You can specify `all` for *form-list* to grant access to all the available forms. By default a printer does not have access to any forms. The *form-list* is a list of forms separated by commas with no spaces. If you use spaces to separate names, enclose the word `allow:` and the list of forms in double quotation marks.

```
terra# lpadmin -p luna -f allow:"medical,dental,check"
```

≡ *B*

Customizing sendmail Configuration Files



The `sendmail` program is a mail transport agent that uses a configuration file to provide “aliasing” and forwarding, automatic routing to network gateways, and flexible configuration. The SunOS 5.x operating system supplies standard configuration files that most sites can use. Chapter 9, “Setting Up and Administering Mail Services,” explains how to set up an electronic mail system using the standard files. This appendix explains how to customize `sendmail` configuration files if you need to tailor them to fit your site’s needs.

The `sendmail` program can use different types of communications protocols, like TCP/IP and UUCP. It also implements an SMTP server, message queueing, and mailing lists. Name interpretation is controlled by a pattern-matching system that can handle both domain-based naming and *ad hoc* conventions.

Sections in this appendix describe the following subjects:

- Command-line arguments to `sendmail`
- `sendmail` parameters that you can alter
- In-depth information about the configuration file for sites that need to write configuration file
- Brief explanations of several less-used features of `sendmail`

See *TCP/IP Network Administration Guide* for information about routers, gateways, and setting up an internetwork.

The `sendmail` program can accept domain-based naming, as well as arbitrary (older) name syntaxes—resolving ambiguities by using heuristics you specify. `sendmail` can also convert messages between disparate naming schemes. The domain technique separates the issue of physical versus logical naming. See *TCP/IP Network Administration Guide* for a complete description of Internet domain naming conventions.

Certain special cases can be handled by *ad hoc* techniques, like providing network names that appear local to hosts on other networks.

Features of the sendmail Program

The `sendmail` program provides the following features:

- It supports UNIX System V mail, UNIX version 7 mail, and Internet mail.
- `sendmail` is reliable. It is designed to correctly deliver every message. No message should ever be completely lost.
- `sendmail` uses existing software for delivery whenever possible.
- `sendmail` expands easily to complex environments, including multiple connections to a single network type (like with UUCP or Ethernet). `sendmail` considers the contents of a name as well as its syntax to determine which mailer to use.
- `sendmail` uses configuration files to control mail configuration.
- Groups can maintain their own mailing lists. Individuals can specify their own forwarding without modifying the domain-wide alias file (typically located in the domain-wide aliases maintained by NIS or NIS+).
- Each user can specify a custom mailer to process incoming mail, which can provide functions like returning an “I am on vacation” message. See the reference manual page for `vacation(1)` for more information.
- `sendmail` minimizes network traffic by batching recipients to a single host where possible, without assistance from the user.

Figure C-1 shows how `sendmail` interacts with the other programs in the mail system.

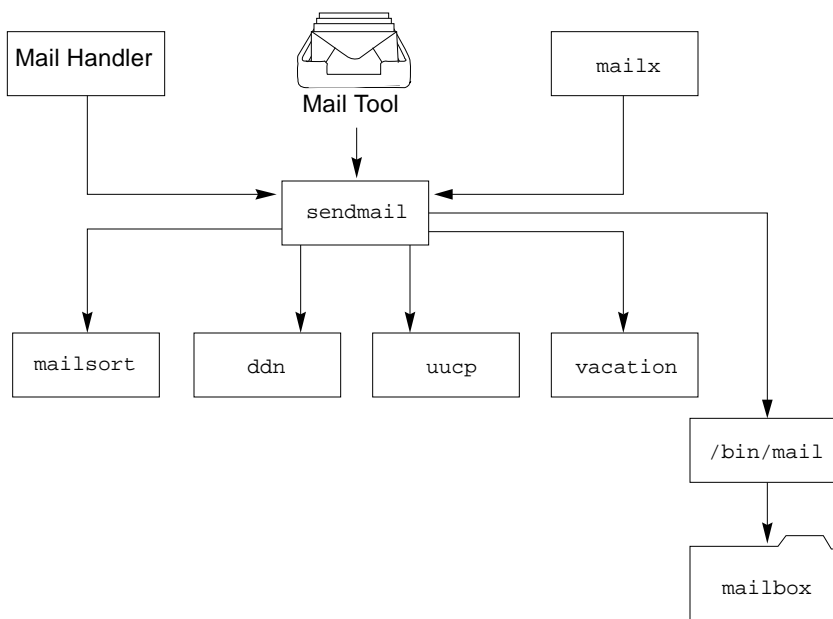


Figure C-1 Interaction of `sendmail` With Other Mail Programs

The user interacts with a mail-generating and sending program. When the mail is submitted, the mail-generating program calls `sendmail`, which routes the message to the correct mailer(s). Since some of the senders may be network servers and some of the mailers may be network clients, `sendmail` may be used as an Internet mail gateway.

Overview of `sendmail` Functions

The `sendmail` program is a message router that calls administrator-defined mailer programs to deliver messages. It collects a message from a program, like `mail`, edits the header of the message as required by the destination mailer, and calls appropriate mailers to do delivery or queueing for network transmission. When mailing to a file, however, `sendmail` delivers directly. New mailers can be added at minimum cost.

Interfaces to the Outside World

The `sendmail` program communicates with the outside world in three ways:

- Using the conventional argument vector/exit status
- Using pairs of pipes
- Using SMTP over a TCP connection

Argument Vector/Exit Status

The standard way to communicate with a process is by using the argument vector (command name and arguments). The argument vector sends a list of recipients, and the message body is sent on the standard input. If problems occur, anything that the mailer prints is collected and returned to the sender. After the message is sent, the exit status from the mailer is collected, and a diagnostic is printed, if appropriate.

SMTP Over Pipes

The SMTP protocol can be used to run an interactive lock-step interface with the mailer. A subprocess is still created, but no recipient names are passed to the mailer via the argument list. Instead, they are passed one at a time in commands sent to the processes' standard input. Anything appearing on the standard output must be a standard SMTP reply code.

SMTP Over a TCP Connection

This technique is similar to SMTP over pipes, except that it uses a TCP connection. SMTP over a TCP connection is normally used to connect to a `sendmail` process on another system. This method is exceptionally flexible because the mailer need not reside on the same machine.

How `sendmail` Works

When a sender wants to send a message, the sender issues a request to `sendmail` using one of the three methods described above. `sendmail` operates as described on the following pages.

Argument Processing and Address Parsing

If `sendmail` is called by using the argument vector or is connected to over a pipe, the arguments are first scanned and option specifications are processed. Recipient names are then collected, either from the command line or from the SMTP command, and a list of recipients is created. Aliases are expanded at this step, including mailing lists. As much validation as possible of the remote recipient is done at this step; syntax is checked, and local recipients are verified, but detailed checking of host names is deferred until delivery. Forwarding is also performed as the local recipients are verified.

`sendmail` appends each name to the recipient list after parsing. When a name is aliased or forwarded, the old name is retained in the list, and a flag is set that tells the delivery phase to ignore this recipient. This list is kept free from duplicates, preventing alias loops and duplicate messages from being delivered to the same recipient, as might occur if a person is in two groups.

Note – Users may receive duplicate copies of the same message when alias lists contain email addresses using different syntax for the same person. `sendmail` cannot always match the email addresses as duplicates of one another.

Message Collection

`sendmail` then collects the message. The message must have a header at the beginning. No formatting requirements are imposed on the message body except that they must be lines of text (in other words, binary data is not allowed). The header is stored in memory, and the body of the message is saved in a temporary file.

To simplify the program interface, the message is collected even if no names were valid. The message will be returned with an error.

Message Delivery

For each unique mailer and host in the recipient list, `sendmail` calls the appropriate mailer. Each mailer invocation sends to all users receiving the message on one host. Mailers that only accept one recipient at a time are handled properly.

The message is sent to the mailer by using one of the same three interfaces used to submit a message to `sendmail`. Each copy of the message is appended by a customized header. The mailer status code is caught and checked, and a suitable error message is given, as appropriate. The exit code must conform to a system standard, or a generic message (`Service unavailable`) is given.

Queueing for Retransmission

When the mailer returns a status that shows that it might be able to handle the mail later, `sendmail` queues the mail and tries again later.

Return to Sender

When errors occur during processing, `sendmail` returns the message to the sender for retransmission. The letter can be mailed back (when the mail comes from a different site) or written in the `dead.letter` file in the sender's home directory.

Message Header Editing

Certain editing of the message header occurs automatically. Header lines can be inserted under control of the configuration file. Some lines can be merged; for example, a `From:` line and a `Full-name:` line can be merged under certain circumstances.

Configuration File

Almost all configuration information is read at runtime from a text file, that includes macro definitions (the value of macros used internally), header declarations (the format of header lines that are specially processed, and lines that are added or reformatted), mailer definitions (giving information like the location and characteristics of each mailer), and name-rewriting rules (a limited pattern-matching system used to rewrite names).

How `sendmail` Is Implemented

You can follow flag arguments with recipient name arguments unless you run `sendmail` in SMTP mode. In brief, the format of recipient names is:

- Anything in parentheses is thrown away (as a comment).

- Anything in angle brackets (<>) is preferred over anything else. This rule implements the Internet standard that sends names of the following form to the electronic *system-name* rather than the human *username*:

```
username <system-name>
```

- Double quotation marks (") quote phrases; backslashes (\) quote characters. Backslashes cause otherwise equivalent phrases to compare differently—for example, `user` and `"use"` are equivalent, but `\user` is different from either of them.

Parentheses, angle brackets, and double quotation marks must be properly balanced and nested. The rewriting rules control the rest of the needed processing.

Mail to Files and Programs

Files and programs are legitimate message recipients. Files provide archival storage of messages, useful for project administration and history. Programs are useful as recipients in a variety of situations, for example, to use `mailsort` to sort mail, or to have the `vacation` program respond with an informational message when users are away.

Any name passing through the initial parsing algorithm as a local name is scanned for two special cases:

- If the prefix is a vertical bar (|), the rest of the name is processed as a shell command.
- If the user name begins with a slash (/), the name is used as a file name instead of a login name.

Message Collection

Once all recipient names are parsed and verified, the message is collected. The message comes in two parts: a message header and a message body. The header and the body are separated by a blank line.

The header is formatted as a series of lines in this form:

```
field-name: field-value
```

For example, a sample header might be:

```
From: John Smith <Smith@Podunk.edu>
```

field-value can be split across lines by starting the subsequent lines with a space or a Tab. Some header fields have special internal meaning, and have appropriate special processing. Other headers are simply passed through. Some header fields, like time stamps, may be added automatically.

The body is a series of text lines. It is completely uninterpreted and untouched, except that lines beginning with a dot have the dot doubled when transmitted over an SMTP channel. This extra dot is stripped by the receiver.

Message Delivery

The send queue is grouped by the receiving host before transmission to implement message batching. An argument list is built as the scan proceeds. Mail being sent to files is detected during the scan of the send list. The interface to the mailer is performed by using one of the techniques described in “Overview of sendmail Functions” on page 359.

After a connection is established, `sendmail` makes the per-mailer changes to the header and sends the result to the mailer. If any mail is rejected by the mailer, a flag is set to invoke the return-to-sender function after all delivery is complete.

Queued Messages

If the mailer returns a `temporary failure` exit status, the message is queued. A control file is used to describe the recipients and various other parameters. This control file is formatted as a series of lines, each describing a sender, a recipient, the time of submission, or some other parameter of the

message. The header of the message is stored in the control file so that the associated data file in the queue is just the temporary file that was originally collected.

Configuration Overview

Configuration is controlled primarily by a configuration file read at startup. Adding mailers or changing the name-rewriting or routing information does not require recompiling. The configuration file encodes macro definitions, header definitions, mailer definitions, name-rewriting rules, and options.

Macros

Macros can be used in several ways. Certain macros transmit unstructured textual information into the mail system, like the name that `sendmail` will use to identify itself in error messages. Other macros are unused internally, and can be used as shorthand in the configuration file.

Header Declarations

Header declarations inform `sendmail` of the format of known header lines. Knowledge of a few header lines is built into `sendmail`, like the `From:` and `Date:` lines.

Most configured headers are automatically inserted into the outgoing message if they do not exist in the incoming message. Certain headers are suppressed by some mailers.

Mailer Declarations

Mailer declarations specify the internal name of the mailer, some flags associated with the mailer, and an argument vector to be used on the call. This vector is expanded by macro before use.

Name-Rewriting Rules

Name-rewriting rules are the heart of name parsing in `sendmail`. They are an ordered list of pattern-replacement rules, which are applied to each name. For example, rule set zero determines which mailer to use. `sendmail` rewrites the message until it is in a form that can be parsed. When a pattern matches, the rule is reapplied until it fails.

The configuration file also supports the editing of names into different formats. For example, a name in this form:

```
ucsfcg1!tef
```

might be mapped into:

```
tef@ucsfcg1.UUCP
```

to conform to the internal syntax. Translations can also be done in the other direction for particular mailers.

Option Setting

Several options can be set from the configuration file. These include the path names of various support files, timeouts, default modes, and so forth.

Arguments to sendmail

Arguments to `sendmail` are listed and described in detail in the section “`sendmail` Command-Line Arguments” on page 409. Some of the important arguments are described in this section.

Queue Interval

The `-q` flag defines how often `sendmail` runs the queue. If you run in mode `i` or `b` (the default), this time can be relatively long, because it is only relevant when a host that was down comes back up. If, however, you run in `q` mode,

the time should be relatively short, since the flag defines the maximum amount of time that a message may sit in the queue. Typical queue time is set to between 15 minutes (-q15m) and one hour (-q1h).

Daemon Mode

If you allow incoming mail over a TCP connection, you should have a daemon running. Set the -bd flag in your /etc/rc3.d/S88sendmail file.

You can combine the -bd flag and the -q flag in one call:

```
# /usr/lib/sendmail -bd -q30m
```

Debugging

sendmail has many debug flags, which you set using the -d option. Each debug flag has a number and a level, where higher levels mean “print more information.” The convention is that you do not need to set levels greater than 9 unless you are debugging that particular piece of code. The syntax for debug flags is:

```
debug-flag :           -d debug-list
debug-list  :           debug-option[ , debug-option ] . . .
debug-option :         debug-range[ . debug-level ]
debug-range :         integer | integer-integer
debug-level :         integer
```

For example:

Table C-1

Debugging Flag	Description
-d12	Set flag 12 to level 1
-d12.3	Set flag 12 to level 3
-d3-17	Set flags 3 through 17 to level 1
-d3-17.4	Set flags 3 through 17 to level 4

If you have source code, you can refer to the list of debug flags in the code.

Trying a Different Configuration File

You can specify an alternative configuration file by using the `-C` flag; for example:

```
# /usr/lib/sendmail -Ctest.cf
```

uses the configuration file `test.cf` instead of the default `/etc/mail/sendmail.cf`. If you do not define a value for the `-C` flag, it uses the `sendmail.cf` file in the current directory.

