ULTRIX

Guide to System and Network Setup

Order Number: AA-ME88C-TE
December 1991

Product Version: ULTRIX and UWS Version 4.2A

digital equipment corporation
Maynard, Massachusetts

Contents

About This Manual

1 Overview

2 Network Setup

3 Distributed System Services Setup

4 System Setup

A Device Mnemonics

B System Files

C Support of the CI/HSC Hardware

D Managing the Print System

E Monitoring and Managing System Accounting

Index

Examples

Figures

Tables

About This Manual

The *Guide to System and Network Setup* identifies a series of tasks that are basic to setting up your system and establishing it on a network, and provides step-by-step information on how to complete each task. If the ULTRIX software has a script that simplifies completing the task, the script is described. The guide also tells you where you can find additional information on each topic.

Prerequisites

This book presumes that you have successfully completed your ULTRIX installation, and that you correctly installed any optional subsets that you need. The basic installation provides the following subsets: Base System, Kernel Configuration Files, TCP/IP Networking Utilities, Network File System Utilities, Extended (Berkeley) Mailer, X11/DECwindows 75dpi Fonts, X11/DECwindows Servers, and X11/DECwindows User Environment.

You must add the optional software subset Printer Support Environment to your system if you intend for your system to access printers either locally or remotely. See the *Guide to Installing ULTRIX* for more information on the mandatory and optional subsets provided by the ULTRIX software installation.

Scope

This book discusses system and network setup tasks for workstation and servers. It does not address the following areas:

- System security—For information on setting up a secure local host or a secure networked environment, see the *Security Guide for Administrators* or the *Guide to Kerberos.*
- Diskless client setup—For information on managing the diskless client environment, see the *Guide to Sharing Software on a Local Area Network.*

Although this guide assumes a first-time installation, most of the task descriptions and steps are applicable if you are modifying an existing environment. If you cannot complete a particular task because of your system or network configuration, the software indicates that there is a conflict and what to do resolve it.

Audience

The audience for this document is a workstation user with `root` privilege or a system and network administrator.

Organization

The *Guide to System and Network Setup* is divided into four chapters and five appendixes:

Chapter 1	Overview
	This chapter presents a general introduction to all system and network setup tasks, explaining which tasks are optional and which are required for workstations and servers.
Chapter 2	Network Setup
	This chapter outlines the tasks required to establish a TCP/IP local area network, and to access remote resources.
Chapter 3	Distributed System Services Setup
	This chapter outlines the tasks required to establish a distributed environment.
Chapter 4	System Setup
	This chapter outlines the system management tasks required to set up your local system.
Appendix A	Device Mneumonics
	This appendix lists all the device mneumonics supported by the ULTRIX operating system.
Appendix B	System Files
	This appendix discusses the system files that make up the ULTRIX operating system.
Appendix C	Support of the CI/HSC Hardware
	This appendix discusses the CI/HSC hardware and its implementation in the ULTRIX operating system.
Appendix D	Managing the Print System
	This appendix discusses printer parameters and explains how to manage the print system.
Appendix E	Monitoring and Managing System Accounting
	This appendix explains how to keep track of daily operations such as user logins, command usage, and printer usage.

Conventions

`%` or `$`	A percent sign represents the C shell system prompt. A dollar sign represents the system prompt for the Bourne and Korn shells.
`#`	A number sign represents the superuser prompt.
`file`	Italic (slanted) type indicates variable values, placeholders, and function argument names.

[|] **{ | }**	In syntax definitions, brackets indicate items that are optional and braces indicate items that are required. Vertical bars separating items inside brackets or braces indicate that you choose one item from among those listed.
. . .	In syntax definitions, a horizontal ellipsis indicates that the preceding item can be repeated one or more times.
`cat`(1)	A cross-reference to a reference page includes the appropriate section number in parentheses. For example, `cat`(1) indicates that you can find information on the `cat` command in Section 1 of the reference pages.
Return	In an example, a key name enclosed in a box indicates that you press that key.
Ctrl/*x*	This symbol indicates that you hold down the first named key while pressing the key or mouse button that follows the slash. In examples, this key combination is enclosed in a box (for example, Ctrl/C).

New and Changed Information

This guide now contains the information previously contained in the *Guide to System Environment Setup*. In addition, there are now more detailed explanations of how to set up your network and system and – when appropriate – examples of command-line shell scripts to simplify your setup tasks.

Overview 1

After completing your installation of ULTRIX and UWS, you will need to set up an efficient working environment and, in all likelihood, connect your system to a network so that you can avail yourself of distributed network services such as BIND/Hesiod and Yellow Pages, and the network file system (NFS), which enables you to import files from or export files to other machines on the same network.

In most cases, setting up a network and a working environment consists of a relatively simple series of tasks that are performed either manually or with system setup scripts, although many system administrators write their own all-inclusive setup scripts that automate the entire process.

It is a good idea to write such setup scripts, even if you reinstall infrequently, because they greatly simplify the setup of your environment, reduce the tendency to make mistakes, and, by accelerating the process, allow you to begin doing actual work more quickly.

As with all of UNIX, many of the setup tasks described in this document can be performed quicker by employing certain shortcuts and command-line shell scripts; whenever appropriate, this guide explains these shortcuts to enable you to work more efficiently. However, regardless of whether or not you understand shell programming, you will be able to perform every setup task listed in this document.

Setup Tasks for Workstations and Servers

While many setup tasks are the same for both workstations and servers, some variance does exist. This guide always begins by explaining tasks common to both. Whenever exceptions occur, separate instructions are listed for workstations or servers.

Table 1-1 lists, in the order in which they are generally performed, the setup tasks that are required or optional for both workstations and servers. The symbol [Yes] emphasizes an optional setup task that you would probably perform when setting up your system.

Table 1-1: Setup Tasks for Workstations and Servers

Setup Task	Workstation		Server	
	Required	Optional	Required	Optional
Setting up or connecting to a network	Yes†	–	Yes	–
Setting up a router	–	Yes††	–	Yes††
Modifying the Simple Network Management Protocol Agent	No	Yes	No	Yes
Setting up the Network File System	No	Yes	No	Yes
Setting up YP	No	Yes	No	Yes
Setting up BIND/HESIOD	No	Yes	No	Yes
Setting up the `svc.conf` file	Yes†††	–	Yes†††	–
Setting up the network time services	Yes††††	Yes	Yes††††	Yes
Adding users	Yes	–	Yes	–
Adding devices	No	Yes	No	Yes
Adding pseudoterminal devices	No	Yes	No	Yes
Adding LAT devices	No	No	No	Yes
Establishing disk quotas	No	Yes	No	Yes
Adding printers	No	Yes	No	Yes

† Only if you installed from tape or CDROM
†† Only if you have more than one network adapter
††† Only if you are running YP or BIND
†††† Only you are running NFS or Kerberos

The following chapters explain how to perform each of the setup tasks listed in Table 1-1.

Network Setup 2

This chapter discusses the following network setup tasks:

- Setting up or connecting to a local area network (LAN) with `netsetup`
- Setting up and accessing a router
- Modifying the Simple Network Management Protocol Agent with `snmpsetup`
- Setting up the Network File System with `nfssetup`

Overview

After you have completed your installation of ULTRIX and UWS, you should immediately set up or connect your system to a local area network (LAN).

Note

If your system is a workstation that used the Remote Installation Service (RIS) to install ULTRIX and UWS, you do not have to set up or connect to a LAN. At the time of the RIS installation, your workstation had all of the LAN setup tasks performed silently by the Remote Installation Service and your system is already connected to the LAN.

You will, however, have to populate the `/etc/hosts` file. For more information on how to do this, see the note in step 11 in the section "Steps."

Local area networks facilitate information exchange and resource sharing by linking together the machines at your site. Establishing a LAN requires careful planning based on many factors, including your current and projected networking needs, cost, reliability and maintainability, and geography. For more information on setting up a LAN, see *Introduction to Networking and Distributed System Services.*

After you have setup or connected to a LAN, you may elect to setup a Simple Network Management Protocol Agent to monitor network traffic or – if you intend to connect your system to more than one network – a router. However, before performing either of these tasks, you must first set up or connect to a LAN.

Network Setup Tasks

Table 2-1 lists, in the order in which they are generally performed, the network setup tasks that are required or optional for both workstations and servers. The symbol Yes emphasizes an optional setup task that you would probably perform when setting up your system.

Table 2-1: Network Setup Tasks for Workstations and Servers

Setup Task	Workstation		Server	
	Required	Optional	Required	Optional
Setting up or connecting to a network	Yes†	–	Yes	–
Setting up a router	–	Yes††	–	Yes††
Modifying the Simple Network Management Protocol Agent	No	Yes	No	Yes
Setting up the Network File System	No	Yes	No	Yes

† Only if you installed from tape or CDROM
†† Only if you have more than one network adapter

Setting Up or Connecting to a Network with netsetup

A local area network (LAN) is a group of two or more computer systems connected by a transmission medium. Each computer system (also called a host) is connected to the transmission medium by a hardware interface.

Every LAN should be assigned a unique network address (also called a network number) by the Network Information Center (NIC). Every host connected to the LAN is assigned an address by the local system administrator that includes the LAN's network address and a host address (also called a host number), which is unique to that host. All of the hosts on a particular LAN share the same network address.

If you are a system administrator, construct a plan for assigning unique host addresses to each system on a particular network before you set up your network.

Note

If you do not intend to connect to the Internet, it is still a good idea to get an Internet address from the NIC, because if at some later date you do connect to the Internet and have used your own addressing scheme, you will have to reassign Internet addresses to all the hosts on your LAN.

The `netsetup` command with the `install` option automates establishing and adding nodes to a LAN through interactive prompts, and places entries in the following system files:

- `/etc/rc.local`
- `/etc/networks`
- `/etc/hosts`
- `/etc/hosts.equiv`

After your LAN is established, you can use the `netsetup` command without the `install` option to update the `/etc/hosts` and `/etc/hosts.equiv` files. See the `netsetup`(8) reference page for more information.

Warning

If your system is already connected to a LAN and you want to connect your system to an additional LAN, you cannot use `netsetup`. The `netsetup` program invoked with the `install` option overwrites all previous network configurations in all the network files.

If you are already connected to a LAN and will be connecting to multiple LANs, see the section "Setting up a Router" in this chapter.

Before You Start

If you are setting up a new LAN, you must obtain a network address from the NIC and have a clear understanding of TCP/IP networking concepts. For information on obtaining a network address, and a discussion of TCP/IP networking concepts, see the *Introduction to Networking and Distributed System Services.*

If you are connecting your system to an existing LAN, read the first chapter of *Introduction to Networking and Distributed System Services* before running `netsetup`.

In either case, whether you are setting up a new LAN or connecting to an existing LAN, have the first chapter of *Introduction to Networking and Distributed System Services* near at hand and refer to it while you gather the following prerequisite information, which you will use to answer the `netsetup` prompts. It will provide explanations for questions that may arise.

Gathering Prerequisite Information

Table 2-2 lists the prerequisite information that you will need to gather to complete `netsetup` and shows whether the user, system administrator, or the NIC is able to provide that information. The term user refers to anyone who has `root` privilege on a workstation and is not a system administrator.

Note that all information that the user can determine can also be requested from the system administrator. If you have any doubts about being able to determine any of the information marked as User in the following table, request that information from your system administrator.

Table 2-2: Who Can Provide Prerequisite Information

netsetup Information	Setting Up a New LAN	Connecting to an Existing LAN
Internet address (network and host addresses)	NIC/Sys Admin	Sys Admin
Broadcast address	Sys Admin	User
If subnet routing is used and, if so, the number of bits used for the subnet address	Sys Admin	User

Table 2-2: (continued)

netsetup Information	Setting Up a New LAN	Connecting to an Existing LAN
Device name and number of your system's network adapter	Sys Admin	User
Network name and alias	Sys Admin	User
List of network hosts	Sys Admin	User
List of trusted hosts	Sys Admin	User

The following list describes in detail how to gather the necessary prerequisite information listed in Table 2-2. Before setting up your network, you must determine the following:

- Your system's Internet address

  If you are setting up a new LAN, you will have to determine an addressing scheme for all potential hosts on your LAN and decide whether or not you will request an Internet address from the NIC.

  If you are connecting your system to an existing LAN, request an Internet address from your system administrator or, if you are reinstalling your system, use the Internet address that you already have.

  What follows is a brief description of Internet addresses for those users who may be unfamiliar with networking.

  An Internet address consists of a total of 32 bits (four octets expressed by four fields of 1, 2, or 3 decimal numbers separated by periods), which together comprise the network address and the host address. The network address is assigned by the NIC; the host address is assigned locally.

  The NIC provides Internet addresses for three classes of networks, A, B, and C, depending on the number of systems you intend to connect to your LAN.

  In a class A network, the first field of the address is assigned by the NIC to designate the network, and the last three fields are left open for you to designate hosts on your network and subnet routing, if you are using subnet routing.

  In a class B network, the first two fields of the address are assigned by the NIC to designate the network, and the last two fields are left open for you to designate hosts on your network and subnet routing, if you are using subnet routing.

  In a class C network, the first three fields of the address are assigned by the NIC to designate the network, and the last field is left open for you to designate hosts on your network or subnet routing, if you are using subnet routing (generally, subnet routing is not used with Class C addresses).

  Table 2-3 illustrates the network fields assigned by the NIC and the fields available for designating hosts and, if applicable, subnetworks for all three network classes. The capital Xs surrounded by brackets ([XXX]) indicate the network fields assigned by the NIC.

Table 2-3: Internet Addresses

Network Class	Network Address Fields	Legal Network Addresses	Sample Internet Address
Class A	[XXX].XXX.XXX.XXX	[1-127]	15.0.0.200
Class B	[XXX].[XXX].XXX.XXX	[128-191][1-254]	179.140.0.200
Class C	[XXX].[XXX].[XXX].XXX	[192-223][0-255][1-254]	193.193.193.200

- Your Internet Protocol broadcast address

  If you are setting up a new LAN, you will have to determine – when you draw up your network plan – whether you will use all ones or all zeros for your broadcast address.

  Note

  All ones is now the industry standard for broadcast addresses. However, if you have any hosts on a network that require that the Internet Protocol broadcast address use all zeros, then all hosts on the LAN must use all zeros for their Internet Protocol broadcast address to preserve backward compatibility.

  If you are connecting your system to an existing LAN, this section offers a way to determine the broadcast address on your LAN. If you are unfamiliar with networking, request this information from your system administrator.

  The Internet Protocol broadcast address is used to send messages to all hosts on a network and therefore it must be the same for all hosts on that network. The Internet Protocol broadcast address consists of the NIC Internet address plus either the decimal numbers 255 (corresponding to binary 11111111) or 0 (corresponding to binary 00000000) in the remaining `host` fields. As a result, Internet Protocol broadcast addresses are referred to as being either all ones or all zeros, although decimal numbers appear in the actual address.

  Since some LANs may be using subnet routing, the simplest way to determine the broadcast address of an existing LAN is to look at the final host field.

  On a Class C network `193.193.193`, for example, a broadcast address of all ones (binary 11111111, expressed as decimal 255) would be `193.193.193.255`; on a Class A network `15` using subnet routing, a broadcast address of all ones (binary 11111111, expressed as decimal 255) would be `15.180.7.255`.

  If you are connecting your system to an existing LAN, you can see what the Internet Protocol broadcast address is on that network by looking in the `/etc/rc.local` on any system on that network on which you have an account.

To do this follow these steps:

1. Have your Internet address close at hand and, using Table 2-3, determine if the LAN you will be connecting to is a class A, B, or C network.
2. On another terminal or workstation, log in to a system that is connected to the same network that you will be connecting your system to.
3. To ensure that the system you have logged in to is connected to the same network that you will be connecting your system to, enter the following command on that system, replacing the italic *system_name* with the name of the system you are logged in to:

   ```
   % grep system_name /etc/hosts
   ```

   The `grep` command returns output like the following, listing the Internet address of the system you are logged in to in the first field, followed by the name of the system:

   ```
   15.180.5.166 gsamsa
   ```

   If the `grep` command returns nothing, enter the following command instead:

   ```
   % nslookup 'hostname'
   ```

   The `nslookup` command returns output like the following, listing the Internet address of the system you are logged in to after listing the entry for `localhost`:

   ```
   Server:  localhost.zk3.dec.com
   Address:  127.0.0.1

   Name:    gsamsa.ka.dec.com
   Address:  15.180.5.166
   ```

   Compare the network part of the Internet address returned by the `grep` or the `nslookup` command to the network part of your Internet address. If they match, you are logged in to a system on the correct network. In the preceding example, the network address is `15`.
4. Once you have determined that you are logged in to a system that is connected to the correct network, you can look at the LAN's broadcast address by entering the following command:

   ```
   % grep broadcast /etc/rc.local
   ```

   The `grep` command returns output like the following:

```
/etc/ifconfig ni0 '/bin/hostname' broadcast 15.255.255.255 netmask 255.254.0.0
```

   If the broadcast address uses the decimal numbers 255 in the last host field (corresponding to 11111111 binary), the broadcast address for that network is all ones; if the broadcast address uses zeros in the last host field (corresponding to 00000000 binary), the broadcast address for that network is all zeros.

   In the preceding example, because the decimal numbers 255 (corresponding to binary 11111111) appear in all the host fields of the broadcast address, `broadcast 15.255.255.255`, the broadcast address is all ones. Were the broadcast address to look like this, `broadcast 15.0.0.0`, the broadcast address would be all zeros (decimal 0 corresponding to binary 00000000).