Tuning Configuration Parameters

You can tune several configuration parameters, depending on the requirements of your site. You can set most of these parameters by using an option in the configuration file. For example, the line `OT3d` sets option `T` to the value `3d` (three days).

Time Values

All time intervals use a syntax of numbers and letters. For example, `10m` is 10 minutes, and `2h30m` is two-and-one-half hours.

Table C-2 lists the time symbols.

Table C-2 Time Syntax Options

Code	Description
s	Seconds
m	Minutes
h	Hours
d	Days
w	Weeks

Queue Interval

The argument to the `-q` flag specifies how often `sendmail` runs the queue. It is usually set to between 15 minutes (`-q15m`) and one hour (`-q1h`).

Read Timeouts

The `Or` option in the configuration file sets the read timeout. The default read timeout is `Or15m`. Although it is technically unacceptable within the published protocols, `sendmail` may time out when reading the standard input or when reading from a remote SMTP server. If your site has problems with read timeouts, set the read timeout to a larger value, like one hour (`Or1h`), to reduce the chance of several idle daemons piling up on your system.

Message Timeouts

The `OT` option in the configuration file sets the message timeout. The default message timeout is three days (`OT3d`). After a message has been in the queue for the message timeout period, the sender is notified that the message could not be delivered.

You can flush messages that have been hanging for a short period by running the queue with a short message timeout. For example:

```
# /usr/lib/sendmail -oT1d -q
```

runs the queue and flushes anything that is one day old or older.

Delivery Mode

The `Od` option in the configuration file sets the delivery mode. The default delivery mode is `Odbackground`. Delivery modes specify how quickly mail is delivered. Legal modes are shown in Table C-3.

Table C-3 Delivery Mode Options

Code	Description
i	Deliver interactively (synchronously)
b	Deliver in background (asynchronously)
q	Queue only (do not deliver)

There are tradeoffs. The interactive mode (i) passes the maximum amount of information to the sender, but it is hardly ever necessary. The queue mode (q) puts the minimum load on your machine, but means that delivery may be delayed for up to the queue interval. The background mode (b), the default, is a good compromise.

Load Limiting

Central mail machines often can be overloaded. The best solution is to dedicate a more powerful machine to handling mail, but the load almost always expands to consume whatever resources are allocated.

The `sendmail` programs enable you to limit the load. The goal of load limiting is to prevent wasting time during loaded periods by attempting to deliver large messages, messages to many recipients, and messages to sites that have been down for a long time.

Use the `Ox` and `OX` options to limit the load caused by `sendmail`. The default is not to set any load limits if no options are used. Both of these configuration options take an argument that is an integer load average. For example, if you specify `Ox4` and `OX8`, then the `x` load limiting will be used when the load is above four, and the `X` load limiting will be used when the load is above eight. When the load is above the value specified in the `x` option, the SMTP server does not accept connections from the network (locally originated mail and other mail like UUCP are not affected). The `x` option has a more subtle effect, controlling whether messages are queued for later delivery or delivered

immediately. The general idea is always to deliver “small” messages immediately, and to defer “large” messages for delivery during off-peak periods.

The `oq` option specifies the maximum size of a message that is delivered immediately. The “size” of the message includes not only the number of bytes in the message, but also the penalty for a large number of recipients, and for unsuccessful delivery attempts. The penalty per recipient is option value `y`, by default set to 1000. The penalty per delivery attempt is the option value `z`, by default set to 9000. The size limit also depends on the current load, so that more and more messages are queued as the load goes higher. If the load is one above the `x` threshold, then the limit is halved; if the load is two above the threshold, the limit is divided by three, and so on. Note that this limit also applies to messages being delivered when running the queue.

Log Level

You can adjust the level of logging for `sendmail`. The levels are shown in Table C-4.

Table C-4 Log Level Codes

Code	Description
0	No logging
1	Major problems only
2	Message collections and failed deliveries
3	Successful deliveries
4	Messages being deferred (due to a host being down, for example.)
5	Normal message “queue ups”
6	Unusual but benign incidents; for example, trying to process a locked queue file
9	(Default) Log internal queue ID to external message ID mappings, which can be useful for tracing a message as it travels between several hosts
12	Messages that are of interest when debugging
16	Verbose information regarding the queue
22	Everything

Refer to the system error logging section in *Common Administration Tasks* for more information.

File Modes

Certain files may have protection modes that control access. This section describes the modes that you can control from the `sendmail.cf` file. The modes you use depend on what functionality you want and the level of security you require.

`setuid`

By default, the `sendmail` program is executed with the user ID set to 0 (`setuid` to `root`) so that it can deliver to programs that might write in a user's home directory. When `sendmail` is ready to execute a mailer program, `sendmail` checks to see if the user ID is 0; if so, it resets the user ID and group ID to the values set by the `u` and `g` options in the configuration file. The user ID and group ID are both set to 1 (`daemon`). You can override these values by setting the `S` flag to the mailer (for mailers that are trusted, and must be called as `root`). However, mail processing is accounted to `root` rather than to the user sending the mail.

Temporary File Modes

The `OF` option sets the mode of all the temporary files that `sendmail` uses. The default, `OF0600`. `0600` is appropriate for secure mail, and `0644` is more permissive. If you use the more permissive mode, it is not necessary to run `sendmail` as `root` at all (even when running the queue). Users will be able to read mail in the queue.

Should My Alias Database Be Writable?

One approach is to provide the alias database (`/etc/mail/aliases*`) with mode `666`. If you use this approach, users can modify any list. However, you may want to limit the aliases that a user can change by putting them into a file that the user can edit and referencing this file from `/etc/mail/aliases`. Such references have the following format:

```
alias-name::include: /filename
```

`sendmail` *Configuration File*

This section describes the configuration file in detail, including hints for writing your own file.

The syntax of the configuration file is reasonably easy to parse, since parsing can be done every time `sendmail` starts. Unfortunately, this syntax sacrifices readability.

`sendmail` uses single letters for several different functions:

- Command-line flags
- Configuration options
- Queue file line types
- Configuration file line types
- Mailer field names
- Mailer flags
- Macro names
- Class names

The following sections provide an overview of the configuration file, and details of its semantics.

Purpose of the `sendmail` Configuration File

The `sendmail` configuration file has three parts:

- Definition of symbols, classes, options, and parameters
- Definitions of mailers and delivery programs
- Rule sets that determine the rules for rewriting addresses

You define symbols, classes, options, and parameters to set up the environment for `sendmail`, setting the options and defining a few critical macros.

You define your mailers and delivery programs so that `sendmail` knows which protocols to use and which delivery programs to interact with.

You define rewriting rules, grouped into rule sets, to transform addresses from one form to another. In general, each rule in a rule set is applied to a particular address. An address might be rewritten several times within a rule set.

There are eight standard rule sets, applied in the order shown in Table C-5.

Table C-5 Order of Application of Rule Sets

Rule set	Description
Rule set 3	The first rule set applied. It tries to put the address into the form: <i>local-address@host-domain</i> .
Rule set 0	Determines the destination, and which mailer program to use to send the message. It resolves the destination into a triple (<i>mailer, host, user</i>).
Rule set 1	Rewrites the sender address.
Rule set X	Specifies additional rule sets that enable the sender addresses to do final mailer-specific cleanup. These rule sets have different names for each mailer. S, for example, stands for a generic "sender."
Rule set 2	Rewrites the recipient address.
Rule set R	Specifies additional rule sets that enable the recipient addresses to do final mailer-specific cleanup. These rule sets have different names for each mailer. R, in this example, stands for a generic "sender."
Rule set 4	Rewrites all addresses for the last time, usually from internal to external form.

Rule set 0 must resolve to the internal form, which is in turn used as a pointer to a mailer descriptor. The mailer descriptor describes the interface requirements of the mailer.

Rewriting names in the message is typically done in two phases. The first phase uses rule set 3 to map names in any format into a *local-address@host-domain* form. The second phase converts map names to the canonical form into the syntax appropriate for the receiving mailer. `sendmail` rewrites names in three subphases. Rule sets 1 and 2 are applied to all sender and recipient names, respectively. You may specify mailer-specific rule sets in rule set 3 for both sender and recipient names. Finally, rule set 4 is applied to do any conversion to external form.

RFC 822 describes the format of the mail message itself. `sendmail` follows this RFC closely, to the extent that many of the standards described in this document cannot be changed without changing the code. In particular, the following characters have special interpretations:

< > () " \



Caution – Use the RFC 822 special characters < > () " \ only for their designated purposes. Information between parentheses, (), is reserved for comments or personal names. Information between angle brackets, <>, is reserved for *local-address@host-domain* addresses.

A Sample sendmail Configuration File

The following code example shows the default `main.cf` file. Subsequent sections describe the syntax and semantics used in this file.

Code Example C-1 The Default `main.cf` File (1 of 10)

```
#####
#
#      Sendmail configuration file for "MAIN MACHINES"
#
#      You should install this file as /etc/sendmail.cf
#      if your machine is the main (or only) mail-relaying
#      machine in your domain. Then edit the file to
#      customize it for your network configuration.
#
#      @(#)main.mc 1.17 90/01/04 SMI
#
###      local info

Lm maildomain
# my official hostname
# You have two choices here. If you want the gateway machine to identify
# itself as the DOMAIN, use this line:
Dj$m
# If you want the gateway machine to appear to be INSIDE the domain, use:
#Dj$w.$m
# Unless you are using sendmail.mx (or have a fully-qualified hostname), use:
#Dj$w

# major relay mailer - typical choice is "ddn" if you are on the
# Defense Data Network (e.g. Arpanet or Milnet)
DMsmartuucp

# major relay host: use the $M mailer to send mail to other domains
```

Code Example C-1 The Default main.cf File (Continued) (2 of 10)

```
DRddn-gateway
CRddn-gateway

# If you want to pre-load the "mailhosts" then use a line like
# FS /usr/lib/mailhosts
# and then change all the occurrences of $%y to be $=S instead.
# Otherwise, the default is to use the hosts.byname map if NIS
# is running (or else the /etc/hosts file if no NIS).

# valid top-level domains (default passes ALL unknown domains up)
CT arpa com edu gov mil net org
CT us de fr jp kr nz il uk no au fi nl se ca ch my dk ar

# options that you probably want on a mailhost:

# checkpoint the queue after this many recipients
OC10

# refuse to send tiny messages to more than these recipients
Ob10

#####
#
#       General configuration information

# local domain names
#
# These can now be determined from the domainname system call.
# The first component of the NIS domain name is stripped off unless
# it begins with a dot or a plus sign.
# If your NIS domain is not inside the domain name you would like to have
# appear in your mail headers, add a "Dm" line to define your domain name.
# The Dm value is what is used in outgoing mail. The Cm values are
# accepted in incoming mail. By default Cm is set from Dm, but you might
# want to have more than one Cm line to recognize more than one domain
# name on incoming mail during a transition.
# Example:
# DmCS.Podunk.EDU
# Cm cs cs.Podunk.EDU
#
# known hosts in this domain are obtained from gethostbyname() call
# Version number of configuration file
```


Code Example C-1 The Default main.cf File (Continued) (3 of 10)

```
#ident      "@(#)version.m4      1.14      90/06/14 SMI"      /* SVr4.0 1.1      */

DVSMI-SVR4

###          Standard macros

# name used for error messages
DnMailer-Daemon
# UNIX header format
DlFrom $g $d
# delimiter (operator) characters
Do.:%@!^=/[ ]
# format of a total name
Dq$g$?x ($x)$ .
# SMTP login message
De$j Sendmail $v/$V ready at $b

###          Options

# Remote mode - send through server if mailbox directory is mounted
OR
# location of alias file
OA/etc/mail/aliases
# default delivery mode (deliver in background)
Odbackground
# rebuild the alias file automagically
OD
# temporary file mode -- 0600 for secure mail, 0644 for permissive
OF0600
# default GID
Og1
# location of help file
OH/var/lib/sendmail.hf
# log level
OL9
# default messages to old style
Oo
# Cc my postmaster on error replies I generate
OPPostmaster
# queue directory
```

Code Example C-1 The Default main.cf File (Continued) (4 of 10)

```
OQ/var/spool/mqueue
# read timeout for SMTP protocols
Or15m
# status file -- none
OS/etc/mail/sendmail.st
# queue up everything before starting transmission, for safety
Os
# return queued mail after this long
OT3d
# default UID
Oul

###      Message precedences
Pfirst-class=0
Pspecial-delivery=100
Pjunk=-100

###      Trusted users
T root daemon uucp

###      Format of headers
H?P?Return-Path: <$g>
HReceived: $?sfrom $s $.by $j ($v/$V)
          id $i; $b
H?D?Resent-Date: $a
H?D?Date: $a
H?F?Resent-From: $q
H?F?From: $q
H?x?Full-Name: $x
HSubject:
H?M?Resent-Message-Id: <$t.$i@$j>
H?M?Message-Id: <$t.$i@$j>
HErrors-To:

#####

###      Rewriting Rules   ###
#####

#
```

Code Example C-1 The Default main.cf File (Continued) (5 of 10)

```

Sender Field Pre-rewriting
S1
# None needed.

#
Recipient Field Pre-rewriting
S2
# None needed.

# Name Canonicalization

# Internal format of names within the rewriting rules is:
#   anything<@host.domain.domain...>anything
# We try to get every kind of name into this format, except for local
# names, which have no host part. The reason for the "<>" stuff is
# that the relevant host name could be on the front of the name (for
# source routing), or on the back (normal form). We enclose the one that
# we want to route on in the <>'s to make it easy to find.
#
S3

# handle "from:<>" special case
R$*<>$*                $$$                turn into magic token

# basic textual canonicalization
R$*<$+>$*                $2                basic RFC822 parsing

# make sure <a,@b,@c:user@d> syntax is easy to parse -- undone later
R@$+,$+:$+                @$1:$2:$3                change all ",", to ":"
R@$+:$+                $$>6<@$1>:$2                src route canonical
R$+:$*;$+                $$1:$2;@$3                list syntax
R$+@$+                $:$1<@$2>                focus on domain
R$+<$+@$+>                $1$2<@$3>                move gaze right
R$+<@$+>                $$>6$1<@$2>                already canonical

# convert old-style names to domain-based names
# All old-style names parse from left to right, without precedence.
R$-!$+                $$>6$2<@$1.uucp>    uucphost!user
R$-.$+!$+                $$>6$3<@$1.$2>    host.domain!user
R$+%$+                $$>3$1@$2        user%host

```

Code Example C-1 The Default main.cf File (Continued) (6 of 10)

```

# Final Output Post-rewriting
S4
R$+<@$.uucp>          $2!$1          u@h.uucp => h!u
R$+                   $: $>9 $1      Clean up addr
R$*<$+>$*             $1$2$3          defocus

# Clean up an name for passing to a mailer
# (but leave it focused)
S9
R$=w!@                @$w!$n
R@                     @$n          handle <> error addr
R$*<$*LOCAL>$*        $1<$2$m>$3      change local info
R<@$+>$*:$+:$+$+     <@$1>$2,$3:$4    <route-addr> canonical

#####
#   Rewriting rules

# special local conversions
S6
R$*<@$*$=m>$*         $1<@$2LOCAL>$4      convert local domain

# Local and Program Mailer specification

Mlocal,      P=/bin/mail, F=flsSDFMmP, S=10, R=20, A=mail -d $u
Mprog,       P=/bin/sh, F=lsDFMeuP, S=10, R=20, A=sh -c $u

S10
# None needed.

S20
# None needed.

#ident      "(#)etherm.m4      1.12      90/06/14 SMI"      /* SVr4.0 1.1      */

#
#####
#####

#####      Ethernet Mailer specification
#####