- Whether your LAN is using subnet routing and, if so, how many bits of the host address will be used for subnets

  If you are setting up a new LAN, you must determine – when you draw up your network plan – if you will use subnet routing and how you will implement it.

  If you are connecting to an existing LAN, this section offers a way to determine whether your LAN is using subnet routing and how many bits of the host address are used for subnets. If you are unfamiliar with networking, request this information from your system administrator.

  Subnet routing enables numerous subnetworks to exist within a single Class A or Class B network.

  To determine subnet routing, the system uses a netmask to alter one or more of the host fields in your Internet address.

  Like an Internet address, a netmask has four fields, which correspond to the network and host fields of your Internet address.

  The network fields of the netmask are always set to all ones (binary 11111111, expressed as 255 decimal); the values placed in the remaining host fields of the netmask determine the subnet address of the network.

  If you are connecting your system to an existing LAN, you can determine if that network is using subnet routing and the number of bits that are being used for the subnet address by looking in the `/etc/rc.local` on any system on that network on which you have an account.

  To do this follow these steps:

  1. Have your Internet address close at hand and, using Table 2-3, determine if the LAN you will be connecting to is a class A, B, or C network.
  2. On another terminal or workstation, log in to a system that is connected to the same network that you will be connecting your system to.
  3. To ensure that the system you have logged in to is connected to the same network that you will be connecting your system to, enter the following command on that system, replacing the italic *system_name* with the name of the system you are logged in to:

     ```
     % grep system_name /etc/hosts
     ```

     The `grep` command returns output like the following, listing the Internet address of the system you are logged in to in the first field, followed by the name of the system:

     ```
     15.180.5.166 gsamsa
     ```

     If the `grep` command returns nothing, enter the following command instead:

     ```
     % nslookup 'hostname'
     ```

The `nslookup` command returns output like the following, listing the Internet address of the system you are logged in to after listing the entry for `localhost`:

```
Server:  localhost.zk3.dec.com
Address:  127.0.0.1

Name:    gsamsa.ka.dec.com
Address:  15.180.5.166
```

Compare the network part of the Internet address returned by the `grep` or the `nslookup` command to the network part of your Internet address. If they match, you are logged in to a system on the correct network. In the preceding example, the network address is `15`.

4. Once you have determined that you are logged in to a system that is connected to the correct network, you can look at the netmask for that LAN by entering the following command:

```
% grep netmask /etc/rc.local
```

The `grep` command returns output like the following:

```
/etc/ifconfig ni0 `/bin/hostname` broadcast 15.255.255.255 netmask 255.255.254.0
```

If the netmask returned by the `grep` command is one of the following for your network class, then the LAN you are connecting to does not use subnet routing:

Network Class	Netmasks with No Subrouting
Class A Network	netmask 255.0.0.0
Class B Network	netmask 255.255.0.0
Class C Network	netmask 255.255.255.0

Any other netmask indicates that subnet routing is used on that LAN.

5. To determine the number of bits being used for subnet routing, first convert the decimal numbers in the host fields of the netmask into binary numbers by using the following table (the table lists the valid decimal values for the host fields of the network mask):

Decimal Value	Binary Equivalent	Number of Valid Subnet Bits
255	11111111	8
254	11111110	7
252	11111100	6
248	11111000	5
240	11110000	4
224	11100000	3
192	11000000	2
128	10000000	1
0	00000000	0

Add together the number of binary ones in each host field of the netmask. This is the number of bits being used on that LAN for subnet routing.

For example, on a Class A network with a netmask of `255.255.254.0`, 15 bits are being used for the subnet address, (255=11111111=8 bits) + (254=11111110=7 bits) = 15 bits.

On a Class B network with the same netmask, `255.255.254.0`, 7 bits are being used for the subnet address, 254=11111110=7 bits.

- The device name of the network interface and its unit number

  Whether you are setting up a new LAN or connecting to an existing LAN, you must determine the device name and unit number of the network interface you are configuring.

  Table 2-4 lists some of the more common interface names.

Table 2-4: Common Interface Names

Name	Controller	Machine
Ethernet		
de	DEUNA or DELUA	VAX
ln	DESVA	VAXstation or DECstation
ni	DEBNA	VAX
qe	DEQNA or DELQA	MicroVAX
xna	DEBNI or DEMNA	VAXstation or DECstation
se	SGEC	DECstation 5500
FDDI		
fza	DEFZA	DECstation 5000 series
Point-to-Point		
dmc	DMR-11 or DMC-11	VAX
dmv	Q-bus	MicroVAX
sl	Serial line IP	VAXstation or DECstation

You can determine the network interfaces on your system by running the `netstat` command with the `i` option. To do this follow these steps:

1. Log in as `root` or become superuser.
2. Enter the following command:

```
# netstat -i
```

The system returns output like the following:

```
Name    Mtu   Network     Address     Ipkts Ierrs    Opkts Oerrs  Coll
ln0*    1500  none        none            0     0        0     0     0
lo0*    1536  none        none            0     0        0     0     0
```

The device name and unit number of the network adapters appear under the leftmost column, entitled "Name." In this example, the device name of the network adapter is `ln` and the unit number is `0`.

Note

The `lo0` device, known as localhost, allows each system to simulate a network and is not one of the network adapters.

- Your network name or alias

  If you are setting up a new LAN, you must determine – when you draw up your network plan – what name and alias you will give that network.

  The `netstat` command uses the name and alias you specify to translate the network address to the network name and alias. If you do not specify a network name, `netsetup` provides the network interface (`ethernet`, for example), as the default network name.

  A class C network without a network name would appear in the `/etc/networks` file as follows:

  ```
  ethernet 193.193.193
  ```

  If, however, that network had a name and an alias associated with it, users could identify where the network connects to and would not have to remember the Internet address, because the network name, the Internet address, and the alias are all equivalent.

  If you are connecting to an existing LAN, this section offers a way to determine the name and alias of that LAN. If you are unfamiliar with networking, request this information from your system administrator.

  If you are connecting your system to an existing LAN, you can determine the name and alias of that LAN by looking in the `/etc/networks` file or the Yellow Pages `networks` map on any system on that network on which you have an account.

  To do this follow these steps:

  1. Have your Internet address close at hand and, using Table 2-3, determine if the LAN you will be connecting to is a class A, B, or C network.
  2. On another terminal or workstation, log in to a system that is connected to the same network that you will be connecting your system to.
  3. To ensure that the system you have logged in to is connected to the same network that you will be connecting your system to, enter the following command on that system, replacing the italic *system_name* with the name of the system you are logged in to:

     ```
     % grep system_name /etc/hosts
     ```

The `grep` command returns output like the following, listing the Internet address of the system you are logged in to in the first field, followed by the name of the system:

```
15.180.5.166 gsamsa
```

If the `grep` command returns nothing, enter the following command instead:

```
% nslookup 'hostname'
```

The `nslookup` command returns output like the following, listing the Internet address of the system you are logged in to after listing the entry for `localhost:`

```
Server:  localhost.zk3.dec.com
Address:  127.0.0.1

Name:    gsamsa.ka.dec.com
Address:  15.180.5.166
```

Compare the network part of the Internet address returned by the `grep` or the `nslookup` command to the network part of your Internet address. If they match, you are logged in to a system on the correct network. In the preceding example, the network address is `15`.

4. Once you have determined that you are logged in to a system that is connected to the correct network, you can display the name and alias of the LAN by doing the following:

   a. Using the `grep` command, search the `/etc/networks` file for the network you are connecting to, replacing the italic *network_number* with your network number:

      ```
      % grep network_number /etc/networks
      ```

      The system should return output like the following:

      ```
      lab_1 15.45.45 engineering
      ```

   b. If no output is returned, enter the following command, replacing the italic *network_number* with your network number:

      ```
      # ypcat networks | grep network_number
      ```

      The system returns output like the following:

      ```
      lab_1 15.45.45 engineering
      ```

      The name of the LAN is in the first field, the alias in the third.

- The names and addresses of other hosts on the LAN

  If you are setting up a new LAN, you will have to determine – when you draw up your network plan – what hosts will be connected to that LAN.

  If you are connecting your system to an existing LAN and do not intend to run BIND/Hesiod or YP, once `netsetup` is finished and the network is established, you can do one of the following, rather than add each host on the network separately using `netsetup`:

  – If you are reinstalling and have saved your original `/etc/hosts` file, you can restore it after running `netsetup`.

  – If you are installing for the first time or you are reinstalling and have not saved your original `/etc/hosts` file, you can copy a complete

/etc/hosts file from a server on the network provided that the server is not running BIND/Hesiod exclusively.

- The names of trusted hosts

  Whether you are setting up a new LAN or connecting to an existing LAN, you may want to designate certain hosts on your LAN as trusted hosts.

  Trusted hosts are listed in the /etc/hosts.equiv file. Systems listed in the /etc/hosts.equiv file are logically equivalent to, and therefore treated exactly the same as, the local system.

  Placing an entry in the /etc/hosts.equiv file allows all users (except for root) on a remote system who also have accounts on your system to log in to your system without supplying a password. This means that the logically equivalent system has all of the privileges of the local system.

  However, the equivalency is not automatically reciprocal. If the system host1 specifies host2 in its /etc/hosts.equiv file, host2 is logically equivalent to host1. But unless host2 specifies host1 in its /etc/hosts.equiv file, host1 is not considered logically equivalent to host2.

  For security reasons, logically equivalent systems are usually run by the same administration.

  See the hosts.equiv(5) reference page for more information.

Steps

These steps explain how to set up your network using netsetup. Read through these steps before running netsetup to ensure that you have all of the information that you need and then refer to this section as you execute the program.

Note that you should have the Internet address broken up into a separate network and host address, because you will have to enter each address separately.

1. Log in as root or become superuser.
2. To invoke netsetup, enter the following command:

   ```
   # netsetup install
   ```

   The install option tells the system that you are either installing a new LAN or connecting to an existing LAN.

   Warning

   If your system is already connected to a LAN and you want to connect your system to an additional LAN, you cannot use netsetup. The netsetup program invoked with the install option overwrites all previous network configurations in all the network files.

   If you are already connected to a LAN and will be connecting to multiple LANs, see the section "Setting up a Router" in this chapter.

3. Verify your system's name and, optionally, specify any abbreviations by which you want your system known.

The netsetup command provides as the default the name that you specified for your system at installation. Following some informational text about abbreviations, it prompts you to specify one or more names and abbreviations for your system.

```
Your system's name is ''host1''.  Is this correct [yes]?
```

4. Enter the network address that was assigned to you by the NIC or by your system administrator.

   For example, on a Class B network 179.140.254.200 with subnet routing, you would enter `179.140`

5. Indicate whether your network is using subnet routing. Answer either `yes` or `no`.

6. Enter the host address.

   The host address must be unique to that host. The netsetup command asks you to include the subnet number. Note that the host number on a LAN using subnet routing already contains the subnet number.

   For example, on a Class B network 179.140.254.200 with subnet routing, you would enter `200`

   The netsetup command then updates the appropriate system files with the information you have provided, displaying a message similar to the following:

   ```
   ***** UPDATING /etc/hosts WITH host1 AND localhost *****
   ```

7. If you answered `yes` in step 5 (you are using subnet routing), the netsetup command asks how many bits to use for specifying subnetworks. Enter the number of bits your LAN is using for subnet routing.

   The netsetup command then determines the appropriate netmask for your system (which must be the same for all systems on your LAN) based on the information you provide.

8. Specify whether to use all zeros or all ones for the Internet Protocol broadcast address.

   The industry standard default is all ones. If you are setting up a LAN for the first time, use all ones.

   The netsetup command then determines the appropriate broadcast address for your system (which must be the same for all systems on your LAN) based on the information you provide.

9. Specify the device name and unit number of your network interface.

   The netsetup command then updates the appropriate system files with the information you have provided, displaying a message similar to the following:

```
**UPDATING /etc/rc.local WITH network configuration information**
```

10. Specify a network name for your network address and any aliases for the network name.

    The netsetup command then updates the `/etc/networks` file with the name of the network displaying a message similar to the following:

    ```
    ** UPDATING /etc/networks WITH doconet **
    ```

    In this example, the network name is `docnet`.

11. Enter the host name, abbreviations, network address, and host address for each host on the network.

    The information you supply is used to update the `/etc/hosts` file. Add the names of key hosts on your network to the `/etc/hosts` file, regardless of whether you intend to use BIND/Hesiod or Yellow Pages to distribute the hosts database on your network. See the the *Guide to the Yellow Pages Service* and the *Introduction to Networking and Distributed System Services* for information on distributing databases in a networked environment.

    Note for Workstation Users

    If you are connecting your system to an existing LAN and do not intend to run BIND/Hesiod or YP, once `netsetup` is finished and the network is established, you can do one of the following, rather than add each host on the network separately using `netsetup`:

    - If you are reinstalling and have saved your original `/etc/hosts` file, you can restore it after running `netsetup`.
    - If you are installing for the first time or you are reinstalling and have not saved your original `/etc/hosts` file, you can copy a complete `/etc/hosts` file from a server on the network provided that the server is not running BIND/Hesiod exclusively.

      To set up a fully populated `/etc/hosts` file quickly, withoutusing `netsetup`, follow these steps.

      Note

      These directions assume that you are an experienced user familiar with working in a networked environment. If you are in any way uncertain about the procedure described here, use the `netsetup` command instead.

      a. Enter the Internet address and hostname of only one server on your LAN so that the `netsetup` utility will place that entry in your `/etc/hosts` file.

         If you have used the Remote Installation Service (RIS) to install your workstation, the Internet address of the RIS server is already in your `/etc/hosts` file.

      b. Complete Steps 12 and 13, and exit the `netsetup` program. After the `netsetup` program completes and you have started the network, continue with step c.

      c. To enable you to copy the `/etc/hosts` file from a server, using the text editor of your choice, edit the `/.rhosts` file and place in it the name of your system and the name of the server from which you will copy the fully populated `/etc/hosts` file.

         You must also ensure that the `/.rhosts` file on the server has the name of your system in it.

d. To copy a fully populated /etc/hosts file from the server to your system, first determine if the server is running Yellow Pages by entering the following command, substituting the name of the server for the italic *server* in the example:

```
# rsh server ps agx | grep yp
```

If the rsh command returns nothing, the server is not running Yellow Pages. If the system returns output like the following, then the server is running Yellow Pages and has a distributed /etc/hosts file:

```
2273 ?  I      1:57 /usr/etc/ypserv
2277 ?  S      0:13 /etc/ypbind
```

– To acquire a fully populated /etc/hosts file for your system from a server running Yellow Pages, enter the following command, substituting the name of the server for the italic *server* in the example. Note that you must use two right angle brackets (>>) to append the hostname entry to the /etc/hosts.all file. One right angle bracket (>) will overwrite the file.

```
# rsh server ypcat hosts > /etc/hosts.all
# cp /etc/hosts /etc/hosts.orig
# grep `hostname` /etc/hosts.orig >> /etc/hosts.all
# mv /etc/hosts.all /etc/hosts
```

– To acquire a fully populated /etc/hosts file for your system from a server that is not running Yellow Pages, first determine if the server is running BIND/Hesiod by entering the following command, substituting the name of the server for the italic *server* in the example:

```
# rsh server ps agx | grep named
```

- If the command returns output like the following, then the server is running BIND/Hesiod and does not have a hosts file that you can copy:

```
1969 ?  I      4:59 /usr/etc/named /var/dss/namedb/named.boot
```

- If the command returns nothing, you can copy an /etc/hosts file from that server by entering the following commands, substituting the name of the actual server for the italic *server* in the example. Note that you must use two right angle brackets (>>) to append the hostname entry to the /etc/hosts.all file. One right angle bracket (>) will overwrite the file.

```
# rcp server:/etc/hosts /etc/hosts.all
# cp /etc/hosts /etc/hosts.orig
# grep `hostname` /etc/hosts.orig >> /etc/hosts.all
# mv /etc/hosts.all /etc/hosts
```

e. If you so desire, using the text editor of your choice, edit the /etc/hosts file and move the entry for your system to a more appropriate place.

12. Enter the names of trusted hosts.

    Users on the trusted host who have a valid account on your machine can log in to your machine without supplying a password. Designate trusted hosts with care.

    The `netsetup` command then updates the appropriate system files with the information you have provided, displaying a message similar to the following:

```
***** SETTING UP /usr/hosts DIRECTORY *****
***** NETWORK SETUP COMPLETE *****
```

13. Reboot your system.

    Use the `shutdown` command with the `-r` option to reboot. The following command immediately performs an orderly shutdown and automatic reboot:

```
# shutdown -r now
```

Note

If you are continuing with setup tasks and do not want to reboot your system, you can start up the network manually by entering the following commands, replacing the italic *system_name* with the name of your system:

```
# hostname system_name
# grep config /etc/rc.local | sh
```

See Also

`hosts`(5), `hosts.equiv`(5), `networks`(5), `ifconfig`(8), `netsetup`(8), `netstat`(8)

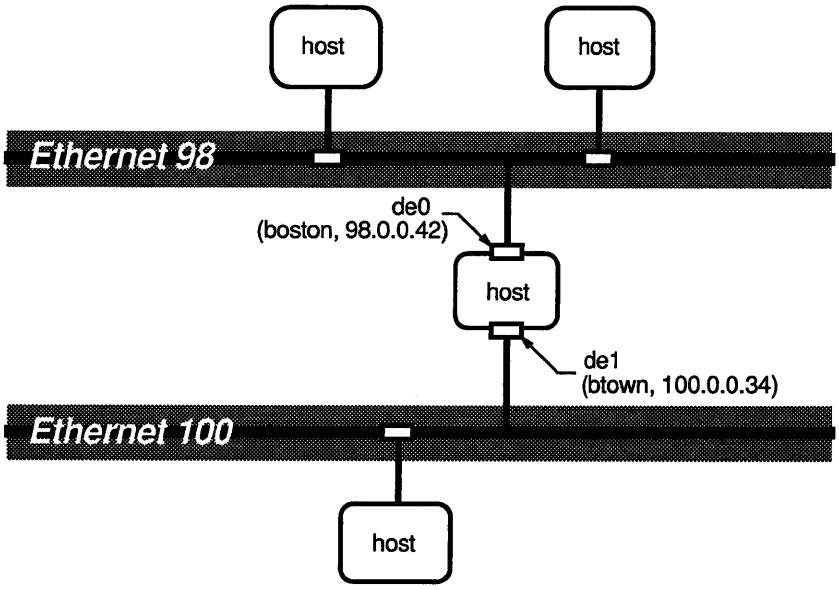
Introduction to Networking and Distributed System Services

Setting Up and Accessing a Router

Routers (commonly known as gateways) are hosts that are connected to multiple LANs. They have a network interface for each LAN to which they are connected, and each network interface is assigned a unique host name and Internet address. Because it is connected to multiple LANs, a router allows data to be transferred between systems on the LANs to which it is connected.

The following figure shows two LANs interconnected by a router. The router host has network interfaces for each of the networks it is connected to. On network 98 the router is known as `boston` and has an Internet address of 98.0.0.42. On network 100 the router is known as `btown` and has an Internet address of 100.0.0.34.

Figure 2-1: Setting Up a Router



ZK-0178U-R

For more information on routers, see the *Introduction to Networking and Distributed System Services.*

Before You Start

For historical reasons, `netsetup` invoked with the `install` option overwrites all previous network configurations in all the network files. As a result, to set up a router you must perform all of the tasks manually. Essentially this entails editing the following files:

- `/etc/rc.local`
- `/etc/networks`
- `/etc/hosts`
- `/etc/hosts.equiv`

While a sophisticated user can set up a router on any system that has more than one network interface, setting up a router is a non-trivial task that requires an understanding of Internet addresses, netmasks, and subnetworks.

For a discussion of TCP/IP LANs, Internet addresses, and subnetworks, see the *Introduction to Networking and Distributed System Services.*

Note

Any networks that you intend to interconnect with a router must be up and running.

Gathering Prerequisite Information

Table 2-5 lists the prerequisite information that you will need to gather to set up and access a router and shows whether the user, system administrator, or the NIC is able to provide that information. The term user refers to anyone who has `root` privilege on a workstation and is not a system administrator.

Note that all information that the user can determine can also be requested from the system administrator. If you have any doubts about being able to determine any of the information marked as User in the following table, request that information from your system administrator.

Table 2-5: Who Can Provide Prerequisite Information

Router Information	Setting Up a Router	Accessing an Existing Router
Device name and number of your system's network adapter	User	–
Pseudo-hostname of your system	Sys Admin	–
Internet address (network and host addresses)	NIC/Sys Admin	–
Broadcast address (new network)	User	–
Network name and alias (new network)	User	User
List of network hosts (new network)	User	User
List of trusted hosts (new network)	User	User

The following list describes in detail how to gather the necessary prerequisite information listed in Table 2-5. Before setting up your router, you must determine the following:

- Whether the appropriate network interfaces are installed and, if so, the device name and unit number of the network adapter you will be using to connect your system to the new network

  Your system must have a network interface for each network to which it will be connected. To see what interfaces are available to be configured, run the `netstat` command with the `-i` option, as follows:

  1. Log in as `root` or become superuser.
  2. Enter the following command:

     ```
     # netstat -i
     ```

     The system will return output like the following:

```
Name    Mtu   Network     Address            Ipkts Ierrs    Opkts Oerrs  Coll
ln0     1500  boston      kafka          120384139   357 28149584 5081  128877
ln1*    1500  none        none                   0     0        0     0     0
lo0     1536  loop        localhost              0     0        0     0     0
```

     In this example, there are two network adapters: `ln0`, which is already connected to the network `boston`, and `ln1`, which is available to be connected to another network.

Note

The `lo0` device, known as localhost, allows the system to simulate a network and is not one of the network adapters.

- The pseudo-hostname and corresponding Internet Protocol address of your system

  Have the system administrator assign a unique name (pseudo-hostname) and address to the new network interface and edit the `/etc/hosts` file with the pseudo-hostname and address.

  Although a router is one physical machine, it functions as multiple hosts. Each network interface is assigned a unique hostname and Internet address in accordance with the numbering and address scheme of the network to which it is directly connected.

  For example, a system that is a router between networks 98 and 100 can be known on network 98 as `boston` and have an Internet address of 98.0.0.42. Its pseudo-hostname on network 100 can be `btown` with an Internet address of 100.0.0.34.

  For information on how Internet Protocol addresses are assigned, see page 2-4 .

- The Internet Protocol broadcast address of the new LAN

  For information on how to determine the Internet Protocol broadcast address of an existing LAN, see page 2-5 .

- The network name or alias of the new LAN

  For information on how to determine your network name or alias, see page 2-10.

- The names and addresses of other hosts on the new LAN

  For information on how to determine the names and addresses of other hosts on a LAN, see page 2-11.

- The names of trusted hosts on the new LAN

  For information on how to determine the names of trusted hosts, see page 2-12.

Steps

These steps explain how to set up and access a router. Read through these steps before setting up or accessing a router to ensure that you have all of the information that you need.

Setting Up a Router

1. Log in as `root` or become superuser.
2. Edit the `/etc/rc.local` file and search for the `broadcast` entry. Your text editor should place the cursor on a line that has the following format:

/etc/ifconfig *device-name-number* **'/bin/hostname' broadcast** *x.x.x.x* **netmask** *y.y.y.y*

   Copy this line and make the following edits to the copied line:

   a. Replace *`device-name-number`* with the name and number of the network device you will be using to connect to the new network.

   b. Replace *`hostname`* with the pseudo-hostname associated with the new network interface.

   c. Replace *`x.x.x.x`* with the Internet broadcast address of the new network you are connecting to.

   d. Replace *`y.y.y.y`* with the netmask of the new network you are connecting to.

   Ensure that the new line is between the original `ifconfig` line and the line for `localhost`. For example, on the system `boston` shown in Figure 2-1, the `/etc/rc.local` file would look like this after being edited to include an entry for the secondary network interface:

```
/etc/ifconfig de0 boston broadcast 98.255.255.255 netmask 255.0.0.0
/etc/ifconfig de1 btown broadcast 100.255.255.255 netmask 255.0.0.0
/etc/ifconfig lo0 localhost
```

   Verify that you have entered the information correctly.

3. In the same file, `/etc/rc.local`, enable the `routed` daemon so that it will be started automatically every time the system is brought to multiuser mode.

   The `routed` daemon periodically updates the internal routing tables and provides information on routing and accessibility among machines on each LAN. The following entry for `routed` is included in the `/etc/rc.local` file by default:

```
# if [ -f /etc/routed ]; then
#       /etc/routed & echo -n ' routed'   >/dev/console
# fi
```

   Search for `routed` and then remove the comment characters (#).

   Verify that you have edited the `/etc/rc.local` file correctly, then write and quit the file.

4. Edit the `/etc/networks` file to include the name, network number, and alias of the additional network connected to your router.

For example, on the system `boston` referred to in Figure 2-1, an edited `/etc/network` file that includes an entry for the additional network 100 would look like this:

```
#
# Internet networks
#
loop                127     loopback
engineering_net      98     engin
service_net         100     serve
```

The network name is in the first field, the network number in the second field, and the network alias in the third field. All are equivalent.

For more information on network names and aliases, see the subsection "Gathering Prerequisite Information" in the section "Setting Up a Network."

Note

If you choose to run either Yellow Pages (YP) or the BIND/Hesiod naming service, the `networks` database is distributed. You need only edit the `networks` database on the YP or BIND/Hesiod master server, and the information is distributed to all other servers and client systems. For more information on distributing databases with BIND/Hesiod, see the *Guide to the BIND/Hesiod Service.* For more information about distributing databases with YP, see the *Guide to the Yellow Pages Service.*

5. Reboot your system to invoke the `routed` daemon.

   If your system is a standalone workstation, enter the `w` command to determine if any other users besides yourself are logged into your system, as follows:

   ```
   # w
   ```

   If you are the only user logged into your system, the `w` command will return data like the following, listing your login name in the first field:

```
eddie   p0 :0.0          1:04pm                    view fixXtm2d
eddie   p1 :0.0          1:04pm     3              -csh
eddie   p2 :0.0          1:04pm     2              -csh
eddie   qf               1:03pm                    -
```

   If you are the only user logged into your system, reboot your system by entering the following command:

   ```
   # /etc/shutdown -r now
   ```

If other users are logged into your system, the w command will return data like the following, listing the users' login names in the first field:

```
tony     p0 samsa        9:24am  3:31     10         -sh
tony     p1 samsa        9:24am  1:33      7         -sh
john     p2 badlands     12:47pm            2      1  -csh
eddie    p3 kafka        1:04pm                       w
```

If other users are logged into your system, notify them that you are rebooting your system by entering the following command:

```
# /etc/shutdown -r +15 Rebooting to start router.
```

If your system is a server or a time-sharing system, notify the clients or users that you are rebooting the system by entering the following command:

```
# /etc/shutdown -r +15 Rebooting to start router.
```

Note

If you are continuing with setup tasks and do not want to reboot your system, you can start up the `routed` daemon manually by entering the following command:

```
# /etc/routed
```

Accessing a Router

To access a new router, system administrators of servers or workstation users with `root` privilege for each host on the networks to which the router is connected must do the following:

1. Edit the `/etc/networks` file to include the name, network number, and alias of the additional network you want to access through the router. See step 4 in the preceding section " Setting Up a Router ."
2. Enable the `routed` daemon by editing the `/etc/rc.local` file. See step 3 in the preceding section " Setting Up a Router ."
3. Reboot your system or invoke the `routed` daemon manually. See step 5 in the preceding section " Setting Up a Router ."

Note

If you choose not to run the `routed` daemon, you can add a new route using the `/etc/route` command. The syntax for the `/etc/route` command is as follows:

/etc/route { **add** | **delete** } [*net* | *host*] *destination gateway* [*metric*]

In this syntax, *destination* is the host or network that the route connects to, *gateway* is the gateway to which packets are to be addressed, and *metric* is an optional count indicating the number of hops to the destination. The metric is required for add commands. It must be zero if the destination is on a directly attached network, and nonzero if the route utilizes one or more gateways.

For example, to access the router `boston` (98.0.0.42) shown in Figure 2-1 from a host on network 100, you would enter a command like the following:

```
# /etc/route add default 98.0.0.42 1
```

If you have not enabled the `routed` daemon in the `/etc/rc.local` file, any routes you add using the `/etc/route` command are only effective until the system is rebooted. See the *Introduction to Networking and Distributed System Services* and the `route`(8c) reference page for more information on adding routes manually.

See Also

`netstat`(1), `ifconfig`(8c), `route`(8c), `routed`(8c)

Introduction to Networking and Distributed System Services
Guide to the BIND/Hesiod Service
Guide to the Yellow Pages Service

Modifying the SNMP Agent with snmpsetup

The Simple Network Management Protocol (SNMP) is the de facto industry standard for managing TCP/IP networks. The protocol defines the role of a Network Management Station (NMS) and an SNMP Agent, allowing remote users on an NMS to monitor and manage TCP/IP network entities.

The `/etc/snmpd.conf` file is the configuration file for the SNMP daemon, `snmpd`.

The `/etc/snmpd.conf` file permits system and network administrators running NMS software to query other hosts on the network for various network parameters that are set in the file.

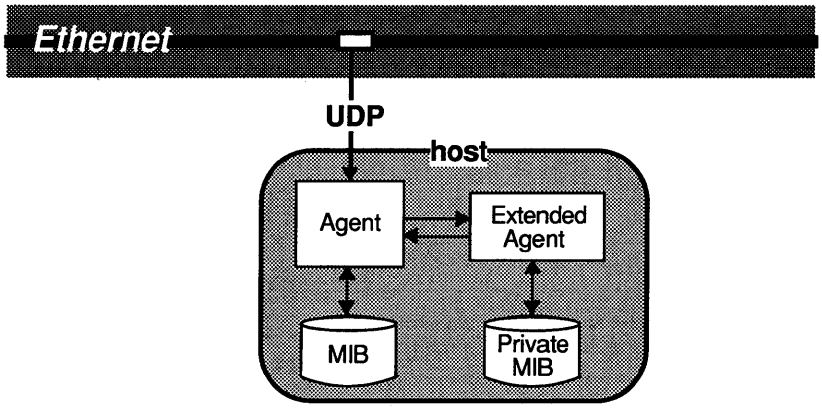
The `/etc/snmpd.conf` file is installed and configured with a default public community at installation time, and, for the most part, does not need to be modified by the user.

In most cases, system and network administrators will modify a `/etc/snmpd.conf` file and then have it installed on all of the network clients they want to monitor with NMS software.

This section explains how to modify the default `/etc/snmpd.conf` file.

The following figure shows a host running the SNMP Agent software. It shows the relationship between the Agent and the Management Information Base (MIB), and the relationship between the Agent and the Extended Agent. The NMS (not shown) can be located either on the same network or on a different one.

Figure 2-2: Setting Up the SNMP Agent



ZK-0181U-R

Note

The ULTRIX operating system supports an implementation of the SNMP Agent. It does not implement the NMS software.

Gathering Prerequisite Information

Table 2-6 lists the prerequisite information that you will need to gather to complete `snmpsetup` and shows whether the user, system administrator, or the NIC is able to provide that information. The term user refers to anyone who has `root` privilege on a workstation and is not a system administrator.

Note that all information that the user can determine can also be requested from the system administrator. If you have any doubts about being able to determine any of the information marked as User in the following table, request that information from your system administrator.

Table 2-6: Who Can Provide Prerequisite Information

snmpsetup Information	Modifying an SNMP Agent
Device name and number of your system's network adapters	Sys Admin
Pseudo-hostname of your system	Sys Admin
Community Name and Type	Sys Admin
Internet address for Your Community	NIC/Sys Admin
Whether you are including user-written Extended Agents	Sys Admin

The following list describes in detail how to gather the necessary prerequisite information listed in Table 2-6. Before setting up your system as an snmp agent, you must determine the following:

- The names of your system's network interfaces

  Run the `netstat` command with the `-i` option to see what network interfaces are available to be configured.

  Simply follow these steps:

  1. Log in as `root` or become superuser.
  2. Enter the following command:

     ```
     # netstat -i
     ```

     The system will return output like the following:

```
Name    Mtu   Network     Address            Ipkts Ierrs    Opkts Oerrs  Coll
ln0     1500  boston      kafka          120384139   357 28149584 5081  128877
ln1*    1500  none        none                   0     0        0     0     0
lo0     1536  loop        localhost              0     0        0     0     0
```

     This example shows two network adapters: `ln0`, which is already connected to the network `boston`, and `ln1`, which is available to be connected to another network.

Note

The `lo0` device, known as localhost, allows the system to simulate a network and is not one of the network adapters.

See the `netstat`(8) reference page for more information.

- Your community name and type

  The community name can be any character string up to 127 characters long.

  Communities can be read-only, read-write, or traps. Read-only communities can be monitored but not managed by an NMS. Read-write communities can be managed by the NMS. Traps are unsolicited messages generated by the Agent that guide the polling from the NMS.

- The Internet address that you want associated with the community

  You will need to specify the Internet address of any NMS that you want to be able to manage or monitor your system.

- Whether you are including user-written Extended Agents

  If you are defining Extended Agents, you must specify the full pathname of the extended agent and its name.

  See the *Guide to Network Programming* for information on defining Extended Agents, and the directory `/usr/examples/snmp/snmpext` for the files necessary to build an example `snmpextd` daemon.

Note

Your network must be up and running before you attempt to set up SNMP.

Steps

This section describes the steps you take to run the `snmpsetup` program. Read through these steps before running `snmpsetup` to ensure that you have all the information you need and then refer to this section as you execute the program.

Note

To terminate `snmpsetup` with no modifications to the `/etc/snmpd.conf` file, press Ctrl/C.

1. Log in as `root` or become superuser.
2. Enter the following command:

   ```
   # snmpsetup
   ```

3. Press the Return key to accept the default `sysDescr` parameter.

   The default `sysDescr` parameter is the system name that you specified at installation. Digital recommends that you change the default `sysDescr` parameter only if you want to add system hardware and software information.

4. Add network interfaces that are not automatically configured by the SNMP Agent.

The SNMP Agent checks the system configuration file and automatically configures the following interfaces, if they are present: de, ln, lo, ni, qe, scs, and xna. No explicit entries for these interfaces appear in the `/etc/snmpd.conf` file.

If you are configuring a network interface other than one of the ones listed, `snmpsetup` prompts you for information about the interface, and then edits the `/etc/snmpd.conf` file with the appropriate information.