```

Code Example C-1 The Default main.cf File (Continued) (7 of 10)

```

#####      Messages processed by this configuration are assumed to remain
#####      in the same domain. This really has nothing particular to do
#####      with Ethernet - the name is historical.

Mether,      P=[TCP], F=msDFMuCX, S=11, R=21, A=TCP $h
S11
R$*<@$+>$*      @$!<@$2>$3      already ok
R$+              @$!<@$w>      tack on our hostname

S21
# None needed.

#####
# General code to convert back to old style UUCP names
S5
R$+<@LOCAL>      @$ $w!$1      name@LOCAL => sun!name
R$+<@$-.LOCAL>   @$ $2!$1      u@h.LOCAL => h!u
R$+<@$+.uucp>    @$ $2!$1      u@h.uucp => h!u
R$+<@$*>         @$ $2!$1      u@h => h!u
# Route-addr's do not work here. Punt til uucp-mail comes up with something.
R<@$+>$*         @$ @$1$2      just defocus and punt
R$*<$*>$*        @$ $1$2$3     Defocus strange stuff

#          UUCP Mailer specification

Muucp,      P=/usr/bin/uux, F=msDFMhuU, S=13, R=23,
            A=uux - -r -a$f $h!rmail ($u)

# Convert uucp sender (From) field
S13
R$+          $:$>5$1      convert to old style
R$=w!$+      $2          strip local name
R$+          $:$w!$1      stick on real host name

# Convert uucp recipient (To, Cc) fields
S23
R$+          $:$>5$1      convert to old style

#ident      "@(#)ddnm.m4      1.5      90/06/14 SMI"      /* SVr4.0 1.1      */

```

Code Example C-1 The Default main.cf File (Continued) (8 of 10)

```
#####
#
#           DDN Mailer specification
#
#           Send mail on the Defense Data Network
#           (such as Arpanet or Milnet)

Mddn,      P=[TCP], F=msDFMuCX, S=22, R=22, A=TCP $h, E=\r\n

# map containing the inverse of mail.aliases
DZmail.byaddr

S22

R$*<@LOCAL>$*           $:$1
R$-<@$->                 $:$>3${Z$1@$2$}           invert aliases
R$*<@$+.$*>$*           @$1<@$2.$3>$4           already ok
R$+<@$+>$*              @$1<@$2.$m>$3           tack on our domain
R$+                     @$1<@$m>               tack on our domain

# "Smart" UUCP mailer: Uses UUCP transport but domain-style naming
Msmartuucp, P=/usr/bin/uux, F=CmsDFMhuU, S=22, R=22,
            A=uux - -r $h!rmail ($u)

#####
#
#           RULESET ZERO
#
#           This is the ruleset that determines which mailer a name goes to.

# Ruleset 30 just calls rulesets 3 then 0.
S30
R$*           $: $>3 $1           First canonicalize
R$*           @$ $>0 $1           Then rerun ruleset 0

S0
# On entry, the address has been canonicalized and focused by ruleset 3.
# Handle special cases.....
R@           $#local $:$n           handle <> form
```

Code Example C-1 The Default main.cf File (Continued) (9 of 10)

```

# resolve the local hostname to "LOCAL".
R$*<$*$=w.LOCAL>$*      $1<$2LOCAL>$4      thishost.LOCAL
R$*<$*$=w.uucp>$*      $1<$2LOCAL>$4      thishost.uucp
R$*<$*$=w>$*          $1<$2LOCAL>$4      thishost

# Mail addressed explicitly to the domain gateway (us)
R$*<@LOCAL>          $@$>30$1      strip our name, retry
R<@LOCAL>:$+        $@$>30$1      retry after route strip

# For numeric spec, you can't pass spec on to receiver, since old rcvr's
# are not smart enough to know that [x.y.z.a] is their own name.
R<@[+]>:$*          $:$>9 <@[1]>:$2      Clean it up, then...
R<@[+]>:$*          $#ether $@[1] $:$2      numeric internet spec
R<@[+]>,$*          $#ether $@[1] $:$2      numeric internet spec
R$*<@[+]>          $#ether $@[2] $:$1      numeric internet spec

# deliver to known ethernet hosts explicitly specified in our domain
R$*<@$%y.LOCAL>$*    $#ether $@$2 $:$1<@$2>$3      user@host.sun.com

# etherhost.uucp is treated as etherhost.$m for now.
# This allows them to be addressed from uucp as foo!sun!etherhost!user.
R$*<@$%y.uucp>$*    $#ether $@$2 $:$1<@$2>$3      user@etherhost.uucp

# Explicitly specified names in our domain -- that we've never heard of
R$*<@$*.LOCAL>$*    $#error $:Never heard of host $2 in domain $m

# Clean up addresses for external use -- kills LOCAL, route-addr ,=>:
R$*                  $:$>9 $1      Then continue...

# resolve UUCP-style names
R<@$-.uucp>:$+      $#uucp          $@$1 $:$2          @host.uucp:...
R$+<@$-.uucp>      $#uucp          $@$2 $:$1          user@host.uucp

# Pass other valid names up the ladder to our forwarder
#R$*<@$*.$=T>$*    $#M          $@$R $:$1<@$2.$3>$4
user@domain.known

# Replace following with above to only forward "known" top-level domains
R$*<@$*.$+>$*      $#M          $@$R $:$1<@$2.$3>$4      user@any.domain

# if you are on the DDN, then comment-out both of the lines above
# and use the following instead:

```

Code Example C-1 The Default main.cf File (Continued) (10 of 10)

```
#R$*<@$*.$+>$*           $#ddn $@ $2.$3 $:$1<@$2.$3>$4           user@any.domain

# All addresses in the rules ABOVE are absolute (fully qualified domains).
# Addresses BELOW can be partially qualified.

# deliver to known ethernet hosts
R$*<@$%y>$*               $#ether $@$2 $:$1<@$2>$3           user@etherhost

# other non-local names have nowhere to go; return them to sender.
R$*<@$+.$->$*             $#error $:Unknown domain $3
R$*<@$+>$*               $#error $:Never heard of $2 in domain $m
R$*@$*                   $#error $:I don't understand $1@$2

# Local names with % are really not local!
R$+%%$+                  $@$>30$1@$2           turn % => @, retry

# everything else is a local name
R$+                      $#local $:$1           local names
```

sendmail *Configuration File Syntax*

The configuration file is organized as a series of lines, each of which begins with a single character defining the semantics for the rest of the line. Lines beginning with a space or a Tab are continuation lines (although the semantics are not well defined in many places). Blank lines and lines beginning with a pound sign (#) are comments.

D and L — Define Macro

Macros are named with a single character. Although a macro can be defined with any character from the complete ASCII set, use only uppercase letters for macros that you define. However, do not use characters like M, R, L, G and V that are already used by sendmail. Lowercase letters and special symbols are used internally.

There are two ways you can define macros:

- D assigns the value directly specified.

- `L` assigns the value looked up in the `sendmailvars` database (either the NIS+ table or `/etc/mail/sendmailvars` file). The `L` command is classified as a Sun uncommitted interface.

The syntax for `D` macro definitions is:

```
D $X$ val
```

where X is the name of the macro and *val* is the value it should have. No spaces are allowed. Macros can be inserted in most places using the escape sequence `$X`.

Here are examples of `D` macro definitions from the configuration file:

```
DRmailhost  
DmEng.Acme.COM
```

The variable `R` is set to contain the value `mailhost` and the internal variable `m` is set to contain the value `Eng.Acme.COM`.

The `m` macro defines the mail domain. If it is not defined, the naming service domain name is used with the first component stripped off. For example, `Ecd.East.Acme.COM` becomes `East.Acme.COM`. An even more flexible way to define the mail domain name is to use an `L` macro definition, as show below.

The syntax for an `L` macro definition is:

```
L $X$ search_key
```

where X is the name of the macro and *search_key* is looked up in the `sendmailvars` database. The value found in the entry located by the search key is assigned to X .

Here is an example of an internal `L` macro definition from the configuration file:

```
Lmmaildomain
```

The variable `m` is set to the value found in the `sendmailvars` database using `maildomain` as the search key. If the entry in the `sendmailvars` database appears as follows:

<code>maildomain</code>	<code>Eng.Acme.COM</code>
-------------------------	---------------------------

The value of `m` becomes `Eng.Acme.COM`.

Note - `sendmail` uses the `sendmailvars` entry in the `/etc/nsswitch.conf` file to determine the order in which to search the NIS+ and `/etc/mail/sendmailvars` tables.

C, F, and G—Define Classes

You can define classes of words to match the left-hand side of address rewriting rules. For example, you might create a class of all local names for this site so that you can eliminate attempts to send mail to yourself.

You can give classes names from the set of uppercase letters. Lowercase letters and special characters are reserved for system use.

There are three ways to define classes:

- `C` assigns the values directly specified.
- `F` reads in the values from another file or from another command.
- `G` assigns the values looked up in the `sendmailvars` database (either the NIS+ table or `/etc/mail/sendmailvars` file). The `G` command is classified as a Sun uncommitted interface.

The syntaxes of the different forms of class definition are:

`CX word1 word2`

`FX file [pattern]`

`FX | command`

`GXsearch_key`

The first form defines the class `X` to match any of the named words. The second form reads words from the file into the class `X`, for example, `FC /.rhosts`. The `pattern` argument to the `FC` class is used with `scanf` to read

from the file; otherwise, the first word from each line is used. The third form executes the given command and reads the elements of the class from standard output of the command. For example:

```
FC | awk '{print $2}' /etc/hosts
```

The fourth form reads the elements of the class from the entry in the `sendmailvars` database pointed to by the search key. For example,

```
GVuucp-list
```

gets the definition of class `V` from the `uucp-list` entry in the `sendmailvars` database.

If the entry in the `sendmailvars` database appears as follows:

```
uucp-list      sunmoon hugo comic
```

the value of `V` becomes `sunmoon hugo comic`.

Note - `sendmail` uses the `sendmailvars` entry in the `/etc/nsswitch.conf` file to determine the order in which to search the `NIS+` and `/etc/mail sendmailvars` tables.

You could split class definitions among multiple lines. For example, the following:

```
CHmonet ucbmonet
```

is equivalent to:

```
CHmonet
CHucbmonet
```

O— *Set Option*

You can set several options (not to be confused with mailer flags or command-line arguments) from a configuration file. Options are also represented by single characters. The syntax of this line is:

```
Oc value
```

This sets option *c* to *value*. Depending on the option, *value* may be a string, an integer, a Boolean (with legal values `t`, `T`, `f`, or `F`—the default is “true”), or a time interval. See the section “sendmail Configuration Options” on page 410 for the list of options.

P— *Precedence Definitions*

You can define values for the `Precedence:` field using the `P` control line. The syntax of this field is:

```
Pname=num
```

When the *name* is found in a `Precedence:` field, the message class is set to *num*. Higher numbers mean higher precedence. Numbers less than zero have the special property that error messages are not returned. The default precedence is 0 (zero). For example:

```
Pfirst-class=0  
Pspecial-delivery=100  
Pjunk=-100
```

T— *Define Trusted Users*

Trusted users are those who are permitted to override the sender name by using the `-f` flag. These typically are `root`, `uucp`, `daemon`, and `network`. For some sites it may be convenient to extend this list to include other users, perhaps to support a separate UUCP login for each host. The syntax of this line is:

```
T user ...
```

You can use more than one line to define trusted users.

H— *Define Header*

The format of the header lines is defined by the `H` line. The syntax of this line is:

```
H[c ?c mflagsc ?]c hnamec : c htemplate
```

Continuation lines in this specification are inserted directly into the outgoing message. The *htemplate* is macro-expanded before it is inserted into the message. If the expansion is empty, the header line is not included. If the *mflags* (surrounded by question marks) are specified, at least one of the specified flags must be stated in the mailer definition before this header can be automatically output. If one of these headers is in the input, it is directed to the output regardless of these flags.

Special Header Lines

Several header lines have special interpretations defined by the configuration file. Others have interpretations built into `sendmail` that cannot be changed without changing the code. These built-in features are described in the following list:

- Return-Receipt-To:

If this header is sent, a message will be sent to any specified names when the delivery is complete. The mailer must have the `l` flag (local delivery) set in the mailer descriptor.

- Errors-To:

If errors occur anywhere during processing, this header sends error messages to the listed names rather than to the sender. Use this header line for mailing lists.

- To:

If a message comes in with no recipients listed in the message (in a `To:`, `Cc:`, or `Bcc:` line) then `sendmail` adds an `Apparently To:` header line for each recipient specified on the `sendmail` command line.

R and S—*Rewriting Rules*

Address parsing is done according to the rewriting rules, which is a simple pattern-matching and replacement system. `sendmail` scans through the set of rewriting rules looking for a match on the left-hand side (LHS) of the rule. When a rule matches, the name is replaced by the right-hand side (RHS) of the rule.

There are several sets of rewriting rules. Some of the rewriting sets are used internally and must have specific semantics. Other rewriting sets do not have specifically assigned semantics, and may be referenced by the mailer definitions or by other rewriting sets.

For example:

```
Sc n
```

sets the current rule set being collected to *n*. If you begin a rule set more than once, the new definition overwrites the old definition.

R is used to define a rule in the rule set. The syntax of the R line is:

```
Rlhs rhs comments
```

The *lhs* is a pattern that is applied to the input. If it matches, the input is rewritten to the *rhs*. The *comments* are ignored.

Here is an example of how a rule definition might look:

```
# handle "from:<>" special case
R<>      $@@          turn into magic token
```

The fields must be separated by at least one tab character; you may use embedded spaces in the fields.

M — *Define Mailer*

Programs and interfaces to mailers are defined on this line. The format is:

```
Mc name, c
{c field=value}*
```

where *name* is the name of the mailer (used in error messages) and the *field=value* pairs define attributes of the mailer. The fields are shown in Table C-6.

Table C-6 Mailer Definition Fields

Field	Description
P	The path name of the mailer
F	Special flags for this mailer
S	A rewriting rule set for sender names
R	A rewriting rule set for recipient names
A	An argument vector to pass to this mailer
E	The end-of-line string for this mailer
M	The maximum message length to this mailer
L	The maximum length of the <code>argv</code> for this mailer

Address Rewriting Rules

This section describes the details of rewriting rules and mailer descriptions.

Special Macros, Conditionals

Special macros are referenced with the construct `$x`, where `x` is the name of the macro to be matched (LHS) or inserted (RHS). Lowercase letters are reserved for special semantics, and some special characters are reserved to provide conditionals.