The following example shows how to add a serial-line (sl0) interface, and how to specify the interface type (`ifType`) and interface speed (`ifSpeed`):

```
Do you wish to add network interfaces [n]? y
Enter new interface name (ifName)? sl0
Enter interface type (ifType) [6]? 1
Enter interface speed (ifSpeed) [10000000]? 9600
```

The `ifType` parameter indicates the code number for the proper interface hardware type. The value of 1, specified in this example, indicates that serial-line interfaces belong in the category `other`. See the `snmpd.conf`(5) reference page or RFC 1066 under the `ifType` object definition for more information on coding interface hardware types.

The `ifSpeed` parameter is an estimate of the interface's current bandwidth in bits per second. The default, 10000000, is appropriate for an Ethernet interface. In this example, the serial line is configured to run over a modem at 9600 baud. See your hardware manual for information on the speed of data transmission for your interface.

5. Specify the community name, Internet address of the NMS, and community type for communities that you want to add to the `/etc/snmpd.conf` file. The community information is mandatory and must be configured.

   The community name is used by the SNMP protocol to authenticate requests from an NMS. The Internet address is the Internet address of the NMS that is allowed to monitor or manage your system. If you specify an Internet address of 0.0.0.0, any NMS can monitor your system. The community type can be read-write, read-only, or traps.

   The following example shows how to set up a read-only and a read-write community:

```
Enter community name? test1
Enter IP address associated with community test1 [0.0.0.0]?128.45.10.100
Select community type (read-only,read-write, traps) [read-only]? Return
Do you wish to add another community [n]?y
Enter community name? testwrite
Enter IP address associated with community testwrite [0.0.0.0]?128.45.12.105
Select community type (read-only,read-write,traps) [read-only]?read-write
```

   The community `test1` is a read-only community that allows the NMS whose Internet address is 128.45.10.100 to monitor it. The community `testwrite` is a read-write community that allows the NMS whose Internet address is 128.45.12.105 to both monitor it and set variables for it.

6. Press the Return key when prompted to configure a public read-only community, if you have not configured any other communities.

   You must configure a public read-only community if you have not defined any other communities. If you do not have any community names configured, SNMP will not work on your system.

7. Specify the full pathname and name of any user-written Extended Agents, if you are adding any.

   The *Guide to Network Programming* has information on defining Extended Agents. Follow the steps described there before specifying Extended Agents using `snmpsetup`.

After you have finished answering the questions about Extended Agents, `snmpsetup` exits.

See Also

`snmpd.conf`(5), `netstat`(8), `snmpd`(8), `snmpsetup`(8)

Introduction to Networking and Distributed System Services
Guide to Network Programming
RFC 1065—*Structure and Identification of Management Information for TCP/IP-based internets*
RFC 1066—*Management Information Base for Network Management of TCP/IP-based internets*
RFC 1098—*A Simple Network Management Protocol (SNMP)*

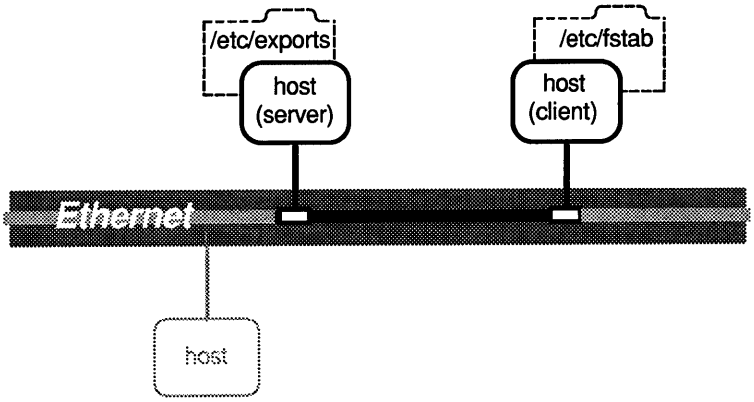
Setting Up the Network File System with nfssetup

The Network File System (NFS) is a facility for sharing files in a heterogeneous environment. It is based on the client/server model: an NFS server is a machine that exports file systems, an NFS client is a machine that imports file systems. In most cases, an NFS server will both import and export files, while a workstation client will usually only import files.

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

The following figure shows an NFS server and an NFS client attached to a LAN. It indicates the system files that are modified when you run the `nfssetup` command. For hosts that are servers, the `/etc/exports` file is modified. For hosts that are clients, the `/etc/fstab` file is modified. The same host can be a server for some file systems and a client for others.

Figure 2-3: Setting Up the Network File System



ZK-0180U-R

Note

Your network must be up and running before you attempt to set up NFS.

Gathering Prerequisite Information

Table 2-7 lists the prerequisite information that you will need to gather to complete `nfssetup` and shows whether the user or system administrator is able to provide that information. The term user refers to anyone who has `root` privilege on a workstation and is not a system administrator.

Note that all information that the user can determine can also be requested from the system administrator. If you have any doubts about being able to determine any of the information marked as User in the following table, request that information from your system administrator.

Table 2-7: Who Can Provide Prerequisite Information

nfssetup Information	Setting Up the Network File System	Connecting to an Existing Network File System
The role each host will play in the distributed environment	Sys Admin	Sys Admin/User
Whether to enable NFS locking	Sys Admin	Sys Admin
For NFS clients, the number of of block I/O (`biod`) daemons to run	Sys Admin	User
For NFS servers, the number of `nfsd` daemons to run	Sys Admin	User
Whether to run the `rwalld` daemon	Sys Admin	User
For NFS servers, the directory pathnames you want to export and the host names of the client machines	Sys Admin	User
For NFS clients, the host names of the servers from which you will be importing directories, the directory pathnames of the directories you will be importing, and the mount points where the imported directories will reside	Sys Admin	User

The following list describes in detail how to gather the necessary prerequisite information listed in Table 2-7. Before setting up the network file system, you must determine the following:

- The role each host will play in the distributed environment

  You should determine whether your system will be an NFS server, NFS client, or both. Note that in a networked environment, workstations are usually configured as NFS clients, receiving files from an NFS server.

- Whether you want NFS locking enabled

  File locking allows you to create advisory locks on local and remote files, and file regions. Locking prevents multiple users from editing the same file simultaneously. Advisory locking, however, is not enforced. For file locking to work, it must be enabled on all clients and servers. For more information on file locking, see the *Guide to the Network File System* and the `fcntl`(2) and `lockf`(3) reference pages.

- For NFS clients, the number of block I/O (`biod`) daemons to run

  Digital recommends that you configure 4 `biod` daemons. Running more than 4 `biod` daemons causes network congestion.

- For NFS servers, the number of `nfsd` daemons to run

  The default number of 4 is adequate for an average workload on a workstation. Server systems should configure between 12 and 20 `nfsd` daemons, depending on their workload. The maximum number of `nfsd` daemons configurable with `nfssetup` is 20.

- Whether to run the `rwalld` daemon

  The `rwalld` daemon sends a broadcast message to clients when a server is shutting down using the `shutdown` command. If the server crashes, or is brought down with the `halt` or `reboot` command, `rwalld` does not send a broadcast message to clients.

- For NFS servers, the directory pathnames of the directories that you want to export, and the host names of the machines to which you plan to export these directories

  If you want to limit what hosts can import a file system, you must specify the individual hosts or network groups explicitly in the `/etc/exports` file. If you do not specify individual hosts or network groups, all hosts can import that file system. For information on defining network groups, see the the `netgroup`(5yp) reference page and the *Guide to the Yellow Pages Service.*

- For NFS clients, the names of the remote hosts from which you are importing directories, the complete directory pathnames of the directories that you want to import, the local mount points where you want the imported directories to reside, and whether the imported file system should be read-only or read-write

  The default permission is read-only.

Note

If you mount a file system on a client with read-write permissions (for example, a home directory), the user identification number (UID) and the group identification number (GID) for the owner of the file system on the server and the client must be the same. If they are not, the user cannot modify any of the mounted files.

If You Are Reinstalling

If you are reinstalling and have saved your old `/etc/fstab` file, you can use it to assist you in populating your new `/etc/fstab` file and in creating the necessary mount points without keying everything in to the `nfssetup` program. For more information, see step 8 in the subsection "Steps" in this section.

Steps

These steps explain how to set up the network file system. Read through these steps before running `nfssetup` to ensure that you have all of the information that you need and then refer to this section as you execute the program.

1. Log in as `root` or become superuser.
2. To invoke, `nfssetup`, enter the following command:

   ```
   # nfssetup
   ```

3. Indicate whether you want NFS locking enabled.
4. Indicate whether you are exporting any directories.

If you answer yes, the nfssetup command prompts you for the number of nfsd daemons to run.

If you answer no, the nfssetup command skips to the next question.

5. If you are importing files, indicate the number of block I/O daemons to run.
6. Indicate whether you want to run the rwalld daemon.

   If you specified in step 4 that you are exporting directories, the nfssetup command next asks whether you want to add any directories to the /etc/exports file.

   If you specified in step 4 that you are not exporting directories, the nfssetup command next asks if you want to add any remote file systems to be mounted.

   If you are both exporting file systems and importing file systems, the nfssetup command asks you about modifying the /etc/exports file, and then about remote mounting file systems.
7. Indicate whether you want to add any directory pathnames to the /etc/exports file.

   If you choose to add any directory pathnames to the /etc/exports file the nfssetup command prompts you for the pathname of the directory to export, and what hosts or network groups to allow to import the file system.

   The following example shows how to export the directory /usr/users/jal to host1.cities.dec.com:

```
Enter the directory pathname: /usr/users/jal
     Netgroup/Machine name: host1.cities.dec.com
     Netgroup/Machine name: Return

Enter the directory pathname: Return
Directory export list complete...
```

8. Indicate whether you want to mount (import) any remote file systems.

   If you choose to import any remote file systems, the nfssetup command prompts you for the following information: the remote host name (server), the remote directory pathname, the local mount point, and whether the file system should be imported read-only.

   The following example shows how to mount the directory /usr/projects from host3 onto the local mountpoint /usr/staff/projects, and to assign read-write permissions to it:

```
Enter the remote host name: host3

     Enter the remote directory pathname: /usr/projects
     Enter the local mount point: /usr/staff/projects
     Is this a read-only mount [y] ? n

     Enter the remote directory pathname: Return

Enter the remote host name: Return
Remote directory mount list complete...
```

   If you specify a local mount point that does not exist, the nfssetup command creates it.

If You Are Reinstalling

If you are reinstalling and have saved your old `/etc/fstab` file, you can use it to assist you in populating your new `/etc/fstab` file and in creating the necessary mount points without keying everything in to the `nfssetup` program.

Simply follow these steps.

Note

These directions assume that you are an experienced user user who has saved an old version of the `/etc/fstab` file and that you will be importing the same directories again. If you are in any way uncertain about the procedure described here, use the `nfssetup` command instead.

a. Provide `nfssetup` with the name of one imported directory and its remote host so that `nfssetup` will place the necessary NFS entries in the `/etc/rc.local` file.

b. Complete steps 9 and 10 and then continue with step c.

c. When the `nfssetup` command completes, make a backup copy of your `/etc/fstab` file by entering the following command:

```
# cp /etc/fstab /etc/fstab.orig
```

d. To ensure that you do not overwrite your new `/etc/fstab` file when you restore the old one, make sure that the `fstab` file you are restoring is named something like `/etc/fstab.old`. These directions assume that the `fstab` file that is being restored is named `/etc/fstab.old`.

e. Restore your old `fstab` file.

f. Change your present working directory to the directory where the old `fstab` file has been restored by entering a command like the following (These directions assume that the old `/etc/fstab` file has been restored to the `/etc` directory):

```
# cd /etc
```

g. Invoke the Bourne shell by entering the following command:

```
# sh
```

h. To create the necessary mount points for the directories you will be importing, enter the following command-line shell script:

```
# for file in `cat /etc/fstab.old | grep -v dev | cut -d: -f2`
> do
> mkdir -p $file
> echo $file done
> done
```

The Bourne shell command-line script will return output like the following:

```
/usr/users/henry done
/usr/doctools done
/usr/projects/deadline done
```

If any other output is returned, reenter the command.

i. To include the entries of NFS mounted directories from the restored `fstab` file in the new `/etc/fstab` file without including old or duplicate `dev` entries, enter the following command. Note that you must use two right angle brackets (>>) to append these entries to the `/etc/fstab` file. One right angle bracket (>) will overwrite the file.

```
# grep -v dev /etc/fstab.old >> /etc/fstab
```

j. Check to ensure that you have not made any errors by comparing the `/etc/fstab` file with the backup file, `/etc/fstab.orig`, by entering the following command:

```
# diff /etc/fstab /etc/fstab.orig
```

If you are reinstalling an old system with no new disks and no changes to the files ystem partitioning, the `diff` command will not return anything.

If you are reinstalling an old system and adding disks or changing file system partitions, the `diff` command returns output like the following, showing the new disks or changed file system partitions:

```
2,4d1
< /dev/rz16a:/usr/local:rw:1:4:ufs::
< /dev/rz16d:/var/spool:rw:1:3:ufs:nosuid:
< /dev/rz16b:/tmp:rw:1:5:ufs:nosuid:
```

If the `diff` command returns any other output, copy the backup `/etc/fstab.orig` file to `/etc/fstab` and begin again with step f.

k. After you are assured that the `/etc/fstab` file is correct, mount the NFS directories by entering the following command:

```
# mount -a -t nfs
```

l. Exit `sh` by typing Ctrl/D.

9. You are prompted to confirm **c** the information you have entered, quit **q** `nfssetup` with no changes, or restart **r** the procedure.

If you choose **c**, the `nfssetup` command displays information similar to the following:

```
Updating files:
        /etc/rc.local
        /etc/fstab
        /etc/exports
```

10. Press Return when the `nfssetup` command asks if you want to start the NFS daemons automatically:

```
Would you like nfssetup to start the daemons automatically [y]? Return
```

If you do not have `nfssetup` start the daemons automatically, you must start them manually. For information on how staring the NFS daemons manually, see the *Guide to the Network File System.*

The `nfssetup` command then displays the following instructional text:

```
In order to mount the remote directories you wish to access,
type the following command after exiting from nfssetup:

        # /etc/mount -a -t nfs
```

Running the `/etc/mount` command prevents you from having to reboot your machine to access imported directories.

See Also

`fcntl`(2), `lockf`(3), `nfssetup`(8nfs)

Guide to the Network File System
Guide to the Yellow Pages Service

Distributed System Services Setup 3

This chapter discusses the following distributed services setup tasks:

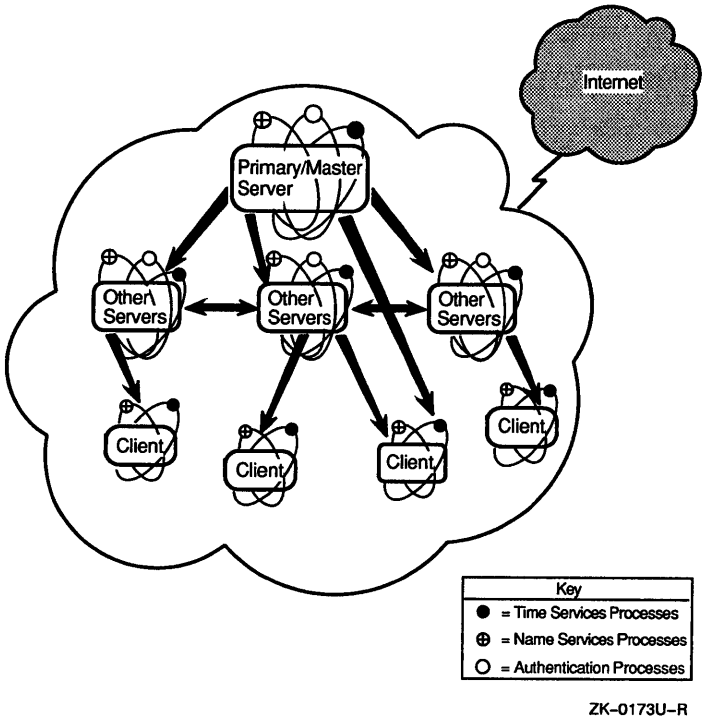
- Selecting a name service
- Setting up the BIND/Hesiod service with `bindsetup`
- Setting up the Yellow Pages service with `ypsetup`
- Setting up the `svc.conf` file with `svcsetup`
- Setting up the network time services

Overview

Your distributed environment is a set of processes that works together to coordinate time synchronization, database lookup services, and network security on your LAN. Each of the services works together and is based on a client/server model.

The following figure depicts a distributed environment that is running processes for naming services, time services, and authentication services. It illustrates, in a general way, the relationship between the master or primary servers, other servers, and clients in a distributed environment. If possible, designate the same machine as the master or primary server for all of the services. The arrows indicate the flow of data between machines.

Figure 3-1: Setting Up Distributed System Services



ZK-0173U-R

The ULTRIX operating system supports the following distributed system services:

- BIND/Hesiod and Yellow Pages naming services (which can be used either singly or in combination)
- Network Time Protocol (NTP) and Time Synchronization Protocol (TSP) time services
- Kerberos authentication service (see the *Guide to Kerberos* for information on Kerberos)

Additionally, there are three configurable security modes: BSD (the default), UPGRADE, and ENHANCED. For more information on configurable security modes, see the *Security Guide for Administrators*.

Note

Before setting up any distributed system services, your network must be up and running.

Distributed Services Setup Tasks

Table 3-1 lists, in the order in which they are generally performed, the Distributed Services setup tasks that are required or optional for both workstations and servers. The symbol Yes emphasizes an optional setup task that you would probably perform when setting up your system.