The macros shown in Table C-7 *must* be defined to transmit information into `sendmail`.

Table C-7 Required `sendmail` Macros

Macro	Description
<code>e</code>	The SMTP entry message
<code>\$e</code>	Printed out when SMTP starts
<code>j</code>	The official domain name for this site
<code>\$j</code>	Should be the first word of the <code>\$e</code> macro; <code>\$j</code> should be in domain name format
<code>l</code>	The format of the UNIX <code>From</code> line
<code>n</code>	The name of the daemon (for error messages)
<code>o</code>	The set of “separators” in names
<code>\$o</code>	List of characters that are considered tokens
<code>q</code>	Default format of sender names
<code>\$q</code>	Specifies how a sender should appear in a message when it is created

For example:

```

De$j Sendmail $v ready at $b*
DnMAILER-DAEMON
DlFrom $g $d
Do.:%@!^=/
Dq$g$?x ($x)$.$†
Dj$H.$D†

```


You should not need to change any of these macros except under unusual circumstances. For example, you might want to change the banner (*) for security. You might want to change † or ‡ to make several hosts look like one host.

An acceptable alternative for the \$q macro is:

```
$?x$x $ .<$g>
```

These correspond to the following two formats:

```
doe@Acme.COM (John Doe)
John Doe <doe@Acme.COM>
```

Some macros are defined by `sendmail` for use in mailer arguments or for other contexts. These macros are shown in Table C-8.

Table C-8 Additional `sendmail` Macro Definitions

Macro	Description
a	Origination date in ARPANET format
b	Current date in ARPANET format
c	Hop count
d	Date in UNIX (<code>ctime</code>) format
f	Sender from name
g	Sender name relative to the recipient
h	Recipient host
i	Queue ID
m	Domain name
p	<code>sendmail</code> process ID
r	Protocol used
s	Sender's host name
t	Numeric representation of the current time
u	Recipient user

Table C-8 Additional `sendmail` Macro Definitions (Continued)

Macro	Description
<code>v</code>	Version number of <code>sendmail</code>
<code>w</code>	Host name of this site
<code>x</code>	Full name of the sender
<code>z</code>	Home directory of the recipient

You can use three types of dates. The `$a` and `$b` macros are in ARPANET format; `$a` is the time as extracted from the `Date:` line of the message (if there was one), and `$b` is the current date and time (used for postmarks). If no `Date:` line is found in the incoming message, `$a` is also set to the current time. The `$d` macro is equivalent to the `$a` macro in UNIX (`ctime`) format.

The `$f` macro is the ID of the sender as *originally determined*; for a message mailed to a specific host, the `$g` macro is set to the name of the sender *relative to the recipient*. For example, suppose the sender `eric` sends a message to `bollard@matisse` from the machine `ucharpa`. The value of `$f` will be `eric` and the value of `$g` will be `eric@ucharpa`.

The `$x` macro is set to the full name of the sender, which can be determined in several ways. It can be passed as a flag to `sendmail` (from the value of the `Full-name:` line in the header or use the comment field of a `From:` line). If the name can't be determined from the `Full-name` or `From` lines, and if the message is being originated locally, the full name is looked up in the `/etc/passwd` file. It can also be read from the `name` environment variable.

When a message is sent, the `$h`, `$u`, and `$z` macros get set to the host, user, and home directory (if local) of the recipient. The first two are set from the `$@` and `$:` part of the rewriting rules, respectively.

The `$p` and `$t` macros are used to create unique strings (for example, for the `Message-Id:` field). The `$i` macro is set to the queue ID on this host; if put into the time stamp line, it can be useful for tracking messages. The `$v` macro is set to be the version number of `sendmail`; this is normally put in time stamps and is extremely useful for debugging. It can, however, be a security risk. The `$w` macro is set to the primary name of this host, as given by `gethostname(3B)` and `gethostbyname(3N)`. The `$c` field is set to the "hop count"; that is, the number of times this message has been processed, which can be determined by counting the time stamps in the message.

The `$r` and `$s` fields are set to the protocol used to communicate with `sendmail` and the sending host name.

You can specify conditionals by using the syntax:

```
$?x text1 $ | text2 $
```

This inserts `text1` if the macro `$x` is set, and `text2` otherwise. The `else (c $ |)` clause may be omitted.

Special Classes

The class `$=w` is the set of all names by which this host is known. It can be used to delete local host names. The class `$=m` is set to the domain name.

Left-Hand Side of Address Rewriting Rules

The left-hand side of rewriting rules contains a pattern. Normal words are matched directly. Dollar signs introduce “metasymbols,” which match things other than simple words, like macros or classes. The metasymbols are shown in Table C-9.

Table C-9 sendmail Left-Hand Side Metasymbols

Symbol	Matches
<code>\$*</code>	Zero or more tokens.
<code>\$+</code>	One or more tokens.
<code>\$-</code>	Exactly one token.
<code>\$=x</code>	Any string in class <code>x</code> .
<code>\$~x</code>	Any token not in class <code>x</code> .
<code>\$%x</code>	Any token in NIS map or NIS+ table <code>\$x</code> .
<code>\$!x</code>	Any token not in NIS map or NIS+ table <code>\$x</code> .
<code>\$x</code>	Macro <code>x</code> .

If any of the patterns matches, it is assigned to the symbol `$c n` for replacement on the right-hand side, where *n* is the index in the LHS. For example, the LHS rules can be applied to this input:

```
$-:$+
```

```
JUPITER:eric
```

The rule will match, and the values passed to the RHS will be:

```
$1 JUPITER
$2 eric
```

The `$$x` uses the macro *x* to specify the name of an NIS map. The special form `$$y` matches any host name in the `hosts.byname` map, or in `/etc/hosts` if not running NIS.

Right-Hand Side of Address Rewriting Rules

When the left-hand side of a rewriting rule matches, the input is replaced by the right-hand side. Tokens are copied directly from the right-hand side, unless they begin with a dollar sign, in which case they are treated as macros and expanded.

Metasymbols for more complicated substitutions are shown in Table C-10.

Table C-10 sendmail Right-Hand Side Metasymbols

Symbol	Description
<code>\$\$x</code>	Expand macro <i>x</i> .
<code>\$\$n</code>	Substitute indefinite token <i>n</i> from LHS. <i>n</i> is a digit.
<code>\$\$>n</code>	Call rule set <i>n</i> . <i>n</i> is a digit.
<code>\$\$#mailer</code>	Resolve to <i>mailer</i> .
<code>\$\$@host</code>	Specify <i>host</i> .
<code>\$\$:user</code>	Specify <i>user</i> .

Table C-10 sendmail Right-Hand Side Metasymbols (Continued)

Symbol	Description
<code>\$(host\$)</code>	Map to primary host name.
<code>\$(x name\$)</code>	Map name through NIS map or NIS+ table <code>\$x</code> . If the map name begins with <code>rev</code> , <code>sendmail</code> will reverse the aliases.

The `$n` (n being a digit) syntax substitutes the corresponding value from a `$+`, `$-`, `$*`, or `$=@` match on the LHS. It may be used anywhere.

The `$> n` syntax substitutes the remainder of the line as usual and then passes it to rule set n . The final value of rule set n then becomes the substitution for this rule (like a procedure or function call).

Only use the `$#` syntax in rule set 0. Evaluation of the rule set stops immediately, and signals are sent to `sendmail` that the name has completely resolved. The complete syntax is:

```
$#mailer$@host$: user
```

This specifies the `{mailer, host, user}` triple necessary to direct the mailer. More processing may then take place, depending on the mailer. For example, local names are aliased.

A right-hand side may also be preceded by a `$@` or a `$:` to control evaluation. A `$@` prefix returns the remainder of the right-hand side as the value. A `$:` prefix terminates the rule immediately, but the rule set continues. Thus it can be used to limit a rule to one application. Neither prefix affects the result of the right-hand side expansion.

The `$@` and `$:` prefixes can precede a `$>` spec. For example:

```
R$+ $:$>7$1
```

matches anything, passes that to rule set seven, and continues; the `$:` is necessary to avoid an infinite loop. The `$(host$)` syntax replaces the host name with the “official” or primary host name, the one listed first in the

hosts.byname NIS map, or */etc/hosts* if not running NIS. It is used to eliminate nicknames for hosts. The `#{x name $}` syntax replaces the string by the result of the `nis_map_name` indicated in macro `$(x)`.

Semantics of Rewriting Rule Sets

Five rewriting sets have specific semantics, as shown in Figure C-2.

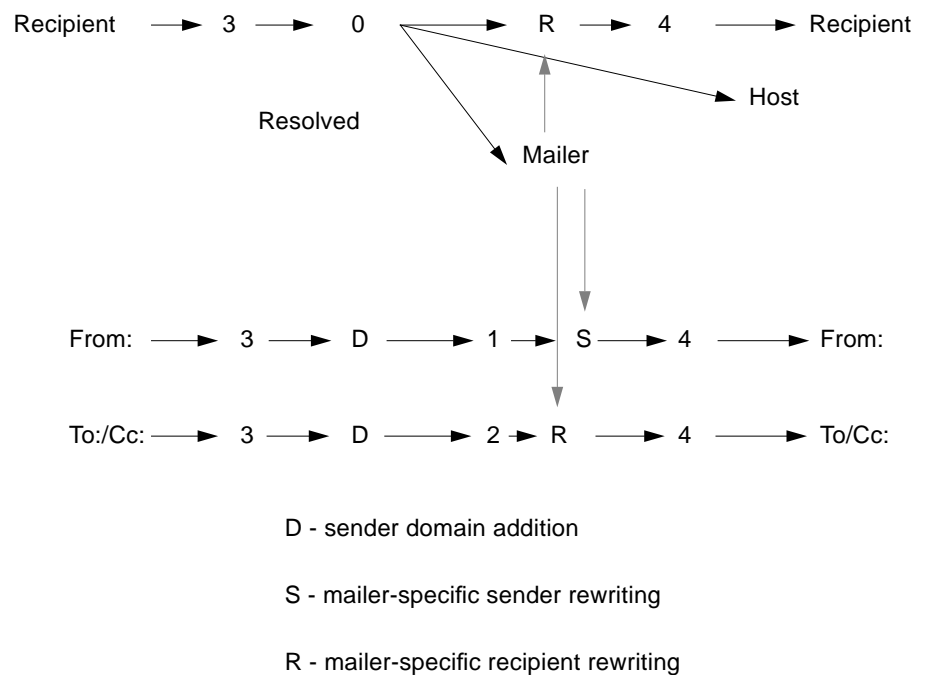


Figure C-2 Rewriting Set Semantics

Rule set 3 is applied by `sendmail` before `sendmail` does anything with any name. Rule set 3 should turn the name into a form, with the basic syntax:

local-part@host-domain-spec

If no @ sign is specified, then the *host-domain-spec* may be appended from the sender name (if the C flag is set in the mailer definition corresponding to the *sending* mailer).

Rule set 0 is applied after rule set 3 to names that are actually going to specify recipients. It must resolve to a *mailer*, *host*, *user* triple. The *mailer* must be defined in the mailer definitions from the configuration file. The *host* is defined into the \$h macro for use in the argument expansion of the specified mailer; the *user* is defined into \$u.

Rule set 1 is applied to all FROM: recipient names, and rule set 2 is applied to all TO: and CC: lines. Then the rule sets specified in the mailer definition line (and R=) are applied. This process is done many times for one message, depending on how many mailers the message is routed to by rule set 0.

Rule set 4 is applied last to all names in the message. It is typically used to translate internal to external form.

error *Mailer*

You can use the mailer with the special name `error` in rule set 0 to generate a user error message. The user field is a message to be printed. For example, the entry:

```
$#error$:Host unknown in this domain
```

on the RHS of a rule generates the specified error, if the LHS matches.

Semantics of Mailer Descriptions

Each mailer has an internal name. It can be arbitrary, except that the names `local` and `prog` must be defined first and second, respectively. Rule set 0 resolves names to this mailer name (and a host and user name).

Give the path name of the mailer in the P field. If this mailer will be accessed by way of a TCP connection, use the string `[TCP]` instead.

Define the mailer flags in the F field. Specify an `f` or `r` flag to pass the name of the sender as a `-f` or `-r` flag respectively. These flags are only passed if they were passed to `sendmail`, so that mailers that give errors under some circumstances can be placated. In some cases, you may be able to specify `-f$g`

in the `argv` template. If the mailer must be called as `root`, and `sendmail` is running `setuid` to `root`, use the `S` flag; it will not reset the user ID before calling the mailer. If this mailer is local (that is, will perform final delivery rather than another network hop), use the flag. Quoted characters (backslashes and double quotation marks) can be stripped from names if the `s` flag is specified; if it is not specified, they are passed through. If the mailer is capable of sending to more than one user on the same host in a single transaction, use the `m` flag. If this flag is on, then the `argv` template containing `$u` is repeated for each unique user on a given host. The `e` flag marks the mailer as being “expensive,” and `sendmail` defers connection until a queue run. Note that the `c` configuration option must also be set.

The `C` flag is a useful case. It applies to the mailer from which the message is sent, rather than the mailer where the message is received. If set, the domain specification of the sender (that is, the `@host.domain` part) is saved and is appended to any names in the message that do not already contain a domain specification. For example, a message in this form:

```
From: eric@jupiter
To: joe@saturn, sam
```

is modified to:

```
From: eric@jupiter
To: joe@saturn, sam@ganymede
```

if and only if the `C` flag is defined in the mailer corresponding to `eric@jupiter`.

The `S` and `R` fields in the mailer description are per-mailer rewriting sets to be applied to sender and recipient names, respectively. These are applied after the sending domain is appended and the general rewriting sets (number one or two) are applied, but before the output rewrite (rule set four) is applied. A typical use is to append the current domain to names that do not already have a domain. For example, a header in this form:

```
From: eric@host
```


might be changed to:

```
From: eric@host.Podunk.EDU
```

or

```
From: saturn!eric
```

depending on the domain into which it is being shipped. These sets can also be used to do special-purpose output rewriting in cooperation with rule set 4.

Table C-11

Field Name	Used To...
E	Define the string to use as an end-of-line indication. A string containing <code>return</code> and <code>newline</code> is the default (if using TCP, otherwise just a <code>newline</code> indicates end-of-line. You can use the <code>print</code> backslash escapes (<code>/r</code> , <code>/n</code> <code>/f</code> , <code>/b</code>).
A	Specify an argument vector template. It may have embedded spaces. The template is macro-expanded before being passed to the mailer. Useful macros include <code>\$h</code> , the host name resolved by rule set 0, and <code>\$u</code> , the user name (or names) resolved. If there is no argument with a <code>\$u</code> macro in it, <code>sendmail</code> uses SMTP to communicate with the mailer. If the path name for this mailer is TCP, use the argument vector: <code>TCP \$h [port]</code> , where <code>port</code> is the optional port number to connect to.
L	Specify the maximum length of the <code>\$u</code> macro passed to the mailer. To make UUCP mail more efficient, the <code>L</code> field can be used with the <code>m</code> flag to send mail to multiple recipients with one call to the mailer, while avoiding mailer limitations on argument length. <code>\$u</code> always expands to at least one recipient even if that recipient exceeds the <code>L=</code> limit.

For example, the specification:

```
Mlocal, P=/bin/mail, F=flsSDFMmnP, S=10, R=20, A=mail -d $u
Mprog, P=/bin/sh, F=lsDFMeuP, S=10, R=20, A=sh -c $u
```

specifies a mailer for local delivery and a mailer for Ethernet delivery. The first is called `local`, is located in the file `/bin/mail`, takes a `-F` flag, does local delivery, strips quotes from names, and delivers mail to multiple users at once. It applies rule set 10 to sender names in the message and applies rule set 20 to recipient names. The argument vector to send to a message is the word `mail`, the word `-d`, and words containing the name of the receiving user. If a `-r` flag is inserted, it is between the words `mail` and `-d`.

The second mailer is called `ether`. It is connected via TCP, and can handle multiple users at once. It defers connections, and appends any domain from the sender name to any receiver name without a domain; processes sender names by rule set 11 and recipient names by rule set 21. Messages passed through this mailer have a 100,000-byte limit.

Building a Configuration File

Building a configuration file “from scratch” is a complex task. Fortunately, it is almost never necessary; you can accommodate almost every situation by changing an existing file. In any case, it is critical that you understand what it is that you are trying to do, and to come up with a policy statement for the delivery of mail. This section explains the purpose of a configuration file and gives you some ideas about policies.

Domains and Policies

RFC 1006 describes domain-based naming. RFC 822 touches on this issue as well. Essentially each host is given a name that is a right-to-left dot-qualified pseudo-path from a distinguished root. The elements of the path are organizational entities, not physical networks.

RFC 822 and RFC 976 specify how certain sorts of addresses should be parsed. You can configure `sendmail` to follow or ignore these rules.

How to Proceed

After you have established a policy, examine the available configuration files to decide if you can use major parts of any of them. Even under the worst of conditions, there is a fair amount of boilerplate information that can be collected safely.

The next step is to build rule set 3, which specifies a rule set for your individual mailers. Building rule set 3 is the hardest part of the job. Here are some guidelines:

- Beware of doing too much to the name in this rule set, since anything you do will be reflected in the message.
- Do not strip local domains in this rule set. Doing so can leave you with names that have no domain specifications at all. `sendmail` appends the sending domain to names with no domain, which can change the semantics of names.
- Do not provide fully qualified domains in this rule set. Although technically correct, fully qualified domain names can lead to unnecessarily long names reflected into messages. The SunOS configuration files define rule set 9 to qualify domain names and strip local domains. Rule set 9 is called from rule set 0 to get all names into a cleaner form.

After you have rule set 3 finished, the other rule sets should be relatively simple. If you need hints, examine the supplied configuration files.

Testing the Rewriting Rules — the `-bt` Flag

When you build a configuration file, you can do a certain amount of testing using the test mode of `sendmail`. For example, you could invoke `sendmail` as:

```
% sendmail -bt -Ctest.cf
```

which would read the configuration file `test.cf` and enter test mode. For example:

```
ADDRESS TEST MODE
Enter <ruleset> <name>
>
```

In this mode, you enter lines in this form:

```
ADDRESS TEST MODE
Enter <ruleset> <name>
> rwset name
```

where *rwset* is the rewriting set you want to use and *name* is a name to which to apply the set. Test mode shows you the steps it takes as it proceeds, and shows you the final name. You may use a comma-separated list of *rwsets* for sequential application of rules to an input; rule set three is always applied first. For example:

```
ADDRESS TEST MODE
Enter <ruleset> <name>
> 1,21,4 jupiter:smith
```

First apply rule set 3 to the input `monet:bollard`. Rule set 1 is then applied to the output of rule set 3, followed similarly by rule sets 21 and 24.

If you need more detail, you can also use the `-d21` flag to turn on more debugging. For example:

```
% sendmail -bt -d21.99
```

turns on large amount of information; a single word name may result in several pages worth of information.

How sendmail Interacts With a Name Service

Mail domain is a concept used by the standard `sendmail.cf` file to determine whether mail should be delivered directly or via mailhost. Intra-domain mail is delivered via direct SMTP connection, while inter-domain mail is forwarded to a mailhost.

In a secure network, only a few selected hosts are authorized to generate packets targeted to external destinations. Even if a host has the IP address of the remote host external to the mail domain, this does not guarantee that an SMTP connection can be established. The standard `sendmail.cf` assumes the following:

- The current host is not authorized to send packets directly to a host outside the mail domain.
- Mailhost is capable of forwarding the mail to an authorized host that can transmit packets directly to an external host. (In fact, mailhost may itself be an authorized host.)

Given these assumptions, it is the responsibility of mailhost to deliver or forward inter-domain mail.

How to Set Up `sendmail` Requirements for Name Services

`sendmail` imposes various requirements on a name service. The following section explains these requirements and how to satisfy them. For more information, refer to the `in.named(1M)`, `nis+(1)`, `nisaddent(1M)`, and `nsswitch.conf(4)` man pages.

1. The mail domain name must be a suffix of the name service domain. For example, if the domain name of name service is "A.B.C.D," then the mail domain name could be one of the following:
 - A.B.C.D
 - B.C.D
 - C.D
 - D

When first established, the mail domain name is identical to the name service domain. As the network grows larger, the name service domain is divided into more manageable pieces. However, the mail domain remains undivided to avoid large aliases and longer mail addresses.

- *If you are setting NIS as the primary name service*, `sendmail` automatically strips off the first component of the NIS domain name and uses the result as mail domain name. For example, `EBS.Admin.Acme.COM` becomes `Admin.Acme.COM`. No special command is needed.

- *If you are setting up NIS+ as your Primary name service*, a hierarchical name space is required by NIS+. `sendmail` can look up the mail domain from the NIS+ `sendmailvars` table, a two column NIS+ table with one key column and one value column. To setup your mail domain, you must add one entry to this table. This entry should have the key column set to the literal string "maildomain" and the value column set to the your mail domain name (for example, "Admin.Acme.COM"). Although NIS+ allows any string in the `sendmailvars` table, the suffix rule still applies for the mail system to work correctly. You can use `nistbladm` to add the `maildomain` entry to the `sendmailvars` table. For example, `nistbleadm -A \`
`key="maildomain" value=<mail domain> \`
`sendmailvars.org_dir.<NIS+ domain>`. Note that this mail domain is a suffix of the NIS+ domain.

2. The host table or map in the name service must be set up to support three types of `gethostbyname()` queries:

a. "mailhost"

Many name service configurations satisfy this requirement. Several typical configurations are described below.

i. NIS with DNS forwarding:

When the DNS forwarding feature is turned on, queries that NIS cannot resolve are forwarded to DNS.

ii. NIS+ in conjunction with DNS:

If your network uses both NIS+ and DNS as the source for the host database, you can put the `mailhost` entry in either the NIS+ or DNS host table. Make sure that your users list NIS+ and DNS as the source for the host database in the `/etc/nsswitch.conf` file.

iii. Using NIS+ or NIS as the only name service:

If your network uses only one of these name services, you must have a `mailhost` entry in the NIS host map or the NIS+ hosts table.

b. full host name (for example, `smith.Admin.Acme.COM`)

Many name service configurations satisfy this requirement. Several typical configurations are described below.

i. NIS with DNS forwarding:

Although NIS does not understand full host name, DNS does. This requirement is satisfied when you follow the regular procedure for setting up NIS and DNS.

ii. NIS+ and DNS:

Both NIS+ and DNS understand full host names. Following the regular NIS+ and DNS set up procedures satisfies this requirement.

iii. Using NIS+ as the only name service:

NIS+ understands the full host name. Following the regular NIS+ satisfies this requirement.

iv. Using NIS as the only name service:

This is a special case. The normal NIS setup does not understand the full host name. Rather than trying to make NIS understand the full host name, turn off this requirement from the `sendmail` side by editing the `sendmail.cf` file and replacing all the occurrence of `%l` with `%y`. This turns off `sendmail`'s inter-domain mail detection. As long as the target host can be resolved to a IP address, a direct SMTP delivery will be attempted. Make sure that your NIS host map does not contain any host entry that is external to the current mail domain. Otherwise, you will need to further customize the `sendmail.cf` file.

c. short host name (for example, `smith`)

`sendmail` must connect to `mailhost` to forward external mail. To determine if a mail address is within the current mail domain, `gethostbyname()` is invoked with the full host name. If the entry is found, the address is considered internal.

NIS, NIS+ DNS all support `gethostbyname()` with a short host name as an argument, so this requirement automatically is satisfied.

3. `Gethostbyname()` with full and short host name should yield consistent results. For example, `gethostbyname(smith.Admin.Acme.COM)` should return the same result as `gethostbyname(smith)` as long as both functions are called from the mail domain `Admin.Acme.COM`.

If you are using DNS in conjunction with NIS or NIS+, for every host entry in the NIS/NIS+ host table, you must have a corresponding host entry in DNS.

If you are using NIS, see Step 2 above for a description of how to turn off `gethostbyname()` with a full host name.

If you are using NIS+, see Step 4 below.

4. For all name service domains under a common mail domain, `gethostbyname()` with a short host name should yield the same result. For example, given the mail domain `smith.Admin.Acme.COM`, `gethostbyname(smith)` should return the same result calling from either domain `EBB.Admin.Acme.COM` or `ESG.Admin.Acme.COM`.

The mail domain name is usually shorter than the name service domain, giving this requirement special implications to various name services.

- *NIS*

All NIS host maps under a common mail domain should have the same set of host entries. For example, the host map in the `EBS.Admin.Acme.COM` domain should be the same as the host map in the `ESG.Admin.Acme.COM`. Otherwise, one address may work in one NIS domain but fail in the other NIS domain.

- *NIS+*

To satisfy this requirement, you may duplicate the host table as described in the NIS information above, or you may take advantage of the pathing/linking feature in NIS+. There are many ways to use this feature to satisfy this requirement; following is one example of a configuration:

- i. **Create an NIS+ table path (or object link) from the host table in the user name service domain to the host table in the mail domain level NIS+ domain.**
- ii. **Enter all hosts entries in the user name service domain(s) into the mail domain level domain.**

Because you are merging (logical or physically) multiple host tables into one host table, the same host name cannot be reused in the multiple name service domain sharing a common mail domain.

sendmail *Command-Line Arguments*

Use command-line arguments on the `/usr/lib/sendmail` command line. These arguments are described in Table C-12.

Table C-12 sendmail Command-Line Arguments

Argument	Description
<code>-r name</code>	Set the sender's name to <i>name</i> . This flag is ignored unless the real user is listed as a trusted user or unless <i>name</i> contains an exclamation point (because of certain restrictions in UUCP).
<code>-f name</code>	This is an obsolete form of <code>-r</code> .
<code>-h cnt</code>	Set the "hop count" to <i>cnt</i> . It sets the number of times this message has been processed by <code>sendmail</code> (to the extent that it is supported by the underlying networks). <i>cnt</i> is incremented during processing, and if it reaches the value of configuration option <code>h</code> , <code>sendmail</code> returns the message with an error.
<code>F name</code>	Set the full name of this user to <i>name</i> .
<code>-n</code>	Do not do aliasing or forwarding.
<code>-t</code>	Read the header for <code>To:</code> , <code>Cc:</code> , and <code>Bcc:</code> lines, and send to everyone listed in those lists. The <code>Bcc:</code> line is deleted before sending. Any names in the argument vector are deleted from the send list.
<code>-bx</code>	Set operation mode to <i>x</i> . Operation modes are: <ul style="list-style-type: none"><code>m</code> Deliver mail (default).<code>a</code> Run in ARPANET mode.<code>s</code> Use SMTP on input side.<code>d</code> Run as a daemon.<code>t</code> Run in test mode.<code>v</code> Just verify recipients.<code>i</code> Initialize the alias database.<code>p</code> Print the mail queue.<code>z</code> Freeze the configuration file.

Table C-12 `sendmail` Command-Line Arguments (Continued)

Argument	Description
<code>-qtime</code>	Try to process the queued-up mail. If the time is given, <code>sendmail</code> repeatedly runs through the queue at the specified interval to deliver queued mail; otherwise, it runs only once.
<code>-Cfile</code>	Use a different configuration file.
<code>-dlevel</code>	Set debugging level.
<code>-oxvalue</code>	Set configuration option <code>x</code> to the specified <code>value</code> .
<code>-M msgid</code>	Run given message ID from the queue.
<code>-R recipient</code>	Run messages for given recipient only from the queue.

These options are described in the next section, “Configuration Options.”

You can specify several configuration options as primitive flags. These are the `c`, `e`, `i`, `m`, `T`, and `v` arguments. Also, you can specify the `f` configuration option as the `-s` argument.

`sendmail` Configuration Options

You can set the options shown in Table C-13 using the `-o` flag on the command line or the `O` line in the configuration file.

Table C-13 `sendmail` Configuration Options (1 of 4)

Option	Description
<code>Afile</code>	Use the named <code>file</code> as the alias file instead of <code>/etc/mail/aliases</code> . If no file is specified, use <code>aliases</code> in the current directory.
<code>atime</code>	Wait a set amount of time for an <code>@: @</code> entry to exist in the alias database before starting up. If it does not appear after that time, rebuild the database.
<code>Bvalue</code>	Blank substitute. Default is the dot (<code>.</code>) character.
<code>bn</code>	Disallow empty messages to more than <code>n</code> recipients.
<code>Cn</code>	Do a check after <code>n</code> recipients.

Table C-13 sendmail Configuration Options (Continued) (2 of 4)

Option	Description
c	If an outgoing mailer is marked as being expensive, do not connect immediately. A queue process must be run to actually send the mail.
D	If set, rebuild the alias database if necessary and possible. If this option is not set, <code>sendmail</code> never rebuilds the alias database unless explicitly requested with <code>-bi</code> .
dx	Deliver in mode <i>x</i> . Legal modes are: <ul style="list-style-type: none"> i Deliver interactively (synchronously). b Deliver in background (asynchronously). q Queue the message (deliver during queue run)
ex	Dispose of errors using mode <i>x</i> . The values for <i>x</i> are: <ul style="list-style-type: none"> p Print error messages (default). q No messages, just give exit status. m Mail back errors to sender. w Write back errors (mail if user not logged in). e Mail back errors and always give zero exit status.
F <i>n</i>	The temporary queue file mode, in octal. Values of 644 and 600 are good choices for <i>n</i> .
f	Save UNIX-style <code>FROM</code> lines at the front of headers. Normally they are assumed to be redundant and discarded.
gn	Set the default group ID for mailers to run in to <i>n</i> .
H <i>file</i>	Specify the help file for SMTP [Postel 82].
h <i>n</i>	Set maximum hop count to <i>n</i> .
i	Ignore dots in incoming messages.
L <i>n</i>	Set the default log level to <i>n</i> .
M <i>xvalue</i>	Set the macro <i>x</i> to <i>value</i> . This is intended only for use from the command line.
m	Send to the sender also, even if the sender is in an alias expansion.