Table 3-1: Distributed Services Setup Tasks for Workstations and Servers

Setup Task	Workstation		Server	
	Required	Optional	Required	Optional
Setting up YP	No	Yes	No	Yes
Setting up BIND/HESIOD	No	Yes	No	Yes
Setting up the `svc.conf` file	Yes†	–	Yes†	–
Setting up the network time services	Yes††	Yes	Yes††	Yes

† Only if you are running YP or BIND
†† Only if you are running NFS or Kerberos

Selecting a Name Service

The ULTRIX operating system supports the BIND/Hesiod and Yellow Pages (YP) naming services. Depending on your network and your security needs, you can run one or both of them. You can also choose not to run a naming service at all.

Both BIND/Hesiod and YP are based on a client/server model, and both enable you to coordinate the distribution of information on your LAN. However, they do not provide exactly the same functionality.

Before selecting which name services to run on your LAN, determine the following:

- Your security needs

  Although security issues are beyond the scope of this manual, they are important when you are deciding which name service to run.

  If you want to take advantage of the enhanced security features supported by the ULTRIX operating system, your environment must be homogeneous (all hosts running ULTRIX Version 4.0 or higher), and you must use BIND/Hesiod to distribute the `passwd` and `auth` databases. Also, you must run BIND/Hesiod servers with Kerberos to prevent server spoofing, and to authenticate the enhanced security features. For more information on the Kerberos authentication service, see the *Guide to Kerberos.*

- Whether your network is heterogeneous or homogeneous

  Your network configuration places constraints on what databases you can serve to which hosts. A heterogeneous environment is one in which other vendors' machines run YP or BIND, or machines run ULTRIX prior to version 4.0. All implementations of YP, regardless of vendor, are compatible, as are all implementations of BIND. However, if your environment has hosts running ULTRIX Version 4.0 or higher, those hosts can only use Hesiod to serve databases within your LAN to machines that are also running Hesiod.

- Which databases you want to distribute, and with which name service

  For the most part, BIND/Hesiod and YP distribute the same databases. You can distribute some databases with BIND/Hesiod and some with YP if you have both services running on your LAN.

  However, if your LAN is connected to the Internet and you want to be able to resolve host names and addresses across the Internet, you must use BIND/Hesiod to distribute the `hosts` database. If your LAN is not connected to the Internet, both BIND/Hesiod and YP provide good host name and address resolution within your LAN.

  Also, you must use BIND/Hesiod to distribute the `auth` database if you are using the ULTRIX operating system's enhanced security features.

  Only YP distributes the `netgroup` database, which defines network-wide groups used for permission checking when doing remote mounts, remote logins, and remote shells.

  Table 3-2 summarizes the features of each naming service, and the circumstances under which you would run each one.

Table 3-2: YP, BIND, BIND/Hesiod Functionality

Functionality	Name Service		
	YP	**BIND**	**BIND/Hesiod**
Enhanced security			Yes
Serving databases in a Digital-only environment	Yes	Yes	Yes
Serving databases in a multi-vendor environment	Yes	Yes	
Wide area network connectivity		Yes	Yes

Table 3-2: (continued)

Functionality	Name Service		
	YP	BIND	BIND/Hesiod
Serving `netgroup` database	Yes		
Serving specific fields of the `passwd` database	Yes		

The following sections provide a brief description of each name service. For more detailed information on name services, see the *Guide to the BIND/Hesiod Service* and the *Guide to the Yellow Pages Service.*

BIND/Hesiod Functionality

BIND/Hesiod distributes the following databases:

```
aliases                 passwd
auth                    protocols
group                   rpc
hosts                   services
networks
```

The Berkeley Internet Name Domain (BIND) service organizes the entire Internet hierarchically, and provides domain name-to-Internet address mapping (or resolution) for hosts throughout the Internet. At the top of the BIND hierarchy are seven root name servers that recognize the top-level domains (for example, `com`, `gov`, `mil`, and `org`). The top-level domains are further divided into subdomains.

When you register your network with the Network Information Center (NIC), it assigns your network a unique number and domain name. If you decide to use BIND to distribute the `hosts` database and are connected to the Internet, you must use the BIND domain name assigned to your network by the NIC.

Your LAN must be connected to the Internet to take advantage of the Internet-wide host name-to-address resolution functionality of BIND, although you can also use BIND to resolve host names and addresses within your LAN.

Hesiod is layered on top of BIND, and enables you (within your LAN) to distribute all the other databases besides `hosts`. You can also write your own Hesiod application and serve your own Hesiod database. For information on writing and distributing a Hesiod application, see the *Guide to the BIND/Hesiod Service.*

If you are concerned about server spoofing (where a machine that is not a server masquerades as one that is, and distributes false information), you can run the Kerberos-authenticated `named` daemon on your BIND/Hesiod servers. Kerberos, an authentication service, guarantees the authenticity of the data that the BIND/Hesiod servers return.

Yellow Pages Functionality

The Yellow Pages Service distributes the following databases:

aliases	passwd
group	protocols
hosts	rpc
netgroup	services
networks	

YP also allows you to serve specific fields of the `passwd` database LAN-wide, while designating other fields locally. For example, the following entry in the `passwd` database indicates that all fields but the login shell field are to be derived from the master `passwd` database. The login shell for this user on the local machine is `/usr/new/csh`.

```
+gene::::::/usr/new/csh
```

YP also organizes the hosts on your LAN into a domain. However, the meaning of a YP domain differs from that of BIND/Hesiod. While a BIND/Hesiod domain allows you to resolve queries both locally and on the Internet, with YP your domain is a local, site-specific, administrative entity whose name is chosen by the local network administrator. YP resolves queries only within your LAN.

Note

If you choose not to run a naming service, you must maintain all of the databases individually on each machine as local `/etc` files, which means, for example, that if a new host is added to your LAN and you want to access it, you must add its name and IP address to your `/etc/hosts` file manually.

Setting Up the BIND/Hesiod Service with bindsetup

The BIND/Hesiod Service is a naming service that allows you to distribute the following network-wide databases:

`aliases`	`passwd`
`auth`	`protocols`
`group`	`rpc`
`hosts`	`services`
`networks`	

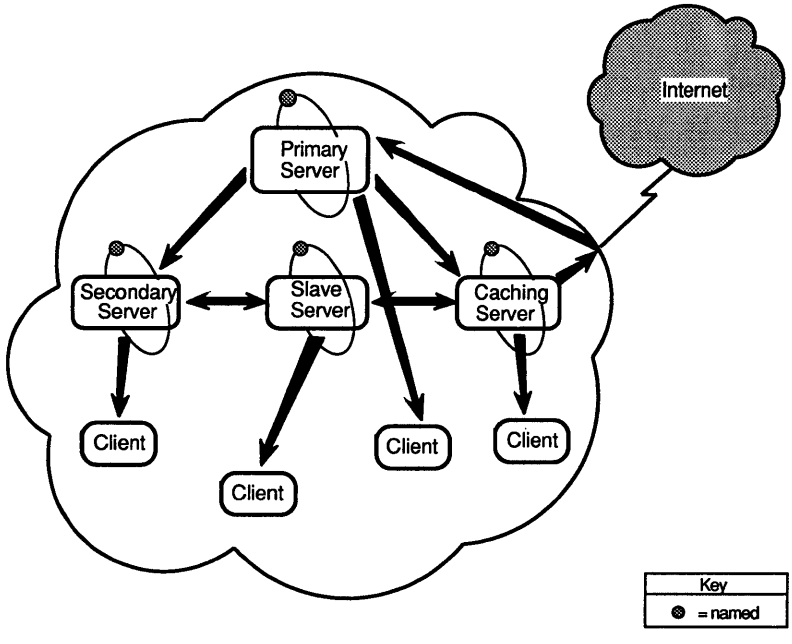
The BIND/Hesiod service is based on a client/server model. Databases are maintained on the primary server, and updated information is distributed to secondary and slave servers. Caching servers have access to the Internet, but do not maintain databases. Instead, they service queries by asking other servers for the information, and then storing the answers they receive.

For security reasons, you may want to use caching servers as routers (commonly called gateways) to the Internet rather than using primary or secondary servers as routers. Clients query a server for information.

The `bindsetup` command automates setting up the BIND/Hesiod service on your system.

The following figure depicts a distributed environment that is running BIND/Hesiod. It illustrates the processes running on each host and the relationship between the primary server, other servers, and clients. The arrows indicate the flow of data between hosts.

Figure 3-2: Setting Up BIND/Hesiod



Gathering Prerequisite Information

Table 3-3 lists the prerequisite information that you will need to gather to complete `bindsetup` and shows whether the user, system administrator, or the NIC is able to provide that information. The term user refers to anyone who has `root` privilege on a workstation and is not a system administrator.

Note that all information that the user can determine can also be requested from the system administrator. If you have any doubts about being able to determine any of the information marked as User in the following table, request that information from your system administrator.

Table 3-3: Who Can Provide Prerequisite Information

bindsetup Information	Setting Up the BIND/Hesiod Name Service	Connecting to an Existing BIND/Hesiod Name Service
The role each host will play in the distributed environment	Sys Admin	Sys Admin/User
BIND default domain name	NIC/Sys Admin	Sys Admin
For primary servers, the databases they will distribute	Sys Admin	Sys Admin
For secondary, slave, and caching servers, the host name and IP address of the primary server and one or more secondary servers	Sys Admin	Sys Admin
For clients, the host name of at least one BIND server	Sys Admin	Sys Admin
Whether you intend to run the Kerberos authentication system	Sys Admin	Sys Admin

The following list describes in detail how to gather the necessary prerequisite information listed in Table 3-3. Before setting up BIND/Hesiod, you must determine the following:

- The role each host will play in your distributed environment

  You must choose one host to be the primary server, one or more hosts to be secondary and slave servers, and (optionally) a host to be a caching server. The rest of the hosts should run as BIND/Hesiod clients. For more information on the organization of the BIND/Hesiod service and the role each of the servers plays, see the *Guide to the BIND/Hesiod Service.*

Note For Workstation Users

In most cases, workstation users with `root` privilege who are connecting their system to an existing name service, configure their systems as clients.

- Your default domain name

  The Network Information Center (NIC) assigns a default domain name when you register for an Internet network number. If you are not connected to the Internet and never plan to be, you can choose your own default domain name. If you are a workstation user with `root` privilege, request the default domain name from your system administrator.

- For the primary server, the databases you want to distribute

  If you want `bindsetup` to create the BIND/Hesiod database files for the databases you plan to distribute, you must copy the `/etc`- style source files for each database to the `/var/dss/namedb/src` directory.

  For example, to create BIND/Hesiod database files for every database that BIND/Hesiod distributes, follow these steps:

  1. Log in as `root` or become superuser.
  2. Invoke the Bourne shell by entering the following command:

     ```
     # sh
     ```

  3. Change your working directory to `/etc` by entering the following command:

     ```
     # cd /etc
     ```

  4. Copy the necessary `/etc` files to the `/var/dss/namedb/src` directory by entering the following command-line shell script:

```
# for file in aliases auth group hosts networks passwd protocols rpc services
> do
> cp $file /var/dss/namedb/src
> echo $file done
> done
```

  5. Exit `sh` by typing Ctrl/D.

- For secondary and slave servers, the host name and Internet address of the primary server, and one or more secondary servers
- For clients, the host name and Internet address of at least one BIND server. If you are a workstation user with `root` privilege, simply request this information from your system administrator.
- Whether you intend to run the Kerberos authentication service

  You can use the `bindsetup` command to configure a Kerberos-authenticated BIND/Hesiod server on your system. For more information, see the *Guide to Kerberos*.

Steps

These steps explain how to set up BIND/Hesiod. Read through these steps before running `bindsetup` to ensure that you have all of the information that you need and then refer to this section as you execute the program.

1. Log in as `root` or become superuser.
2. To invoke `bindsetup`, enter the following command:

   ```
   # bindsetup
   ```

3. Select the `add` option from the configuration menu:

   ```
   Berkeley Internet Name Domain (BIND)
           Action Menu for Configuration

           Add                 => a
           Modify              => m
           Remove              => r
           Exit                => e

   Enter your choice [a]: Return
   ```

4. Enter the your default domain name.

If you are setting up your system as a primary server go to "Primary Server" section. If you are setting up your system as a secondary or slave server go to "Secondary or Slave Server" section. If you are setting up your system as a caching server go to "Caching Server" section. If you are setting up your system as a client go to "Client" section.

Configuring a Primary Server

If you are setting up your system as a primary server, complete the following steps:

1. Select the `primary` option from the configuration menu, and answer `yes` when `bindsetup` asks if you want to convert the source files in `/var/dss/namedb/src` to the appropriate BIND/Hesiod format.

   If you answer `no`, or if the `/var/dss/namedb/src` directory is empty, `bindsetup` edits the appropriate system files, but prints a warning that you must create the database files once `bindsetup` is complete.

2. Answer `no` when `bindsetup` asks if you want to run a Kerberos-authenticated `named` daemon.

   If you decide to run Kerberos later, see the *Guide to Kerberos* for setup information.

3. Answer `yes` when `bindsetup` asks if you want to start the `named` daemon automatically.

   The `bindsetup` command starts the `named` daemon and exits.

4. After `bindsetup` is finished, you must edit the `/etc/svc.conf` file. For information on editing the `svc.conf` file see the section "Setting Up the svc.conf File with svcsetup" in this chapter.

Configuring a Secondary or Slave Server

If you are setting up your system as a secondary or slave server, complete the following steps:

1. Select the `secondary` or `slave` option from the configuration menu.

   The setup for secondary and slave servers is the same, although the servers function differently.

2. Enter the host name and Internet address of the primary server for your domain.
3. Answer `no` when `bindsetup` asks if you want to run a Kerberos-authenticated `named` daemon.

   If you decide to run Kerberos later, see the *Guide to Kerberos* for setup information.

4. Answer `yes` when `bindsetup` asks if you want to start the `named` daemon automatically.

   The `bindsetup` command starts the `named` daemon and exits.

5. After `bindsetup` is finished, you must edit the `/etc/svc.conf` file. For information on editing the `svc.conf` file see the section "Setting Up the svc.conf File with svcsetup" in this chapter.

Configuring a Caching Server

If you are setting up your system as a caching server, complete the following steps: Select the `caching` option from the configuration menu.

1. Answer `no` when `bindsetup` asks if you want to run a Kerberos-authenticated `named` daemon.

   If you decide to run Kerberos later, see the *Guide to Kerberos* for setup information.

2. Answer `yes` when `bindsetup` asks if you want to start the `named` daemon automatically.

   The `bindsetup` command starts the `named` daemon, and exits.

3. After `bindsetup` is finished, you must edit the `/etc/svc.conf` file. For information on editing the `svc.conf` file see the section "Setting Up the svc.conf File with svcsetup" in this chapter.

Configuring a Client

If you are setting up your system as a client, complete the following steps:

Note

There must be a primary server already configured for the domain before any clients run.

1. Select the `client` option from the configuration menu.
2. Enter the host name and Internet address of at least one server for your domain.

   It is recommended that you enter the primary server and one or more secondary servers.

3. After `bindsetup` is finished, you must edit the `/etc/svc.conf` file. For information on editing the `svc.conf` file see the section "Setting Up the svc.conf File with svcsetup" in this chapter.

See Also

`bindsetup`(8), `svcsetup`(8)

Introduction to Networking and Distributed System Services
Guide to the BIND/Hesiod Service

Setting Up the Yellow Pages Service with ypsetup

The Yellow Pages (YP) service provides a distributed data lookup service for sharing information between systems on a local area network (LAN). YP allows you to distribute the following network-wide databases:

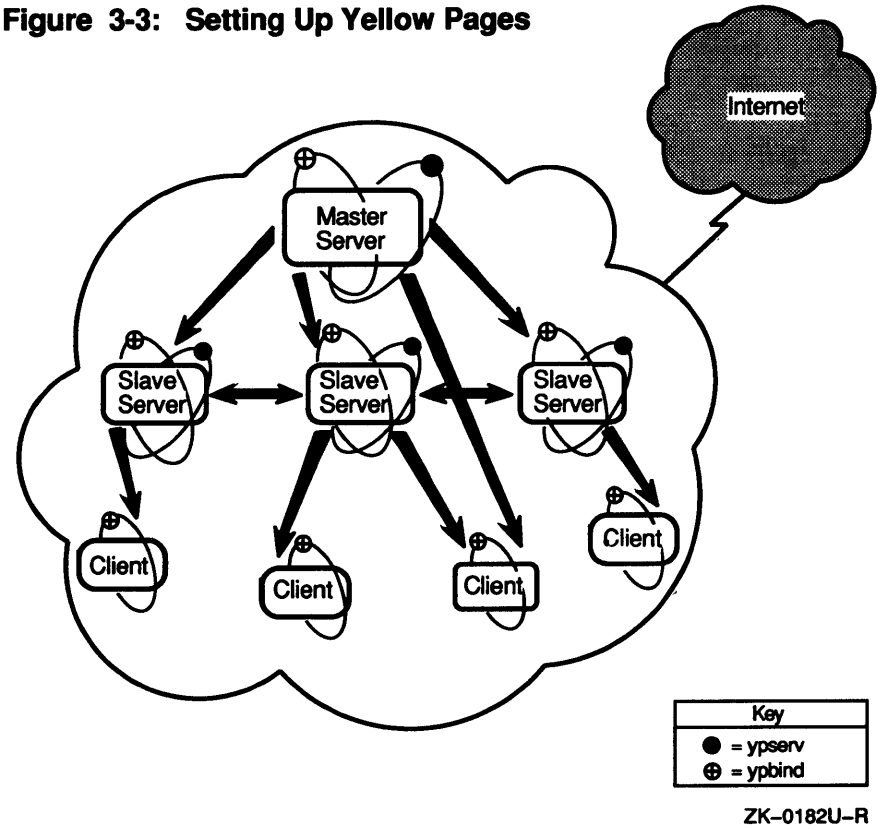
`aliases`	`passwd`
`group`	`protocols`
`hosts`	`rpc`
`netgroup`	`services`
`networks`	

YP is based on a client/server model. Database files (also known as maps) are located in `/var/yp/`*domainname*, and are stored and maintained on a master server. Changes to the database files are propagated at regular intervals to the slave servers. Clients do not store databases locally; they query servers for information.