Table C-13 `sendmail` Configuration Options (Continued) (3 of 4)

Option	Description
<code>o</code>	Assume that the headers may be in old format; that is, spaces delimit names. This flag actually turns on an adaptive algorithm: If any recipient name contains a comma, parenthesis, or angle bracket, it is assumed that commas already exist. If this flag is not on, only commas delimit names. Headers are always output with commas between the names.
<code>Pname</code>	Send the header from error messages from the <code>MAILER-DAEMON</code> to this name (which is the local Postmater).
<code>Qdir</code>	Use the named <code>dir</code> as the queue directory.
<code>qlimit</code>	This is the size limit of messages to be queued under heavy load. Default is 10,000 bytes.
<code>Rserver</code>	Remote mode. Deliver through remote SMTP server. The default is location of <code>/var/mail</code> .
<code>rtime</code>	Timeout reads after <code>time</code> interval.
<code>Sfile</code>	Save statistics in the named <code>file</code> .
<code>s</code>	Always initiate the queue file, even if you are going to try immediate delivery. <code>sendmail</code> always initiates the queue file before returning control to the client under any circumstance.
<code>Ttime</code>	Set the queue timeout to <code>time</code> . After this interval, messages that have not been successfully sent are returned to the sender.
<code>un</code>	Set the default user ID for mailers to <code>n</code> . Mailers without the <code>S</code> flag in the mailer definition are run as this user.
<code>v</code>	Run in verbose mode.
<code>Xn</code>	Set the load average value, so that the <code>sendmail</code> daemon refuses incoming SMTP connections when the system is overloaded to reduce system load. The default is 0, which disables this feature.
<code>xn</code>	Set the load average value so that <code>sendmail</code> simply queues mail (regardless of the <code>dx</code> option) to reduce system load. Default is 0, which disables this feature.
<code>Yname</code>	Use the named NIS map for aliases. The default is <code>mail.aliases</code> .

Table C-13 sendmail Configuration Options (Continued) (4 of 4)

Option	Description
<i>yn</i>	Penalize messages with at least this many bytes per recipient.
<i>zn</i>	Penalize messages with at least this many bytes per delivery attempts.
<i>zn</i>	Penalize messages with this many bytes per class.

Mailer Flags

The flags you can set in the mailer description are described in Table C-14.

Table C-14 `sendmail` Flags Set in the Mailer Description

Flag	Description
C	Append the @domain clause from the sender to any names in the header that do not have an at sign (@) after being rewritten by rule set 3. This flag allows mail with headers with this form: From: <i>usera@local</i> To: <i>userb, userc@remote</i> to be automatically rewritten as: From: <i>usera@local</i> To: <i>userb@local, userc@remote</i>
D	Look for a Date: header line.
E	Escape From lines to be >From (usually specified with U).
e	Avoid connecting to this mailer, which is expensive, normally; any necessary connection occurs during a queue run.
F	Look for a From: header line.
f	Look for an -f from flag, but only if this is a network forward operation (that is, the mailer gives an error if the executing user does not have special permissions).
h	Preserve uppercase in host names for this mailer.
L	Limit the line lengths as specified in RFC 821.
l	Perform final delivery because this mailer is local.
M	Look for a Message-Id: header line.
m	Enable the mailer to send a message to multiple users on the same host in one transaction. When a \$u macro occurs in the argv part of the mailer definition, that field is repeated as necessary for all qualifying users. The L= field of the mailer description can be used to limit the total length of the \$u expansion.
n	Do not insert a UNIX-style From line on the front of the message.
P	Look for a Return-Path: line.

Table C-14 `sendmail` Flags Set in the Mailer Description (Continued)

Flag	Description
<code>p</code>	Always add local host name to the <code>MAIL FROM:</code> line of SMTP, even if there already is one.
<code>r</code>	Send a <code>-r</code> flag. Performs the same function as <code>-f</code> .
<code>S</code>	Do not reset the user ID before calling the mailer. This flag would be used in a secure environment where <code>sendmail</code> ran as root. This flag could be used to avoid forged names.
<code>s</code>	Strip quote characters off the name before calling the mailer.
<code>U</code>	Look for UNIX-style <code>From</code> lines with the UUCP-style <code>remote from <host></code> on the end.
<code>u</code>	Preserve uppercase in user names for this mailer.
<code>X</code>	Use the hidden dot algorithm as specified in RFC 821; basically, any line beginning with a dot will have an extra dot appended (to be stripped at the other end). This flag ensures that lines in the message containing a dot do not terminate the message prematurely.
<code>x</code>	Look for a <code>Full-Name:</code> header line.

Bibliography



This appendix contains a list of books that describe system administration and UNIX System V Release 4. Although these books are not specific to SunOS 5.2 systems, you may find them to be useful.

General References

UNIX System V Release 4, The Complete Reference; Stephen Coffin; published by Osborne McGraw-Hill, 1990; 905 pages.

UNIX System V Release 4, An Introduction; Kenneth H. Rosen, Richard R. Rosinski, and James M. Farber; published by Osborne McGraw-Hill, 1990; 1228 pages.

UNIX System Administration Handbook; Evi Nemeth, Garth Snyder, and Scott Seebass; Prentice Hall Software Series, 1989.

Essential System Administration; Arleen Frisch; O'Reilly & Associates, Inc., 1991.

System Performance Tuning; Mike Loukides; O'Reilly & Associates, 1990.

Managing NFS and NIS; Hal Stern; O'Reilly & Associates, 1991.

Practical UNIX Security; Simson Garfinkel and Gene Spifford; O'Reilly & Associates, 1991.

Electronic Mail References

sendmail, Bryan Costales with Eric Allman and Neil Rickert, published by O'Reilly & Associates, Inc., 1993; 792 pages.

!%@:: A Directory of Electronic Mail Addressing & Networks, 2nd Edition; Donnalyn Frey and Rick Adams, published by O'Reilly & Associates, Inc., 1990; 420 pages.

The DDN Protocol Handbook, 1985, a three-volume set of RFCs, and *Internet Protocol Handbook*, 1989, an additional volume, available from SRI International, 333 Ravenswood Avenue, Menlo Park, CA 94025.

RFC 1211, *Problems with Maintaining Large email Lists*.

RFC 822, *Standard for the Format of ARPA INTERNET Text Messages*.

Glossary

Administration Tool

An OpenWindows tool from which you can access Host Manager and Database Manager applications, which are used to administer NIS+ tables and local files in the `/etc` directory. You can also use these applications to examine the contents of NIS maps.

alert

A message generated by the LP print service to notify the system administrator, or any designated user, of requests by users to mount forms or character sets; a message generated by the LP print service to notify the system administrator, or any designated user, of printer faults.

alias

An alternative name or names assigned to a program or to an electronic mail address.

allow list

A named file containing a list of users, forms, or filters that is used by the LP print service to control access to printers.

Automounter

Software that automatically mounts a directory when a user changes into it, and unmounts the directory when it is no longer in use.

Auto_home database

The database that you use to add home directories to the Automounter. You access the Auto_home database using the Administration Tool.

bang

An exclamation point that acts as a single-character UNIX command or as a separator between the routes of a route-based email address.

banner page

A page printed with each print request that shows the user who submitted the print request, the request ID, and when the request was printed.

daemon

A special type of program that, once activated, starts itself and carries out a specific task without any need for user input. Daemons are typically used to handle jobs that have been queued, like printing, mail, and communication.

Database Manager

An application under Administration Tool that is used to administer NIS+ tables and local files in the `/etc` directory. You can also use the Database Manager to examine the contents of NIS maps.

deny list

A named file containing a list of users, forms, or filters denied access to printers.

default printer

The printer designated for each system as the destination for print requests when no printer name is used.

domain

A directory structure for electronic mail addressing and network address naming. Within the United States, top-level domains include `com` for commercial organizations, `edu` for educational organizations, `gov` for governments, `mil` for the military, `net` for networking organizations, and `org` for other organizations. Outside of the United States, top-level domains designate the country. Subdomains designate the organization and the individual system.

domain addressing

Using a domain address to specify the destination of an electronic mail message.

download filter

A fast filter that uses PostScript language structured conventions to decide which host-resident fonts to download to the printer for each print request.

downloading fonts

Copying fonts from a font directory on a system to a printer.

electronic mail

A set of programs that transmit mail messages from one system to another, usually over communications lines. Electronic mail is frequently referred to as *email*.

email

See **electronic mail**.

envelope

A file that the `sendmail` program creates when it parses electronic mail addresses that contains information about how to deliver the message.

environment variable

A system- or user-defined variable that provides information about the operating environment to the shell.

fast filter

A filter that quickly prepares a file for printing and requires access to the printer while the filter is processing.

file server

A system that shares its file systems with other systems in a network.

filter

A named file that converts a print request into a format that can be processed by a particular type of printer.

form

(1) A special paper, like letterhead or blank checks. (2) A named file that contains printing characteristics, like page length, page width, number of pages, line pitch, character pitch, character set choice, ribbon color, and alignment pattern. The LP print service uses the `lpforms` command to administer mounting and printing of forms.

fully qualified domain name

A domain name that contains all of the elements needed to specify where an electronic mail message should be delivered. (See also **domain**.)

gateway

A system that handles electronic mail traffic between differing communications networks.

GCOS

A field in the `/etc/passwd` file that is used to store identifying information, like name, office, extension, or home phone.

GID number

The group identification number used by the system to control access to accounts owned by other users.

Group database

The database that you use to create new group accounts or to modify existing group accounts. You access the Group database from Administration Tool.

home directory

The part of the file system allocated to an individual user for private files.

host-resident fonts

Fonts stored on one system that are shared by a group of users of a particular printer. Host-resident fonts are usually different from printer-resident fonts, and are used less frequently.

Hosts database

The database that you use to control network security. You access the Hosts database from the Administration Tool.

initialization files

The “dot” files in a user’s home directory that set the path, environment variables, windowing environment, and other characteristics to get users set up and functioning.

leading

The vertical distance between lines of text.

listenBSD

An LP print service daemon that is run on a SunOS 5.x print server to listen for print requests from SunOS 4.1 print clients on the network.

listenS5

An LP print service daemon that is run on a print server to listen for print requests from SunOS 5.x print clients on the network.

login name

The name assigned to an individual user that controls access to a system.

mail address	The name of the recipient and the location to which an electronic mail message is delivered.
mail alias	See alias .
mailbox	A directory on a mail host where mail messages are stored. Usually, the directory is named <code>/var/mail/username</code> and is automatically created the first time the user receives mail.
mail client	A system that does not provide mail spooling for its users. Mail is spooled on a mail server.
mailer	A protocol that specifies the policy and mechanics used by the <code>sendmail</code> program when it delivers mail.
mail host	The main mail system on a network that receives and distributes mail outside of the network or the domain. A mail host can also be a mail server.
mail server	Any system that stores mailboxes in the <code>/var/spool/mail</code> directory. A mail server can also serve as a mail host.
mail services	Services provided by a set of programs and daemons that transmit electronic mail messages between systems and distribute them to individual mail boxes.
mount point	A directory in the file-system hierarchy where another file system is attached to the hierarchy.
None (/etc)	In Administration Tool, <code>None (/etc)</code> means user information is in files on a user's local system, as opposed to a network-wide name service, like NIS+.
NIS	The SunOS 4.1 network information service.
NIS+	The SunOS 5.x network information service.

OpenWindows

A windowing system based on the OPEN LOOK® graphical user interface.

parse

To divide a string of characters or a series of words into parts to determine their collective meaning. Virtually every program that accepts command input must do some sort of parsing before the commands can be acted upon. For example, the `sendmail` program divides an email address into parts to decide where to send the message.

Passwd database

The database that you use to add, modify, or delete user accounts. You access the Passwd database from Administration Tool.

path

The list of directories that are searched to find an executable command.

path name

A list of directory names, separated with slashes (/), that specifies the location of a particular file.

permanently downloaded fonts

Fonts that are copied from a font directory on a system to a printer, and which remain stored in printer memory until the printer is turned off.

port

A physical connection between a peripheral device, like a terminal, printer, or modem, and the device controller.

port monitor

Networking daemons provided by the Service Access Facility that monitor incoming requests from the network to a port and execute the appropriate service. Each type of port has its own listener to handle incoming calls. `ttymon` and `listen` are both port monitor daemons that handle different types of requests.

print client

Any system on a network that has printing services provided by a print server.

print server

Any system that has a printer physically connected to it and that users can access over the network for printing services.

printer-resident fonts

Fonts that are stored permanently in memory on a printer.

Printer Manager

A distributed Administration Tool application for setting up printers, used instead of the LP print service's command line interface.

relay host

A system that transmits to and receives mail from outside the network or domain using the same communications protocol.

request ID

The identification number assigned by the LP print service to each print request.

RFC

Request for Comments, specify protocols and standards (for example, TCP/IP). RFCs are submitted to SRI-NIC, where they are assigned numbers and distributed by electronic mail to the Internet community. The most important RFCs (through 1985) are available in a three-volume publication, *The DDN Protocol Handbook*, which is available from SRI International in Menlo Park, California.

shadow file

The SunOS 5.x file in the `/etc` directory that contains user passwords.

shell

The command interpreter for a user, specified in the `Passwd` database; SunOS 5.x supports the Bourne, C, and Korn shells.

slow filter

A filter that takes a relatively long time to prepare a file for printing and does not require access to the printer while the filter is processing.

SMTP

Simple Mail Transport Protocol.

spooling directory

A directory where files are stored until they are processed.

spooling space

The amount of space allocated on a print server for storing requests in the printer queue.

universal address

An address of a type of network, like TCP/IP, in hexadecimal form, that is used to configure the port monitor to listen for print requests from print clients on a network.

UID number

The user identification number assigned to each login name. UID numbers are used by the system to identify, by number, the owners of files and directories.

user account

An account set up for an individual user in the `Passwd` database that specifies the user's login name, UID, primary GID, login directory, and login shell.

User Account Manager

A distributed Administration Tool application for adding and administering user accounts on a network-wide basis.

user mask

The setting that controls default file permissions assigned when a file or directory is created. The `umask` command controls the user mask settings.

Index

Symbols

- % sign, in mail addresses, 227
- @ sign, in mail addresses, 222, 226

A

- accept command, 103, 112, 285, 293, 297, 299
- Add Entry window, in User Account Manager, 29
- Add Local Printer window, 88
- addresses, mail. *See* mail addresses
- administration commands, printing, 123
- administration policies for users and groups, 4
- Administration Tool
 - applications, 2
 - available name services, 2
 - hardware and software requirements, 3
 - required access privileges, 3
 - starting, 3
- alerts
 - error checking, 307
 - form-mounting, 146
 - how to specify for character sets, 138
 - how to specify for forms, 146
 - how to specify for printer faults, 164
 - how to stop, 147
 - how to suppress, 139
 - setting, 156
 - types, 139
- aliasadm command, 230
- aliases, mail
 - alias file setting permissions, 372
 - creating, 257
 - definition of, 228
 - files, 228
 - for storing mail, 228
 - formats of, 230
 - how to check, 272
 - how to set up postmaster, 264
 - .mailrc file, 230
 - NIS aliases, 230
 - NIS+ aliases, 229
 - NIS+ table, format of, 229
- allow lists, control printer access, 34, 160
- architecture, setting for user's system, 38
- argument processing, by sendmail, 245
- argument vector, *See* sendmail
- ASCII file, 81
- auto_home database, 10, 29, 46
- Automounter, 9, 10, 28
- automounting home directory, 46

B

bang-style addressing, 221
banner pages
 deciding banner policy, 155
 how to control, 89, 162
 nothing else prints, 200
 troubleshooting incorrect output, 175
baud settings, checking, 175
/bin directory, 242

C

cable, pin configuration, 179
calendar, setting path to, 38
cancel command, 111
cartridge, printer, *See* print cartridges
character sets
 See also print wheels
 display names of, 128
 for printers, 126
 hardware, 128
 number, 127
class, printer
 about defining, 158
 cannot enable or disable, 116
 how to define, 166
classes, mail
 defining in rewriting rules, 386
 special, 395
comment field, in passwd file, 15, 46
comment, user account, 24
communications protocols, 357
conditionals, mail configuration file
 specifying, 395
configuration file, 404
configuration file, mail, 373 to 404
 See also sendmail.cf file; main.cf
 file; sendmail
 building from scratch, 402
 conditionals, specifying, 395
 definition of, 233
 delivery modes, 233
 file modes, 234

 list of options, 410
 log level, 234
 macros, 365, 384, 393
 overview of, 365
 precedence definitions, 388
 rule sets, 373
 sample sendmail configuration
 file, 375
 setting options, 388
 specifying an alternate configuration
 file, 368
 summary of functions, 362
 syntax of, 384
 time values, 233
 tuning configuration parameters, 368

CR ddn-gateway, entry in mail
 configuration file, 256
.cshrc file, 11, 39

D

daemon account, 5
daisy wheel printer, 282
Database Manager, 22
 adding groups, 21
 creating and maintaining groups, 9
 modify or deleting groups, 58
ddn mailer, 221, 256
dead.letter file, 246
delivery mode, mail, 233, 370
delivery path
 mail addressed with user and domain
 name, 248
 mail addressed with user and host
 name, 248
 mail addressed with user name, 247
 mail with fully qualified domain
 name, 249
 mail with sender's host name, 250
deny lists, to deny printer access, 160
deskset, setting path to, 38
 /dev/term/a file, 281
device name, printer, 79
directories

home, *See* home directory
 NFS-mounted, 40
 setting up search path, 40
 disable command, 103, 104, 117, 118, 163
 disk configuration, recommended for
 print servers, 78
 disk quota, 52
 DMsmartuucp, entry in mail
 configuration file, 256
 DNS naming service, 263
 setting up an alias file, 263
 domains
 definition of, 222
 list of top-level for countries, 223
 list of top-level in the United
 States, 223
 mail, 404
 “dot” files, *See* initialization files
 DR ddn-gateway, in mail configuration
 file, 256

E

enable command, 103, 104, 116, 163, 286,
 293, 297, 299
 encrypted password, 15
 env command, 32
 environment defaults, setting in system
 initialization file, 13
 environment variables, 11, 37
 adding or changing, 32
 ARCH, 38
 CALENDAR, 38
 cdpath, 38
 DESKSET, 38
 HISTORY, 32
 HISTORY, 38
 HOME, 11, 38
 LANG, 33, 38
 LC, 33
 LOGNAME, 11
 LPDEST, 38
 MAIL, 38
 MANPATH, 38
 OPENWINHOME, 39
 PATH, 31, 39
 prompt, 39
 PS1, 39
 setting for user applications, 32
 shell syntax for setting, 31
 TERM, 39
 TERMINFO, 39
 TZ, 11, 40
 Epson printer, 85
 error mailer, 399
 error messages, printing
 sent by printer, 156
 /etc/.login file, 13, 14, 19, 34, 13, 37
 /etc/.profile file, 19
 /etc/auto_home file, 29
 /etc/auto_master file, 10
 /etc/dfs/dfstab file, 51
 /etc/group file, 28
 /etc/hosts file, 73
 /etc/hosts file, adding home directory
 information to, 52
 /etc/hosts.lpd file, 191
 /etc/init.d/nfs.server
 command, 52
 /etc/lp directory, 310
 /etc/lp/fd directory, 283, 306
 /etc/lp/filter.table file, 285
 /etc/lp/printers directory, 310
 /etc/mail directory, 242
 list of mail files in, 242
 /etc/mail/aliases file, 230, 227, 228,
 254
 adding postmaster alias to, 264
 adding root, 262
 on NIS master, 260
 setting up, 261
 /etc/mail/sendmail.cf file, 256, 233
 /etc/motd file, 41
 /etc/named.boot file, 263
 /etc/passwd file, 28, 5
 /etc/printcap file, 296, 195, 295

`/etc/profile` file, 13, 14, 34, 37
`/etc/shadow` file, 48
`/etc/skel` directory, 14
`/etc/syslog.conf` file, 270
`/etc/vfstab` file, 10, 30
ether mailer, 221, 256
exit codes, printer interface
 customizing, 330
 table of, 330

F

faults, *See* printers, faults
file content types
 accepted by printers, 81
 any, 83
 converting with print filter, 333
 definition of, 81
 list of, 282
 PostScript, 81
 simple, 81
 troubleshooting incorrect output, 175
file modes, mail configuration, 234, 372
file permissions, assigning defaults, 41
files
 mail service, used by, 241
 print service, used by, 310, 312
filters, *See* print filters
fonts
 cartridges, setting up, 136
 downloading, 134, 201
 host-resident, 133, 135
 how to install, 151
 installing and maintaining PostScript
 fonts, 132
 printer-resident, 134
 styles, 126
forms, *See* print forms

G

gateway
 definition of, 217, 252
 used in mail service, 240

GCOS field, 15

group

nobody and noaccess, 15, 17
sysadmin, 7, 8, 28

group database

fields in, 16
group ID (GID) field, 17
group name field, 17
user list field, 17

group ID (GID)

field in group database, 17
field in passwd file, 15
number range, 17
system account numbers, 17
user account numbers, 17

group name field, in group database, 17

groups

assigning primary group to home
 directory, 51
creating new groups, 8
defined in default `/etc/groups`
 file, 8
entering primary group with User
 Account Manager, 24
entering secondary groups with User
 Account Manager, 24
how to add groups, 21
how to modify or delete groups, 58
modifying and deleting, 49
primary, 7, 15, 54
secondary, 7

groups command, 7

H

header lines

defining for mail service, 389
in mail messages, 364, 389

history log, print requests, 314

history, setting for the C shell, 38

home directory

overview, 9
 assigning to user's primary group, 51
 automatic mounting, 9, 10, 46

conventions for naming, 9
creating with User Account Manager, 26
field in passwd file, 15
how to manually create, 49 to 52
how to manually move the contents, 52
how to set up automounting, 29
mount point, 9, 29, 30
mounting in NFS, 29
permissions on, 51
setting up, 9
setting up automounting with User Account Manager, 27
setting up disk quotas, 52
sharing with other systems, 51
space allocation, 9
HOME environment variable, 14
/home mount point, 10

I

initialization files
See also system initialization file; user initialization files
assigning to user's primary group, 54
copied into home directory by User Account Manager, 12
copying in to home directory, 53
.cshrc file, 11
default files provided, 11
for users, 13
how to customize, 31 to 33
how to set up, 19 to 21
.login file, 12
permissions on, 21
.profile file, 11
specifying path with User Account Manager, 27
interface program, *See* printers, interface program
Internet Protocol (IP) address, 52, 280
IP address, *See* Internet Protocol address

L

language variables, 34, 38
left-hand side, mail message, 395
listen service
See also port monitor
check set up, 189, 198
example of setting up, 291
registering with the port monitor, 277, 289
starting on SunOS 5.1 print server, 289
types of, 277
load limit, setting for mail, 370
local mail service, 236
local passwords
changing, 64
creating, 63
local printer
definition, 70
example of setting up, 287
how to add, using Printer Manager, 87
overview of setting up, 74
SunOS 4.1, how to set up, 288
SunOS 5.1, how to set up using CLI, 281
local printing, diagram of process, 319
localization, 38
log files
for print service, 122, 174, 313
print queue, 313
print requests
cleaning out, 105
codes in request log, 316
history log, 314
log level, for mail, 234, 371
loghost, 270
.login file, 12
login names, *See* user names
login program, 11
login shell
See also Bourne shell; C shell; Korn shell

- features of, 36
- field in passwd file, 15
- specifying with User Account Manager, 24
- logins
 - checking for duplicates, 6
 - pseudo user, 5
- lp account, 25
- lp command, 114, 313
 - list of options, 308
- lp commands, hung, 204
- LP print service, *See* print service
- lpadmin command, 34, 85, 118, 281, 293, 297, 298
- lpd daemon
 - check configuration, 190
 - check if running, 190
- LPDEST, used to designate default printer, 84
- lpfilter command, 130, 284, 336
- lpforms command, 342
- lpmove command, 113, 118
- lpNet daemon, 105, 122
- lpNet log, check for problems, 174
- lpsched command, 120, 303, 305
- lpsched log, check for problems, 174
- lpshut command, 120, 305
- lpstat command, 101
 - options, 105
- lpsystem command, 119, 290, 293, 296

M

- macros, *See* mail macros
- mail, 404
 - aliases, *See* aliases, mail
 - classes
 - defining in rewriting rules, 386
 - special, 395
 - client
 - definition of, 219, 253
 - how to set up, 254
 - configuring parameters, 368

- delivery mode, 370
- file modes, 372
- load limits, 370
- log level, 371
- time values, 368
- delivery path, *See* delivery path
- files, list of, 242
- gateway, definition of, 217, 252
- mail host
 - definition of, 217, 252
 - how to set up, 255
- mail server
 - definition of, 218, 252
 - how to set up, 254
- relay host
 - definition of, 216
 - how to set up, 256
- router, 231
 - See also* sendmail
- sending to files and programs, 363
- troubleshooting, *See* mail problems
- mail addresses
 - % sign, 227
 - @ sign, 222
 - bang-style, 221
 - definition of, 225
 - local address, 222
 - parsing by sendmail, 390
 - parsing by sendmail, 361
 - rewriting rules, 390
 - route-based, 225
 - route-independent, 226
- mail aliases, *See* aliases, mail
- mail configuration file, *See* configuration file, mail
- mail connections, how to verify, 273
- mail daemon, 367
- mail macros
 - in configuration file, 365, 384
 - list of definitions, 393
 - predefined lowercase macros, 393
 - semantics of, 392
- mail messages
 - body of, 364
 - collecting, 361

-
- collection, by `sendmail`, 245
 - delivering, 361, 364
 - delivery, by `sendmail`, 246
 - header format, 364
 - headers, 362, 365, 389
 - left-hand side rule, 395
 - queueing, 362, 364
 - queues for, 246
 - rewriting rules, 395
 - right-hand side rule, 396
 - setting size of, 371
 - setting timeout value, 369
 - statistics, 219
 - mail problems
 - debugging `sendmail`, 367
 - diagnostic information, 273
 - how to check aliases, 272
 - how to test `sendmail` program, 272
 - how to verify connections to other systems, 273
 - troubleshooting, 272
 - mail queue
 - format of files in, 267
 - how to replace, 268
 - how to run an old queue, 269
 - looking at, 266
 - setting time interval, 369
 - mail service
 - files used by, 241
 - local, 236
 - logging errors, 270
 - programs, 212
 - `syslogd` program, use of, 270
 - using a gateway, 240
 - using a router, 238
 - with `uucp` connection, 237
 - `mail_aliases` table, 228
 - mailbox, 218, 226
 - creating an entry for root in NIS files, 261
 - names, 227
 - setting path to, 38
 - mailer, 216, 220
 - `ddn`, 221, 256
 - declarations, 365
 - defining, 391
 - defining flags, 399
 - descriptions, 399
 - designating as expensive, 400
 - error, 399
 - ether, 221, 256
 - `smartuucp`, 256
 - `uucp`, 221, 256
 - `MAILER-DAEMON`, 227, 273
 - `.mailrc` file, 229
 - `mailtool` command, 213, 231
 - `mailx` program, 213
 - `main.cf` file, 233, 375
 - `mconnect` program, 273
 - memory, required for the print server, 78
 - message headers, mail, 362, 365, 389
 - message of the day, 41
 - messages, mail, *See* mail messages
 - metasymbols
 - for left-hand side of mail message, 395
 - for right-hand side of mail message, 396
 - MOTD, *See* message of the day
 - `mount` command, 30
 - mount point, home directory, 9
 - `mountall` command, 30
 - multibyte characters, 34
- ## N
- name rewriting, in mail messages, 366
 - name service
 - checking user names and UIDs, 6
 - enabling access between systems on network, 73
 - how `sendmail` interacts with, 404
 - impact on setting up user and group accounts, 44
 - Name Services Administration Guide*, 173
 - naming convention, home directory, 30
 - NEC printer, 85
 - `newaliases` command, 262

newgrp command, 7, 17
 NIS mail aliases
 format of, 230
 setting up aliases files, 260
 NIS passwords, 62
 NIS+ aliases table
 adding entries to, 258
 changing entries in, 259
 deleting entries in, 260
 editing, 259
 format of, 229
 listing the contents of, 257
 NIS+ name service
 credentials, setting up with User Account Manager, 27
 registering printer with, using Printer Manager, 90
 sendmail requirements, 406
 NIS+ passwords, 61
 nisaddcred local command, 9
 nispasswd command, 46, 48, 61
 nistbladm command, 90
 nlsadmin command, 289, 290
 noaccess account, 5, 15, 17
 nobody account, 5, 15, 17
 non-PostScript printers, list of, 96

O

OpenWindows
 setting path to, 39

P

parallel port, 87, 88, 281
 parity bit, check setting, 176
 passwd command, 46, 48, 61, 63, 64
 passwd file
 comment field, 15, 46
 fields in, 14
 group ID field, 15
 home directory field, 15
 login shell field, 15
 password field, 15
 user ID field, 15
 user name field, 15
 password field, in passwd file, 15
 passwords, 47
 aging, 47
 changing local passwords, 64
 creating and changing, 47
 creating in NIS, 62
 creating in NIS+, 61
 creating with no naming service, 63
 field in passwd file, 15
 how to create or change, 60 to 60
 local, 63
 setting status with User Account Manager, 25
 PATH environment variable, 13, 124
 percent sign (%), in mailbox names, 227
 pin configuration in cables, 179
 pmadm command, 290
 policies, setting for printing, 154, 162
 port monitor
 check set up, 189, 198
 configure as a listener, 290
 how to configure, 289
 reasons for configuring, 277
 register listen service with, 289
 postmaster
 alias, how to set up, 264
 duties of, 266
 editing the entry in NIS files, 261
 PostScript printers
 default print filters, 279
 example of setting up, 287
 file content type, 81
 printer type, 80, 86
 precedence, setting in mail configuration file, 388
 primary group
 assigning initialization files to, 54
 specifying with User Account Manager, 24
 print cartridges
 definition, 128
 specifying alerts for, 139