The `ypsetup` command automates setting up YP on your system.

The following figure depicts a distributed environment that is running YP. It illustrates the processes running on each host and the relationship between the master server, slave servers, and clients. The arrows indicate the flow of data between hosts.

Figure 3-3: Setting Up Yellow Pages



Gathering Prerequisite Information

Table 3-4 lists the prerequisite information that you will need to gather to complete `ypsetup` and shows whether the user, system administrator, or the NIC is able to provide that information. The term user refers to anyone who has `root` privilege on a workstation and is not a system administrator.

Note that all information that the user can determine can also be requested from the system administrator. If you have any doubts about being able to determine any of the information marked as User in the following table, request that information from your system administrator.

Table 3-4: Who Can Provide Prerequisite Information

bindsetup Information	Setting Up the Yellow Pages Name Service	Connecting to an Existing Yellow Pages Name Service
The role each host will play in the distributed environment	Sys Admin	Sys Admin/User
YP default domain name	NIC/Sys Admin	Sys Admin
For the master server, the databases it will distribute	Sys Admin	–
Whether to lock the `ypbind` daemon to a particular domain name and server list	Sys Admin	Sys Admin
Whether to run the `yppasswd` daemon	Sys Admin	Sys Admin

The following list describes in detail how to gather the necessary prerequisite information listed in Table 3-4. Before setting up the Yellow Pages, you must determine the following:

- The role each host will play in your distributed environment

  Select one host to be the master server. There can be only one master server for each domain. Select one or more hosts to be slave servers. The rest of the hosts should run as YP clients.

  Note For Workstation Users

  In most cases, workstation users with `root` privilege who are connecting their system to an existing name service, configure their systems as clients.

  For more information on the organization of YP, and the roles each of the servers plays, see the *Guide to the Yellow Pages Service.*

- Your default domain name

  A YP domain is an administrative entity that is organized into a master server, one or more slave servers, and numerous clients. The domain name that you choose can be any string of 31 or fewer alphanumeric characters. All systems in the domain must declare the same domain name.

- For the master server, what databases it will distribute.

  For more information on which databases to distribute, see the preceding section as well as the *Guide to the Yellow Pages Service.*

- Whether you want to lock the `ypbind` daemon to a particular domain name and server list

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

  If you choose to run YP with the `-S` option configured, you must know the host names of the servers to which you are locking the `ypbind` daemon.

  If you are a workstation user with `root` privilege, request this information from your system administrator.

- Whether to run the `yppasswdd` daemon

  The `yppasswdd` daemon allows the master copy of the password file to be updated remotely.

Steps

These steps explain how to set up the Yellow Pages. Read through these steps before running `ypsetup` to ensure that you have all of the information that you need and then refer to this section as you execute the program.

1. Log in as `root` or become superuser.
2. To invoke `ypsetup`, enter the following command:

   ```
   # ypsetup
   ```

3. Enter the default domain name.

   The domain name can be any combination of letters and numbers. All systems in the domain must enter the same domain name.

4. Select whether you are configuring a master server (`m`), slave server (`s`), or a client (`c`).

Note

If you are establishing a YP domain for the first time, you must configure the master server first.

Configuring a Master Server

If you are setting up a master server, complete the following steps:

1. Enter `yes` or `no` when asked if you want to run the `yppasswdd` daemon.
2. List the names of other systems that will be configured as servers.

   The hosts that you specify must be listed in the `/etc/hosts` file. These hosts automatically receive updated versions of the `hosts` database.

   After `ypsetup` initializes the domain maps, it displays an informational message.
3. Indicate whether you want to add the `-S` option to the `ypbind` daemon to lock it to a specific domain name and server list. Digital recommends that you run YP with the `-S` option configured.

   If you choose not to configure the `-S` option, go to step 5.
4. Indicate the number of servers that will make up the set of servers to which the `ypbind` daemon locks, and their host names.

   You can specify up to four servers, although three are usually adequate. The servers that you specify are queried in the order that you specify them. Therefore, on systems that are servers, always specify the local system first. Note that each server that you specify must have an entry in the local `/etc/hosts` file.

   The following example shows how to specify that you want the `ypbind` daemon locked to three servers, `server1` (which is the host name of the local system), `server2`, and `server3`:

```
Would you like to add the -S option to ypbind [n] ? y

How many servers do you wish to specify [1] ? 3

Server 1 name: server1

Server 2 name: server2

Server 3 name: server3
```

5. Answer `yes` when `ypsetup` asks if you want to start the YP daemons automatically.

   After `ypsetup` starts the daemons, it displays an informational message reminding you to edit the `/etc/svc.conf` file with the order in which you want the name services queried for each distributed database. Then it exits. You must edit the `svc.conf` file. For information on editing the `svc.conf` file, see the section "Setting Up the svc.conf File with svcsetup" in this chapter.
6. If YP is serving either the `/etc/passwd` or `/etc/group` file (or both), you must add the character sequence `+:` to the end of the appropriate database file. These characters tell YP to go off to the main YP database for additional information.

The following example shows an /etc/passwd file that has been edited to include the +: character sequence:

```
#  /etc/passwd file that is served by YP
#
root:M11slKjPj59vA:0:1:System PRIVILEGED Account:/:/bin/csh
field:eluAN/FcWqZg.:0:1:Fld Svc PRIVILEGED Account:/usr/field:/bin/csh
nobody:Nologin:-2:-2:anonymous NFS user:/:
operator:PASSWORD HERE:0:28:Operator PRIVILEGED Account:/opr:/opr/opser
ris:Nologin:11:11:RIS Account:/usr/adm/ris:/bin/sh
daemon:*:1:1:Mr Background:/:
sys:PASSWORD HERE:2:3:Mr Kernel:/usr/sys:
+:
```

Configuring Slave Server

Before configuring your system as a slave server, be sure that there is a master server configured for your domain, that you know its name, and that it is up and running. If you are setting up a slave server, complete the following steps:

1. Specify the name of the master server for your domain.

   The ypsetup command displays a message that it is copying the YP maps from the master server.

   After ypsetup transfers the domain maps, it displays an informational message.

2. Indicate whether you want to add the -S option to the ypbind daemon to lock it to a specific domain name and server list. Digital recommends that you run YP with the -S option configured.

   If you choose not to configure the -S option, go to step 4.

3. Indicate the number of servers that will make up the set of servers to which the ypbind daemon locks, and their host names.

   You can specify up to four servers, although three are usually adequate. The servers that you specify are queried in the order that you specify them. Therefore, on systems that are servers, always specify the local system first. Note that each server that you specify must have an entry in the local /etc/hosts file.

   The following example shows how to specify that you want the ypbind daemon locked to three servers server1 (which is the name of the local system), server2, and server3:

   ```
   Would you like to add the -S option to ypbind [n] ? y

   How many-servers do you wish to specify [1] ? 3

   Server 1 name: server1

   Server 2 name: server2

   Server 3 name: server3
   ```

4. Answer yes when ypsetup asks if you want to start the YP daemons automatically.

After ypsetup starts the daemons, it displays an informational message reminding you to edit the /etc/svc.conf file with the order in which you want the name services queried for each distributed database. Then it exits. You must edit the svc.conf file. For information on editing the svc.conf file, see the section "Setting Up the svc.conf File with svcsetup" in this chapter.

5. If YP is serving either the /etc/passwd or /etc/group file (or both), you must add the character sequence +: to the end of the appropriate database file. These characters tell YP to go off to the main YP database for additional information.

   The following example shows an /etc/passwd file that has been edited to include the +: character sequence:

```
#  /etc/passwd file that is served by YP
#
root:M11slKjPj59vA:0:1:System PRIVILEGED Account:/:/bin/csh
field:eluAN/FcWqZg.:0:1:Fld Svc PRIVILEGED Account:/usr/field:/bin/csh
nobody:Nologin:-2:-2:anonymous NFS user:/:
operator:PASSWORD HERE:0:28:Operator PRIVILEGED Account:/opr:/opr/opser
ris:Nologin:11:11:RIS Account:/usr/adm/ris:/bin/sh
daemon:*:1:1:Mr Background:/:
sys:PASSWORD HERE:2:3:Mr Kernel:/usr/sys:
+:
```

Configuring a Client

Before configuring your system as a client, be sure that there is at least one master or slave server configured for your domain. To set up a client, complete the following steps:

1. Enter a c when the ypsetup command prompts you to be sure that a server is configured for your domain.
2. Indicate whether you want to add the -S option to the ypbind daemon to lock it to a specific domain name and server list. Digital recommends that you run YP with the -S option configured.

   If you choose not to configure the -S option, go to step 4.
3. Indicate the number of servers that will make up the set of servers to which the ypbind daemon locks, and their host names.

   You can specify up to four servers, although three are usually adequate. The servers that you specify are queried in the order that you specify them. Note that each server that you specify must have an entry in the local /etc/hosts file.

   The following example shows how to specify that you want the ypbind daemon locked to three servers, server1, server2, and server3:

```
Would you like to add the -S option to ypbind [n] ? y

How many servers do you wish to specify [1] ? 3

Server 1 name: server1

Server 2 name: server2

Server 3 name: server3
```

4. Answer `yes` when `ypsetup` asks if you want to start the YP daemons automatically.
5. After `ypsetup` starts the daemons, it displays an informational message reminding you to edit the `/etc/svc.conf` file with the order in which you want the name services queried for each distributed database. Then it exits. You must edit the `svc.conf` file. For information on editing the `svc.conf` file, see the section "Setting Up the svc.conf File with svcsetup" in this chapter.
6. If YP is serving either the `/etc/passwd` or `/etc/group` file (or both), you must add the character sequence `+:` to the end of the appropriate database file. These characters tell YP to go off to the main YP database for additional information.

   The following example shows an `/etc/passwd` file that has been edited to include the +: character sequence:

```
#  /etc/passwd file that is served by YP
#
root:M11slKjPj59vA:0:1:System PRIVILEGED Account:/:/bin/csh
field:eluAN/FcWqZg.:0:1:Fld Svc PRIVILEGED Account:/usr/field:/bin/csh
nobody:Nologin:-2:-2:anonymous NFS user:/:
operator:PASSWORD HERE:0:28:Operator PRIVILEGED Account:/opr:/opr/opser
ris:Nologin:11:11:RIS Account:/usr/adm/ris:/bin/sh
daemon:*:1:1:Mr Background:/:
sys:PASSWORD HERE:2:3:Mr Kernel:/usr/sys:
+:
```

See Also

`domainname`(1yp), `ypfiles`(5yp), `svcsetup`(8), `ypbind`(8yp), `yppasswd`(8yp), `ypserv`(8yp), `ypsetup`(8yp), `ypwhich`(8yp)

Guide to the Yellow Pages Service

Setting Up the svc.conf File with svcsetup

After you set up BIND/Hesiod or YP, you must run `svcsetup` or edit the `/etc/svc.conf` file manually.

The `/etc/svc.conf` file is the database service selection and security configuration file. It enables you to specify for each database the order in which the name services running on your system are to be queried.

The `svcsetup` command automates modifying the `/etc/svc.conf` file.

Gathering Prerequisite Information

Table 3-5 lists the prerequisite information that you will need to gather to complete `svcsetup` and shows whether the user, system administrator, or the NIC is able to provide that information. The term user refers to anyone who has `root` privilege on a workstation and is not a system administrator.

Note that all information that the user can determine can also be requested from the system administrator. If you have any doubts about being able to determine any of the information marked as User in the following table, request that information from your system administrator.

Table 3-5: Who Can Provide Prerequisite Information

svcsetup Information	Setting Up a New Name Service	Connecting to an Existing Name Service
Which distributed system services are running on your system	Sys Admin	User
The order in which you want the databases queried.	Sys Admin	User
Whether your environment is heterogeneous or homogeneous	Sys Admin	–

The following list describes in detail how to gather the necessary prerequisite information listed in Table 3-5. Before setting up the `svc.conf` file, you must determine the following:

- Which distributed system services are running on your system

  See the *Introduction to Networking and Distributed System Services* for information on planning a distributed environment.

- Whether your environment is Digital-only or multi-vendor

  If your network is heterogeneous and you want to distribute databases in addition to the `hosts` database, you must use Yellow Pages (YP). If you are only distributing the `hosts` database, you can use BIND/Hesiod or YP.

- For each database, the order in which the name services running on your system should be queried

The order can be different for different databases. It is recommended that `local` be the first service that your system queries for all databases, regardless of what services you are running.

Steps

These steps explain how to set up the `svc.conf` file. Read through these steps before running `svcsetup` to ensure that you have all of the information that you need and then refer to this section as you execute the program.

Note

To terminate `svcsetup` with no modifications to the `/etc/svc.conf` file, press Ctrl/C.

1. Log in as `root` or become superuser.
2. To invoke `svcsetup`, enter the following command:

   ```
   # svcsetup
   ```

   Following some explanatory text, a configuration menu is displayed; `svcsetup` prompts you to select from the following options: print the current database entries (`p`) modify them (`m`) or exit the `svcsetup` command (`e`).
3. Select the `m` option to edit the `svc.conf` file:

   ```
           Configuration Menu for the /etc/svc.conf file
           Modify File       => m
           Print File        => p
           Exit              => e
   Enter your choice [m]: m
   ```

4. Select from the menu the databases whose entries you want to edit.

   The system assigns each database a number. If you are editing multiple entries, separate the database numbers by spaces. The following example shows how to select the `aliases` and `group` database entries:

   ```
   Change Menu for the /etc/svc.conf file

           aliases           => 0
           auth              => 1
           group             => 2
           hosts             => 3
           netgroup          => 4
           networks          => 5
           passwd            => 6
           protocols         => 7
           rpc               => 8
           services          => 9

           all of the above  => 10
           none of the above => 11

   Enter your choice(s): 0 2
   ```

5. Select the order in which you want the services on your system queried, and enter the number that corresponds to it.

The following example shows how to change the setting of the `aliases` databases to `local`, and the setting of the `group` database to `local,yp`:

```
        local           => 1
        yp              => 2
        bind            => 3
        local,yp        => 4
        local,bind      => 5
        yp,local        => 6
        bind,local      => 7

Enter the naming service order for the ''aliases'' database [5]: 1

        local           => 1
        yp              => 2
        bind            => 3
        local,yp        => 4
        local,bind      => 5
        yp,local        => 6
        bind,local      => 7

Enter the naming service order for the ''group'' database [5]: 4
```

After you have indicated the changes you want to make, the `svcsetup` command exits. The changes take effect immediately.

Note

The `svcsetup` command only supports two levels of queries. If you want more than two databases queried (for example, `local`, `yp`, `bind`), you must edit the `/etc/svc.conf` file manually after `svcsetup` completes.

See Also

`svc.conf`(5), `svcsetup`(8)

Introduction to Networking and Distributed System Services
Guide to the BIND/Hesiod Service
Guide to the Yellow Pages Service

Setting Up the Network Time Services

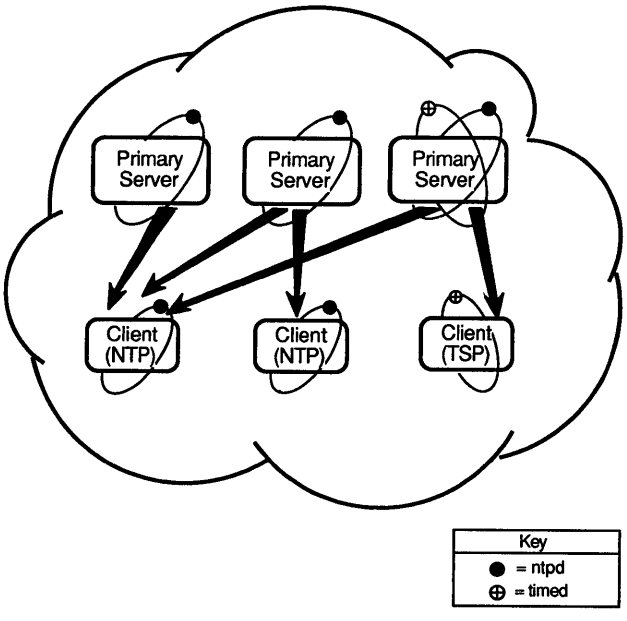
The Network Time Protocol, alone or in combination with the Time Synchronization Protocol, allows you to synchronize time in a distributed environment. As a result, any system connected to a network should run one or both of the network time services, especially if the system is running NFS or Kerberos.

The Network Time Protocol (NTP) provides accurate, dependable, and synchronized time for hosts on both wide area networks (like the Internet) and local area networks (LANs). In particular, NTP provides synchronization traceable to clocks of high absolute accuracy, and, through its various algorithms, avoids synchronization to clocks keeping bad time. NTP is implemented by the University of Maryland's `ntpd` daemon. This daemon implements Version 1 of the NTP protocol. The `/etc/ntp.conf` file is the configuration file for the daemon.