-
- print client
 - definition of, 70
 - SunOS 5.x, port monitor and listen services, 75
 - print clients
 - check configuration, 183
 - disconnect from a server, 118
 - freeing jobs in, 207
 - SunOS 4.1, example of setting up, 296
 - SunOS 5.1, example of setting up, 294
 - SunOS 5.1, how to set up, 292
 - SunOS 4.1, how to set up, 295
 - print daemons, 312
 - print filters, 305
 - bypassing, 200, 201
 - changing, 129
 - characteristics of, 337 to 339
 - creating, 129, 332
 - defining, 336, 350
 - displaying definition, 142
 - download, 134
 - handling special modes, 333
 - how to provide new filter, 349
 - how to remove, 141
 - initializing, 283
 - list of installed, 143
 - list of non-PostScript content types, 285
 - not provided in SunOS 5.1, 279
 - options keywords, 340
 - PostScript, default, 279
 - removing, 129
 - requirements for, 334
 - restoring, 129
 - setting up, 283
 - templates to define options, 339
 - to convert from troff to PostScript, 351
 - types of, 332
 - used to convert files, 333
 - used to put request on hold, 334
 - print forms, 130
 - adding, removing, or changing, 131
 - alerts for mounting, 131
 - allowing printer access, 150
 - allowing user access, 149
 - controlling access to, 132
 - creating, 342
 - denying printer access, 151
 - denying user access, 150
 - displaying attributes, 132
 - how to add, 143
 - how to change, 144
 - how to display status of, 148
 - how to get information about, 147
 - how to mount, 145
 - how to remove, 144
 - how to specify alerts for, 146
 - how to stop alerts, 147
 - how to unmount, 145
 - mounted on printer, 306
 - mounting, 131
 - providing a new form, 353
 - print jobs, *See* print requests
 - print queue
 - controlling, 103
 - how to reposition a request, 114
 - log files, 313
 - look at requests, 186
 - setting priorities for print requests, 159
 - print requests
 - accepting, 103
 - canceling, 101
 - changing priority of, 102
 - controlling flow to queue, 103
 - how to accept or reject, 112
 - how to cancel, 111
 - how to change priority level, 114
 - how to check status of, 105
 - how to move requests to another printer, 113
 - how to put at head of queue, 115
 - how to put on hold, 115
 - how to resume printing, 115
 - log file, 105, 123
 - managing, 101
 - moving, 102
 - moving to head of queue, 102
 - putting on hold, 102

-
- rejecting and accepting, 103
 - scheduling, 305
 - tracing status, 306
 - print scheduler
 - how to start and stop, 120
 - if not running, 181
 - starting and stopping, 104, 305
 - trouble starting, 120
 - updating LP system files, 303
 - print server
 - overview of setting up, 277
 - check access to, 185
 - check configuration, 183
 - check connections, 182
 - check if correctly identified, 186
 - configurations, 75
 - centralized configuration, 75
 - combined distributed and centralized configuration, 76
 - distributed configuration, 75
 - definition of, 70
 - how to disconnect a client, 118
 - printer type for, 293
 - recommended disk configuration, 78
 - requirements
 - hard disk, 78
 - memory, 78
 - spooling space, 77
 - swap space, 79
 - system requirements, 76
 - SunOS 4.1, how to set up, 292
 - SunOS 5.1, how to set up using CLI, 288
 - SunOS 5.x, how to set up with Printer Manager, 87
 - type, 92
 - print service
 - administrative commands, 313
 - check basic functions, 181, 185
 - commands, list of, 124
 - compatibility with Sun 4.1 printing, 71, 288
 - configuration files, 309
 - content of `lp` directory, 310
 - customizing, 327
 - daemons, 312
 - files used by, 312
 - how to start scheduler, 120
 - how to stop the scheduler, 121
 - hung `lp` commands, 204
 - interface program, 306
 - list of directories, 308
 - log files
 - how to monitor and clean out, 122
 - See also* log files
 - scheduler, *See* print scheduler
 - structure of, 308
 - troubleshooting, 173, 176
 - See also* printers, troubleshooting
 - user commands, 308
 - print wheels, 128
 - setting up, 136
 - specifying alerts for, 139
 - tracking, 307
 - `printcap` entry, checking, 195
 - `printcap` file, 309
 - printer deny list, 47
 - Printer Manager
 - overview of, 71
 - Access to Remote printer window, 91
 - Add Access to Remote Printer option, 72
 - Add Local Printer option, 72
 - Goto option, 73, 88, 91
 - how to add a printer, 87
 - Modify Printer window, 93
 - modifying or deleting printers, 93
 - viewing printer information, 94
 - printers
 - See also* local printer; PostScript printers
 - overview to instructions for setup, 86, 280
 - access, controlling user access, 90
 - adding a printer with Printer Manager, 87
 - administration, 100
 - administration commands, 124

alerts, *See* alerts
 baud settings, 175
 character sets, *See* character sets
 characteristics, displaying, 109
 class
 about defining, 158
 cannot enable or disable, 116
 how to define a class, 166
 configuration information, 79, 279
 controlling access, 161
 controlling availability of, 103
 daisy wheel, 282
 default printer, 38, 84, 89, 293, 297
 description, 83, 293, 297
 how to add to print server, 166
 used to identify a printer, 159
 destination
 determining the default
 destination, 154
 how to designate, 162
 device name, 79
 displaying characteristics, 109
 enabling and disabling, 104, 155
 Epson, 85
 example of checking availability, 108
 example of obtaining all status
 information, 108
 example of obtaining status of all
 printers, 109
 example of removing from
 service, 119
 faults
 detecting, 334
 how to specify alerts, 164
 messages, 331
 restarting printing, 206
 types of fault recovery, 157
 file content types
 accepted by printers, 81
 affect on filtering, 200
 any, 82
 common types, 82
 definition of, 81
 incorrect output (only banner
 prints), 200
 list of, 282
 simple, 81
 fonts, *See* fonts
 forms, *See* print forms
 how to allow or deny user access, 90,
 168
 how to check status of, 105
 how to define class, 166
 how to disable, 116
 how to enable, 116
 how to modify or delete, using Print
 Manger, 93
 how to remove from service, 118
 how to view printer information,
 using Printer Manager, 94
 interface program
 customizing, 328, 332, 348
 exit codes, 330
 standard, 329
 used by LP print service, 306
 local, 319
 name, 79
 parity bit, 176
 port
 adjusting characteristics, 327
 device name, 79, 281
 how to adjust characteristics, 344
 paralleled versus serial, 281
 parallel versus serial, 87, 88
 registering with NIS+ using Printer
 Manager, 90
 remote, 321
 removing, 104
 settings
 baud, 175
 return, 176
 stty, 202, 203, 330, 344
 tab, 176
 status
 conflicting messages, 178
 how to check, 105
 stty settings, 202, 203
 troubleshooting
 See also printers, settings
 backup in printer queue, 177
 conflicting status messages, 178,
 207

- hung `lp` commands, 204
 - hung printers, 177, 205
 - not accepting requests, 182
 - only banner page prints, 200
 - print service commands, 176
 - problems with printer
 - output, 174, 199
 - scheduler not starting, 121
- type, 80
 - defined by `terminfo` entry, 311
 - incorrect output, 199
 - unusual types, 86
- printing
 - administration commands, quick
 - reference, 123
 - banner pages, 155
 - configurations
 - centralized configuration, 75
 - combined distributed and
 - centralized
 - configuration, 76
 - distributed configuration, 75
 - local, diagram of process, 319
 - network prerequisites, 73
 - priorities
 - how to designate limits, 167
 - setting for print requests, 159
 - remote, 321
 - setting policies, 154, 162
 - special modes, 333
 - spooling directory, 318
 - status messages, conflicting, 207
 - priorities, *See* printing
 - `.profile` file, 11, 39
 - pseudo user logins, 5

Q

- `qf` file, 267
 - codes in, 267
- queues
 - See also* print queues
 - for mail messages, 246, 364
 - messages by `sendmail`, 246
 - setting interval, 366

R

- read timeouts, mail configuration, 369
- `reject` command, 103, 112, 118
- relay host, 253
 - definition of, 216
 - how to set up, 256
- relay mailer, 256
- remote printing, diagram of process, 321
- requests log, codes in, 316
- return settings, checking, 176
- rewriting rules
 - left-hand side, 395
 - right-hand side, 396
 - semantics of, 398
- right-hand side, of a mail message, 396
- root account, 5
- routes, for mail, 225
 - See also* addressing, mail
- rule sets
 - mail configuration file, 373
 - order of, 374
 - semantics, 398

S

- SAC (service access controller), 288
- `sacadm` command, 289
- `SCHEDLOCK` file, 120, 122
- scheduler, *See* print scheduler
- search path for directories, 39, 40
- search path, set up to include home
 - directory, 31
- Security, Performance, and Accounting Administration*, 40, 47
- semantics
 - for rewriting rule sets, 398
 - of mailer descriptions, 399
- sendmail
 - name services requirements, 405
 - requirements for name services, 405
- sendmail
 - debugging, 367

-
- sendmail, 214, 220
 - address parsing, 361
 - argument processing, 245
 - argument vector, 360, 361
 - arguments to, 366, 409
 - configuration file, 362
 - configuration table, 257, 385
 - diagram of programs, 359
 - exit status, 360
 - features of, 358
 - flags, 414
 - how to test, 272
 - implementation, 362
 - interactions, 244
 - interfaces, 360
 - message collection, 245, 361
 - message delivery, 246
 - name parsing, 366
 - NIS+ aliases, 229
 - operation of, 360
 - returning messages to sender, 362
 - sample configuration file, 375
 - setting file mode, 372
 - setting load limit, 370
 - setting log level, 371
 - setting queue interval, 366
 - starting daemon, 367
 - TCP connections, 360
 - testing, 403
 - sendmail configuration file, *See*
 - sendmail.cf file; configuration file, mail
 - sendmail.cf file, 404
 - sendmail.cf file, 214, 233
 - sendmail.mx program, 263
 - sendmail.subsidiary.cf file, 233
 - sendmailvars table, 224, 257, 385
 - serial port, 87, 281
 - serial printers, adding to workstation, 76
 - share command, 9, 51
 - shareall command, 51
 - shell
 - customizing, 37
 - initialization files, 11
 - setting prompt, 39
 - simple file content type, 81, 83
 - skeleton files, *See* initialization files
 - smartuucp mailer, 220, 256
 - SMTP protocol, 405
 - SMTP protocol, 221, 357, 360
 - space allocation, home directory, 9
 - spooling directory
 - creating for a printer, 195
 - creating for a SunOS 4.1 print client, 295
 - print service, 318
 - spooling space, needed for a print server, 77
 - status
 - all printers, 109
 - how to check status of printer and print requests, 105
 - stty settings
 - customizing, 330
 - default settings, 202, 344
 - suggestions, 203
 - swap space, required for print server, 79
 - sysadmin group, 7, 8, 28
 - syslogd program, used by mail services, 270
 - system initialization file
 - /etc/.login file, 13, 14
 - /etc/profile file, 13, 14, 37
 - setting defaults in, 13, 37
 - setting for multibyte characters, 34
 - system profile, *See* system initialization file
- ## T
- tab settings, checking, 176
 - TCP/IP, 357
 - templates, for print filters, 339
 - terminal characteristics, changing, 330
 - terminal variable, 39
 - terminfo database, 39, 84
 - for unsupported printer, 327, 345

- list of character sets, 127
 - list of required items, 346
 - printer entries, 84
 - time intervals
 - setting for mail, 368
 - setting for mail queues, 369
 - setting for message timeout, 369
 - setting read timeout, 369
 - time values, for mail configuration
 - file, 233
 - time zone, setting, 40
 - tput command, 136
 - TranScript filters, 279
 - transport agent, 220
 - troff filter to convert to PostScript, 351
 - troubleshooting
 - See also* mail problems
 - See also* printers, troubleshooting
 - file content type, 175
 - mail services, 272 to 273
 - mail, debugging sendmail, 367
 - printing problems, 171 to 207
 - trusted users, defining for mail
 - service, 389
- U**
- UID, *See* user IDs
 - umask settings, 33, 41
 - universal address, 280, 290
 - User Account Manager
 - account security, defining, 26
 - Add Entry window, 29
 - Add User option, 24
 - automounting home directory, 18
 - automounting home directory, setting up, 27
 - comment, entering, 24
 - copying user initialization files into home directory, 12
 - home directory, creating, 26
 - initialization files, specifying path, 27
 - login shell, specifying, 24
 - mail server, specifying, 27
 - NIS+ credentials, setting up, 27
 - prevents duplicate UIDs, 6
 - primary group, specifying, 24
 - secondary groups, specifying, 24
 - setting password status, 25
 - user IDs, entering, 24
 - user names, entering, 24
 - user accounts
 - overview, of setting up, 17
 - adding, 17
 - checking for duplicates, 6
 - comment entered with User Account Manager, 24
 - disabling for security, 47
 - how to add a user account, ?? to 28
 - how to delete, 57
 - how to disable, 58
 - how to manually move the contents, 52
 - how to modify, 54
 - login shell, specifying with User Account Manager, 24
 - modifying, 46
 - moving or changing, 17, 46
 - security, 26
 - setting password status with User Account Manager, 25
 - user agent, 219
 - user environment
 - defining default shell, 11
 - initialization, 11, 12
 - setting up, 49
 - user ID field, in passwd file, 15
 - user IDs, 6
 - ensuring uniqueness, 6
 - entering with User Account Manager, 24
 - field in passwd file, 15
 - numbers for system accounts, 6
 - numbers for user accounts, 6
 - reusing numbers, 47
 - standard for choosing, 4
 - user initialization files, 13, 20
 - user list field, in group database, 17

user mask, 33, 41
 See also umask

user name field, in passwd file, 15

user names

- entering with User Account Manager, 24
- home directory name matches, 51
- maximum length of, 15, 24
- standard for choosing, 4

user profile, *See* user initialization files

user setup, completing, 28

/usr/lib directory, 124

- list of mail files in, 243

/usr/lib/lp/lpsched command, 120

/usr/lib/lp/lpshut command, 305

/usr/lib/lp/model directory, 306

/usr/lib/lp/postscript directory, 285, 305

UUCP, 357

uucp account, 5, 25

uucp connection, 237

uucp mailer, 220, 221, 231, 256

uux mailer, 220

V

/var partition, 77

/var/lp/logs directory, 105

/var/lp/logs/lpNet log file, 174

/var/lp/logs/lpsched log file, 174

/var/mail directory, 218, 226

- list of mail files in, 243

/var/saf/tcp/log file, 290

/var/spool/cron/crontabs file, 105, 123

/var/spool/lp directory, 77, 318

/var/spool/lp/requests directory, 313

/var/spool/lp/tmp directory, 313

vi, setting up default shell for, 39

Y

yppasswd command, 46, 48, 62, 63