The Time Synchronization Protocol (TSP) from the University of California synchronizes network time. TSP is implemented by the `timed` daemon and has no configuration file. Digital recommends using the `timed` only on hosts that do not have the NTP daemon. If you use the `timed` daemon with the appropriate options, you can suppress the TSP averaging algorithm in favor of distributing NTP time. This allows you to use the `timed` daemon to distribute NTP time to machines that cannot run NTP. For more information, see the *Introduction to Networking and Distributed System Services.*

The following figure depicts a distributed environment that is running processes for time services. It illustrates the relationship between the primary servers and clients. Clients at sites where there is a local reference clock, or where fewer than 50 hosts are running NTP get their time, get their time from each of three primary servers. Clients that are unable to run NTP can get NTP time from a server that is running both NTP and TSP. The arrows indicate the flow of data between primary servers, a time client running NTP, and a time client running TSP.

Figure 3-3: Setting Up Time Services



Note

Your network must be up and running before you attempt to set up the network time services.

Gathering Prerequisite Information

Table 3-6 lists the prerequisite information that you will need to gather to set up the NTP or TSP time services and shows whether the user or system administrator is able to provide that information. The term user refers to anyone who has `root` privilege on a workstation and is not a system administrator.

Note that all information that the user can determine can also be requested from the system administrator. If you have any doubts about being able to determine any of the information marked as User in the following table, request that information from your system administrator.

Table 3-6: Who Can Provide Prerequisite Information

Time Services Information	Setting Up a Network Time Service	Connecting to an Existing Network Time Service
Whether you will run NTP or TSP	Sys Admin	Sys Admin/User
Whether you will be taking your NTP time from the Internet or from a local reference clock	Sys Admin	–
The number of primary NTP servers you will need	Sys Admin	–
The number of secondary NTP servers you will need	Sys Admin	–
The names and Internet addresses of the NTP servers you will place in the `/etc/ntp.conf` file	Sys Admin	Sys Admin/User

The following list describes in detail how to gather the necessary prerequisite information listed in Table 3-6. Before setting up a network time service, you must determine the following:

- Whether you will run NTP or TSP

  If your site is running ULTRIX Version 4.0 or higher, all systems on your network should run NTP and not TSP. Whereas TSP only synchronizes time among a network of clients, NTP is anchored to an actual reference clock and corrects network time to that clock. As a result, unlike TSP, NTP does not drift.

Note

Only hosts that are running ULTRIX Version 4.0 software or higher, or that have explicitly loaded the NTP software from a public source, can run NTP.

Hosts unable to run NTP must run TSP. Each LAN with hosts running TSP must have an NTP server configured on it. The NTP server should run TSP with the `-E` and `-M` options to suppress TSP's averaging algorithm in favor of distributing the NTP time of the server. The number of TSP clients does not affect the number of NTP servers.

- Whether you will be taking your time from the Internet or from a local reference clock
  - Internet Clock

    If your site is connected to the Internet your best available time source is the Internet NTP service, which consists of a set of hosts on the Internet that receive time from a highly accurate source, such as a radio receiver tuned to a time code signal broadcast by a government agency.
  - Local Reference Clock

    If your site is not connected to the Internet, select a system that is wristwatch- or radio clock-controlled to be the local reference clock. The local reference clock should be your most accurate and highly available system because the time set by the local reference clock is distributed to all hosts at your site.
- The number of primary NTP servers you will need
  - Internet Clock

    If your site is connected to the Internet, you should configure three (but no more than three) NTP primary servers at your site that synchronize to three highly accurate (stratum 1 or stratum 2) hosts on the Internet.

    The three systems that you select as primary servers should be carefully monitored and lightly loaded, if possible.

Notes

There must be at least one NTP server (primary or secondary) on each LAN that has clients running the `timed` daemon.

If you are running Kerberos, the Kerberos master should also be a primary time server. See the *Guide to Kerberos* for more information.

Use stratum 1 or stratum 2 servers for the three Internet hosts that you intend to use as peers for your primary servers.

The list of the possible Internet servers and information about their stratum level is available by means of anonymous FTP from `louie.udel.edu`. The following shows a sample FTP session in which the list of NTP servers is copied from `louie.udel.edu` to the local host:

```
% ftp louie.udel.edu
220 louie.udel.edu FTP server (Version 4.108 Sun Feb 19 22:09:45
EST 1989) ready.
Name (louie.udel.edu:user_name): anonymous
Password (louie.udel.edu:anonymous): user_name
331 Guest login ok, send ident as password.
230 Guest login ok, access restrictions apply.
ftp> cd pub/ntp
250 CWD command successful.
ftp> get clock.txt
200 PORT command successful.
150 Opening ASCII mode data connection for clock.txt (57002
bytes). 226 Transfer complete.
local: clock.txt remote: clock.txt
58409 bytes received in 14 seconds (4.2 Kbytes/s)
ftp> bye
221 Goodbye.
```

For security reasons, not all machines at a site may have anonymous FTP access.

Note

Select three machines with which to synchronize the time on your local NTP servers from the list you receive from the anonymous FTP from `louie.udel.edu`. The machines that you select are called peers. Be sure that you obtain permission from the contact person listed for the Internet server before specifying it as a peer for your local NTP servers.

- Local Reference Clock

  If your site is using a local reference clock, you can configure an unlimited number of primary servers.

  Choose three primary servers for each set of up to 50 NTP clients. The systems that you select should be carefully monitored and lightly loaded, if possible.

  Note

  There must be at least one server (primary or secondary) on each LAN that has clients running the `timed` daemon.

- The number of secondary NTP servers you will need
  - Internet Clock

    If your site has fewer than 50 hosts running NTP, you do not need to set up any secondary NTP servers. If your site has 50 or more hosts running NTP, you must configure secondary time servers.

    Choose three NTP secondary servers for each set of up to 50 hosts running NTP. The systems that you select to be secondary time servers should be carefully monitored and lightly loaded, if possible.

Note

There must be at least one server (primary or secondary) on each LAN that has clients running the `timed` daemon.

- Local Reference Clock

  If your site is using a local reference clock, you do not need to configure secondary NTP servers.

- The Internet addresses of the three NTP servers you intend to refer to in the `/etc/ntp.conf` file

  Note that for every NTP server that you refer to in the `/etc/ntp.conf` file there must be a corresponding entry for that host in your `/etc/hosts` file.

  Up to 50 NTP clients can specify the same set of three NTP time servers in their `/etc/ntp.conf` file. Therefore, the system administrator may want to distribute a common `/etc/ntp.conf` file to sets of 50 client systems.

Steps

These steps explain how to set up a network time service. Read through these steps before setting up a network time service to ensure that you have all of the information that you need and then refer to this section as you configure servers and clients.

This section explains how to perform the following tasks:

- Configuring a Local Reference Clock
- Configuring A Primary NTP Server (Local Reference Clock)
- Configuring A Primary NTP Server (Internet Clock)
- Configuring A Secondary NTP Server (Internet Clock Only)
- Configuring an NTP Client (Internet Clock and Local Reference Clock)
- Configuring a TSP Client

Essentially, with the exception of configuring a TSP client, these tasks require you to edit both the `/etc/ntp.conf` file and the `/etc/rc.local` file. Example 3-1 shows the unconfigured `/etc/ntp.conf` file that is installed on your system by default.

Example 3-1: Default /etc/ntp.conf File

```
# @(#)ntp.conf 3.1  (ULTRIX)  4/20/90
#
#               NTP  Configuration File
#        This file is mandatory for the ntpd daemon
#
#
#
#   **  A L L  **
#
#  "trusting no" prevents this host from synchronizing
#   to any host that is not listed below.  It is recommended
#   that all hosts include the line "trusting no".
#
trusting no

#
#
#    **  S E R V E R  **
#
#  If you are configuring a server, use "peer" entries to
#  synchronize to other NTP servers.  For example, server1,
#  server2, and server3.
#
#peer            server1
#peer            server2
#peer            server3

#
#
#    **  C L I E N T  **
#
#  If you are configuring a client, use "server" entries to
#  synchronize to NTP servers.  For example, server1, server2,
#  and server3.
#
#server          server1
#server          server2
#server          server3
#
#

#
#    **  L O C A L  R E F E R E N C E  C L O C K  **
#
#  If you are configuring a local reference clock, include the
#  following entry and the "trusting no" entry ONLY.
#
#peer     /dev/null     LOCL     1     -5     local
#
```

Configuring a Local Reference Clock

If you are not connected to the Internet and will synchronize your time to a local reference clock, you will need to do the following:

1. Log in as `root` or become superuser.
2. Edit the `/etc/ntp.conf` file on the system that you have chosen as the local reference clock and search for the following section:

```
#
#    **  L O C A L   R E F E R E N C E   C L O C K  **
#
#  If you are configuring a local reference clock, include the
#  following entry and the "trusting no" entry ONLY.
#
#peer     /dev/null     LOCL     1     -5     local
#
```

3. Remove the comment character (#) in front of the word `peer`, so that the line reads as follows:

```
peer     /dev/null     LOCL     1     -5     local
```

4. Edit the `/etc/rc.local` file on the system that you have chosen as the local reference clock and put in entries for the following commands. Note that the `timed` entry is optional and needs to be added only if you are supporting TSP clients.

`/usr/etc/ntpd -n`	The `/usr/etc/ntpd -n` command starts the `ntpd` daemon. The `-n` option prevents the `ntpd` program from being swapped from memory.
`/usr/etc/timed -E -M`	The `/usr/etc/timed -E -M` command starts the `timed` daemon. The `-E` option tells `timed` to distribute the time of the local machine, rather than using the TSP averaging algorithm. The `-M` option tells `timed` that this system is a time server and that it is capable of distributing time to `timed` clients. With these options set, the `timed` daemon on a server or a local reference clock distributes NTP time to TSP clients on the LAN.

To edit the `/etc/rc.local` file, follow these steps:

a. Search for the following `syslog` entry:

```
[ -f /etc/syslog ] && {
     /etc/syslog   & echo -n ' syslog'  >/dev/console
}
```

b. Place the `ntpd` and (optionally) `timed` entries immediately after the `syslog` entry. Key in the entries exactly as they appear in Example 3-2.

Example 3-2: /etc/rc.local Entries for a Local Reference Clock

```
[ -f /etc/syslog ] && {
     /etc/syslog   & echo -n ' syslog'  >/dev/console
}
[ -f /usr/etc/ntpd ] && {
     /usr/etc/ntpd -n & echo -n ' ntpd'  >/dev/console
}
[ -f /usr/etc/timed ] && {
/usr/etc/timed -E -M & echo -n ' timed'    >/dev/console
}
```

c. Ensure that you have keyed in the entries exactly as they appear in the previous example.

d. Write and quit the file.

5. Reboot your system.

   If your system is a standalone workstation, enter the w command to determine if any other users besides yourself are logged into your system, as follows:

   ```
   # w
   ```

   If you are the only user logged into your system, the w command will return data like the following, listing your login name in the first field:

```
eddie   p0 :0.0            1:04pm                      view fixXtm2d
eddie   p1 :0.0            1:04pm     3                -csh
eddie   p2 :0.0            1:04pm     2                -csh
eddie   qf                 1:03pm                      -
```

   If you are the only user logged into your system, reboot your system by entering the following command:

   ```
   # /etc/shutdown -r now
   ```

   If other users are logged into your system, the w command will return data like the following, listing the users' login names in the first field:

```
tony     p0 samsa        9:24am  3:31     10          -sh
tony     p1 samsa        9:24am  1:33      7          -sh
john     p2 badlands     12:47pm            2      1   -csh
eddie    p3 kafka        1:04pm                        w
```

   If other users are logged into your system, notify them that you are rebooting your system by entering the following command:

   ```
   # /etc/shutdown -r +15 Shutting down to start up time daemons
   ```

   If your system is a server or a time-sharing system, notify the clients or users that you are rebooting the system by entering the following command:

   ```
   # /etc/shutdown -r +15 Shutting down to start up time daemons
   ```

6. Verify that NTP is working correctly.

   To verify that NTP is working correctly, enter the /usr/etc/ntpdc command with the host name of your system as an argument, as follows:

   ```
   # /usr/etc/ntpdc `hostname`
   ```

If NTP is working, the /usr/etc/ntpdc command returns output like the following:

```
   (rem)  Address    (lcl)       Strat Poll Reach    Delay    Offset     Disp
==============================================================================
*mango           130.180.4.5        2   64  377      20.0      25.0     3.0
+super_server    130.180.4.5        2 1024  377      20.0      21.0     8.0
+batman          130.180.4.5        3 1024  376      20.0      27.0    26.0
```

For information on monitoring the ntpd daemon and using the /usr/etc/ntpdc command, see the ntpdc(8) reference page.

Configuring a Primary NTP Server (Local Reference Clock)

To configure a primary NTP server for a network using a local reference clock, follow these steps:

1. Log in as root or become superuser.
2. Edit the /etc/ntp.conf file on each of the systems you have chosen as a primary NTP server and search for the following section:

```
#
#
#    **  S E R V E R  **
#
#  If you are configuring a server, use "peer" entries to
#  synchronize to other NTP servers.  For example, server1,
#  server2, and server3.
#
#peer            server1
#peer            server2
#peer            server3
```

3. Remove the comment character (#) in front of the words peer, and replace the words server1, server2, and server3 with the names of the local reference clock and two other primary NTP servers on your network. The edited file should look similar to this:

```
#
peer            mango
peer            super_server
peer            batman
#
#
```

Note that each peer listed in the /etc/ntp.conf file must have a corresponding entry in the /etc/hosts file on the system you are configuring as a primary NTP server.

4. Edit the /etc/rc.local file on each of the systems you have chosen as a primary NTP server and put in entries for the following commands. Note that the timed entry is optional and needs to be added only if you are supporting TSP clients.

/etc/rdate -s	The /etc/rdate -s command sets this host's time to the approximate network time. The /etc/rdate -s command is included as a backup, in case all three of the peer servers are down when this system reboots.

`/usr/etc/ntp -s -f`	The `/usr/etc/ntp -s -f` command causes NTP to poll one of the peer servers specified for the time, and then synchronizes the time on this system to match that of the peer server.
`/usr/etc/ntpd -n`	The `/usr/etc/ntpd -n` command starts the `ntpd` daemon. The `-n` option prevents the `ntpd` program from being swapped from memory.
`/usr/etc/timed -E -M`	The `/usr/etc/timed -E -M` command starts the `timed` daemon. The `-E` option tells `timed` to distribute the time of the local machine, rather than using the TSP averaging algorithm. The `-M` option tells `timed` that this system is a time server and that it is capable of distributing time to `timed` clients. With these options set, the `timed` daemon on a server or a local reference clock distributes NTP time to TSP clients on the LAN.

To edit the `/etc/rc.local` file, follow these steps:

a. Search for the following `syslog` entry:

```
[ -f /etc/syslog ] && {
     /etc/syslog   & echo -n ' syslog'  >/dev/console
}
```

b. Place the `rdate`, `ntp`, `ntpd`, and (optionally) `timed` entries immediately after the `syslog` entry. Key in the entries exactly as they appear in Example 3-3, replacing the words `LocalRefClock`, `PriServer1`, and `PriServer2` with the names of your local reference clock and two peer primary NTP servers, respectively.

Example 3-3: /etc/rc.local Entries for a Primary NTP Server (Local Reference Clock)

```
[ -f /etc/syslog ] && {
     /etc/syslog   & echo -n ' syslog'  >/dev/console
}
[ -f /etc/rdate ] && {
     /etc/rdate -s   & echo -n ' rdate'          >/dev/console
}
[ -f /usr/etc/ntp ] && {
/usr/etc/ntp -s -f LocalRefClock PriServer1 PriServer2
                     & echo -n ' ntp'           >/dev/console
}
[ -f /usr/etc/ntpd ] && {
     /usr/etc/ntpd -n & echo -n ' ntpd'  >/dev/console
}
[ -f /usr/etc/timed ] && {
/usr/etc/timed -E -M & echo -n ' timed'    >/dev/console
}
```

c. Ensure that you have keyed in the entries exactly as they appear in the previous example.

d. Write and quit the file.

5. Reboot your system.

   If your system is a standalone workstation, enter the w command to determine if any other users besides yourself are logged into your system, as follows:

   ```
   # w
   ```

   If you are the only user logged into your system, the w command will return data like the following, listing your login name in the first field:

```
eddie   p0 :0.0            1:04pm                      view fixXtm2d
eddie   p1 :0.0            1:04pm     3                -csh
eddie   p2 :0.0            1:04pm     2                -csh
eddie   qf                 1:03pm                      -
```

   If you are the only user logged into your system, reboot your system by entering the following command:

   ```
   # /etc/shutdown -r now
   ```

   If other users are logged into your system, the w command will return data like the following, listing the users' login names in the first field:

```
tony      p0 samsa        9:24am  3:31     10          -sh
tony      p1 samsa        9:24am  1:33      7          -sh
john      p2 badlands     12:47pm            2      1   -csh
eddie     p3 kafka        1:04pm                        w
```

   If other users are logged into your system, notify them that you are rebooting your system by entering the following command:

   ```
   # /etc/shutdown -r +15 Shutting down to start up time daemons
   ```

   If your system is a server or a time-sharing system, notify the clients or users that you are rebooting the system by entering the following command:

   ```
   # /etc/shutdown -r +15 Shutting down to start up time daemons
   ```

6. Verify that NTP is working correctly.

   To verify that NTP is working correctly, enter the /usr/etc/ntpdc command with the host name of your system as an argument, as follows:

   ```
   # /usr/etc/ntpdc `hostname`
   ```

   If NTP is working, the /usr/etc/ntpdc command returns output like the following:

```
     (rem)  Address    (lcl)       Strat Poll Reach    Delay    Offset     Disp
==============================================================================
*mango           130.180.4.5         2   64  377       20.0      25.0     3.0
+super_server    130.180.4.5         2 1024  377       20.0      21.0     8.0
+batman          130.180.4.5         3 1024  376       20.0      27.0    26.0
```

   For information on monitoring the ntpd daemon and using the /usr/etc/ntpdc command, see the ntpdc(8) reference page.

Configuring a Primary NTP Server (Internet Clock)

To configure a primary NTP server for a network synchronizing its time from the Internet, follow these steps:

1. Log in as `root` or become superuser.
2. Edit the `/etc/ntp.conf` file on each of the systems you have chosen as a primary NTP server and search for the following section:

```
#
#
#    **  S E R V E R  **
#
#  If you are configuring a server, use "peer" entries to
#  synchronize to other NTP servers.  For example, server1,
#  server2, and server3.
#
#peer            server1
#peer            server2
#peer            server3
```

3. Remove the comment character (#) in front of the words `peer`, and replace the words `server1`, `server2`, and `server3` with the names of the three Internet hosts that you have chosen as peers. The edited file should look similar to this:

```
#
peer            mango
peer            super_server
peer            batman
#
#
```

   Note that each peer listed in the `/etc/ntp.conf` file must have a corresponding entry in the `/etc/hosts` file on the system you are configuring as a primary NTP server.

4. Edit the `/etc/rc.local` file on each of the systems you have chosen as a primary NTP server and put in entries for the following commands. Note that the `timed` entry is optional and needs to be added only if you are supporting TSP clients.

`/etc/rdate -s`	The `/etc/rdate -s` command sets this host's time to the approximate network time. The `/etc/rdate -s` command is included as a backup, in case all three of the peer servers are down when this system reboots.
`/usr/etc/ntp -s -f`	The `/usr/etc/ntp -s -f` command causes NTP to poll one of the peer servers specified for the time, and then synchronizes the time on this system to match that of the peer server.
`/usr/etc/ntpd -n`	The `/usr/etc/ntpd -n` command starts the `ntpd` daemon. The `-n` option prevents the `ntpd` program from being swapped from memory.
`/usr/etc/timed -E -M`	The `/usr/etc/timed -E -M` command starts the `timed` daemon.

The -E option tells timed to distribute the time of the local machine, rather than using the TSP averaging algorithm.

The -M option tells timed that this system is a time server and that it is capable of distributing time to timed clients. With these options set, the timed daemon on a server or a local reference clock distributes NTP time to TSP clients on the LAN.

To edit the /etc/rc.local file, follow these steps:

a. Search for the following syslog entry:

```
[ -f /etc/syslog ] && {
     /etc/syslog   & echo -n ' syslog'  >/dev/console
}
```

b. Place the rdate, ntp, ntpd, and (optionally) timed entries immediately after the syslog entry. Key in the entries exactly as they appear in Example 3-4, replacing the words *InetServer1*, *InetServer2*, and *InetServer3*, with the names of your three network peers.

Example 3-4: /etc/rc.local Entries for a Primary NTP Server (Internet Clock)

```
[ -f /etc/syslog ] && {
     /etc/syslog   & echo -n ' syslog'  >/dev/console
}
[ -f /etc/rdate ] && {
     /etc/rdate -s   & echo -n ' rdate'          >/dev/console
}
[ -f /usr/etc/ntp ] && {
usr/etc/ntp -s -f InetServer1 InetServer2 InetServer3
                      & echo -n ' ntp'          >/dev/console
}
[ -f /usr/etc/ntpd ] && {
     /usr/etc/ntpd -n & echo -n ' ntpd'  >/dev/console
}
[ -f /usr/etc/timed ] && {
/usr/etc/timed -E -M & echo -n ' timed'    >/dev/console
}
```

c. Ensure that you have keyed in the entries exactly as they appear in the previous example.

d. Write and quit the file.

5. Reboot your system.

   If your system is a standalone workstation, enter the w command to determine if any other users besides yourself are logged into your system, as follows:

```
# w
```

If you are the only user logged into your system, the w command will return data like the following, listing your login name in the first field:

```
eddie    p0 :0.0              1:04pm                        view fixXtm2d
eddie    p1 :0.0              1:04pm      3                 -csh
eddie    p2 :0.0              1:04pm      2                 -csh
eddie    qf                   1:03pm                        -
```

If you are the only user logged into your system, reboot your system by entering the following command:

```
# /etc/shutdown -r now
```

If other users are logged into your system, the w command will return data like the following, listing the users' login names in the first field:

```
tony     p0 samsa         9:24am  3:31     10          -sh
tony     p1 samsa         9:24am  1:33      7          -sh
john     p2 badlands      12:47pm            2       1  -csh
eddie    p3 kafka         1:04pm                        w
```

If other users are logged into your system, notify them that you are rebooting your system by entering the following command:

```
# /etc/shutdown -r +15 Shutting down to start up time daemons
```

If your system is a server or a time-sharing system, notify the clients or users that you are rebooting the system by entering the following command:

```
# /etc/shutdown -r +15 Shutting down to start up time daemons
```

6. Verify that NTP is working correctly.

   To verify that NTP is working correctly, enter the /usr/etc/ntpdc command with the host name of your system as an argument, as follows:

   ```
   # /usr/etc/ntpdc `hostname`
   ```

   If NTP is working, the /usr/etc/ntpdc command returns output like the following:

```
     (rem)  Address    (lcl)       Strat Poll Reach    Delay    Offset     Disp
==============================================================================
*mango           130.180.4.5         2   64  377       20.0      25.0     3.0
+super_server    130.180.4.5         2 1024  377       20.0      21.0     8.0
+batman          130.180.4.5         3 1024  376       20.0      27.0    26.0
```

   For information on monitoring the ntpd daemon and using the /usr/etc/ntpdc command, see the ntpdc(8) reference page.

Configuring a Secondary NTP Server (Internet Only)

To configure a secondary NTP server for a network synchronizing its time from the Internet, follow these steps:

1. Log in as root or become superuser.

2. Edit the /etc/ntp.conf file on each of the systems you have chosen as a secondary NTP server and search for the following section:

```
#
#
#    **  S E R V E R  **
#
#  If you are configuring a server, use "peer" entries to
#  synchronize to other NTP servers.  For example, server1,
#  server2, and server3.
#
#peer            server1
#peer            server2
#peer            server3
```

3. Remove the comment character (#) in front of the words peer, and replace the words server1, server2, and server3 with the names of three primary NTP servers at your site. The edited file should look similar to this:

```
#
peer            mango
peer            super_server
peer            batman
#
#
```

Note that each peer listed in the /etc/ntp.conf file must have a corresponding entry in the /etc/hosts file on the system you are configuring as a secondary NTP server.

4. Edit the /etc/rc.local file on each of the systems you have chosen as a secondary NTP server and put in entries for the following commands. Note that the timed entry is optional and needs to be added only if you are supporting TSP clients.

/etc/rdate -s	The /etc/rdate -s command sets this host's time to the approximate network time. The /etc/rdate -s command is included as a backup, in case all three of the peer servers are down when this system reboots.
/usr/etc/ntp -s -f	The /usr/etc/ntp -s -f command causes NTP to poll one of the peer servers specified for the time, and then synchronizes the time on this system to match that of the peer server.
/usr/etc/ntpd -n	The /usr/etc/ntpd -n command starts the ntpd daemon. The -n option prevents the ntpd program from being swapped from memory.
/usr/etc/timed -E -M	The /usr/etc/timed -E -M command starts the timed daemon.

The -E option tells timed to distribute the time of the local machine, rather than using the TSP averaging algorithm.

The -M option tells timed that this system is a time server and that it is capable of distributing time to timed clients. With these options set, the timed daemon on a server or a local reference clock distributes NTP time to TSP clients on the LAN.

To edit the /etc/rc.local file, follow these steps:

a. Search for the following syslog entry:

```
[ -f /etc/syslog ] && {
     /etc/syslog   & echo -n ' syslog'  >/dev/console
}
```

b. Place the rdate, ntp, ntpd, and (optionally) timed entries immediately after the syslog entry. Key in the entries exactly as they appear in Example 3-5, replacing the words *PriServer1*, *PriServer2*, and *PriServer3* with the names of the three primary NTP servers at your site.

Example 3-5: /etc/rc.local Entries for a Secondary NTP Server

```
[ -f /etc/syslog ] && {
     /etc/syslog   & echo -n ' syslog'  >/dev/console
}
[ -f /etc/rdate ] && {
     /etc/rdate -s   & echo -n ' rdate'          >/dev/console
}
[ -f /usr/etc/ntp ] && {
usr/etc/ntp -s -f PriServer1 PriServer2 PriServer3
                      & echo -n ' ntp'          >/dev/console
}
[ -f /usr/etc/ntpd ] && {
     /usr/etc/ntpd -n & echo -n ' ntpd'  >/dev/console
}
[ -f /usr/etc/timed ] && {
/usr/etc/timed -E -M & echo -n ' timed'    >/dev/console
}
```

c. Ensure that you have keyed in the entries exactly as they appear in the previous example.

d. Write and quit the file.

5. Reboot your system.

   If your system is a standalone workstation, enter the w command to determine if any other users besides yourself are logged into your system, as follows:

```
# w
```

If you are the only user logged into your system, the w command will return data like the following, listing your login name in the first field:

```
eddie    p0 :0.0            1:04pm                        view fixXtm2d
eddie    p1 :0.0            1:04pm      3                 -csh
eddie    p2 :0.0            1:04pm      2                 -csh
eddie    qf                 1:03pm                        -
```

If you are the only user logged into your system, reboot your system by entering the following command:

```
# /etc/shutdown -r now
```

If other users are logged into your system, the w command will return data like the following, listing the users' login names in the first field:

```
tony     p0 samsa         9:24am  3:31     10          -sh
tony     p1 samsa         9:24am  1:33      7          -sh
john     p2 badlands      12:47pm            2       1  -csh
eddie    p3 kafka         1:04pm                        w
```

If other users are logged into your system, notify them that you are rebooting your system by entering the following command:

```
# /etc/shutdown -r +15 Shutting down to start up time daemons
```

If your system is a server or a time-sharing system, notify the clients or users that you are rebooting the system by entering the following command:

```
# /etc/shutdown -r +15 Shutting down to start up time daemons
```

6. Verify that NTP is working correctly.

   To verify that NTP is working correctly, enter the /usr/etc/ntpdc command with the host name of your system as an argument, as follows:

   ```
   # /usr/etc/ntpdc `hostname`
   ```

   If NTP is working, the /usr/etc/ntpdc command returns output like the following:

```
   (rem)  Address    (lcl)       Strat Poll Reach    Delay    Offset     Disp
==============================================================================
*mango            130.180.4.5        2   64  377       20.0      25.0     3.0
+super_server     130.180.4.5        2 1024  377       20.0      21.0     8.0
+batman           130.180.4.5        3 1024  376       20.0      27.0    26.0
```

   For information on monitoring the ntpd daemon and using the /usr/etc/ntpdc command, see the ntpdc(8) reference page.

Configuring an NTP Client (Internet and Local Reference Clock)

To configure an NTP client follow these steps:

1. Log in as root or become superuser.

2. Edit the /etc/ntp.conf file on each of the systems you have chosen as a NTP client and search for the following section:

```
#
#
#    **  C L I E N T  **
#
#  If you are configuring a client, use "server" entries to
#  synchronize to NTP servers.  For example, server1, server2,
#  and server3.
#
#server          server1
#server          server2
#server          server3
#
#
```

3. Remove the comment character (#) in front of the words server, and replace the words server1, server2, and server3 with the names of three primary NTP servers if your site has fewer than 50 hosts running NTP, or three secondary NTP servers if your site has more than 50 hosts running NTP. The edited file should look similar to this:

```
#
server          mango
server          super_server
server          batman
#
```

   Note that each server listed in the /etc/ntp.conf file must have a corresponding entry in the /etc/hosts file on the system you are configuring as an NTP client.

4. Edit the /etc/rc.local file on the system you have chosen as an NTP client and put in entries for the following commands. Note that the timed entry is optional and needs to be added only if you are supporting TSP clients.

/etc/rdate -s	The /etc/rdate -s command sets this host's time to the approximate network time. The /etc/rdate -s command is included as a backup, in case all three of the peer servers are down when this system reboots.
/usr/etc/ntp -s -f	The /usr/etc/ntp -s -f command causes NTP to poll one of the peer servers specified for the time, and then synchronizes the time on this system to match that of the peer server.
/usr/etc/ntpd -n	The /usr/etc/ntpd -n command starts the ntpd daemon. The -n option prevents the ntpd program from being swapped from memory.

To edit the `/etc/rc.local` file, follow these steps:

a. Search for the following `syslog` entry:

```
[ -f /etc/syslog ] && {
     /etc/syslog   & echo -n ' syslog'  >/dev/console
}
```

b. Place the `rdate`, `ntp`, and `ntpd` entry immediately after the `syslog` entry. Key in the entries exactly as they appear in Example 3-6, replacing the words *Pri/SecServer1* , *Pri/SecServer2* , and *Pri/SecServer3* , with the names of the three primary or secondary NTP servers, depending upon the configuration at your site.

Example 3-6: /etc/rc.local Entries for an NTP Client

```
[ -f /etc/syslog ] && {
     /etc/syslog   & echo -n ' syslog'  >/dev/console
}
[ -f /etc/rdate ] && {
     /etc/rdate -s   & echo -n ' rdate'          >/dev/console
}
[ -f /usr/etc/ntp ] && {
usr/etc/ntp -s -f Pri/SecServer1 Pri/SecServer2 Pri/SecServer3
                       & echo -n ' ntp'          >/dev/console
}
[ -f /usr/etc/ntpd ] && {
     /usr/etc/ntpd -n & echo -n ' ntpd'  >/dev/console
}
```

c. Ensure that you have keyed in the entries exactly as they appear in the previous example.

d. Write and quit the file.

5. Reboot your system.

   If your system is a standalone workstation, enter the `w` command to determine if any other users besides yourself are logged into your system, as follows:

   ```
   # w
   ```

   If you are the only user logged into your system, the `w` command will return data like the following, listing your login name in the first field:

```
eddie   p0 :0.0           1:04pm                    view fixXtm2d
eddie   p1 :0.0           1:04pm     3              -csh
eddie   p2 :0.0           1:04pm     2              -csh
eddie   qf                1:03pm                    -
```

   If you are the only user logged into your system, reboot your system by entering the following command:

   ```
   # /etc/shutdown -r now
   ```

If other users are logged into your system, the w command will return data like the following, listing the users' login names in the first field:

```
tony     p0 samsa        9:24am  3:31     10         -sh
tony     p1 samsa        9:24am  1:33      7         -sh
john     p2 badlands     12:47pm            2      1  -csh
eddie    p3 kafka        1:04pm                       w
```

If other users are logged into your system, notify them that you are rebooting your system by entering the following command:

```
# /etc/shutdown -r +15 Shutting down to start up time daemons
```

If your system is a server or a time-sharing system, notify the clients or users that you are rebooting the system by entering the following command:

```
# /etc/shutdown -r +15 Shutting down to start up time daemons
```

6. Verify that NTP is working correctly.

   To verify that NTP is working correctly, enter the /usr/etc/ntpdc command with the host name of your system as an argument, as follows:

   ```
   # /usr/etc/ntpdc `hostname`
   ```

   If NTP is working, the /usr/etc/ntpdc command returns output like the following:

```
   (rem)  Address    (lcl)       Strat Poll Reach    Delay    Offset     Disp
==============================================================================
*mango           130.180.4.5        2   64  377      20.0      25.0     3.0
+super_server    130.180.4.5        2 1024  377      20.0      21.0     8.0
+batman          130.180.4.5        3 1024  376      20.0      27.0    26.0
```

   For information on monitoring the ntpd daemon and using the /usr/etc/ntpdc command, see the ntpdc(8) reference page.

Configuring a TSP Client

To configure a TSP client follow these steps:

1. Log in as root or become superuser.
2. Edit the /etc/rc.local file on the system you have chosen as a TSP client and put in an entry for the timed daemon.

To edit the /etc/rc.local file, follow these steps:

a. Search for the following syslog entry:

```
[ -f /etc/syslog ] && {
     /etc/syslog   & echo -n ' syslog'  >/dev/console
}
```

b. Place the timed entry immediately after the syslog entry. Key in the entry exactly as it appears in Example 3-7.

Example 3-7: /etc/rc.local Entries for a TSP Client

```
[ -f /etc/syslog ] && {
     /etc/syslog   & echo -n ' syslog'  >/dev/console
}
[ -f /usr/etc/timed ] && {
/usr/etc/timed  & echo -n ' timed'    >/dev/console
}
```

c. Ensure that you have keyed in the entry exactly as it appears in the previous example.

d. Write and quit the file.

3. Start the `timed` daemon by entering the following command:

```
# /usr/etc/timed
```

4. Verify that TSP is working correctly.

   To verify that TSP is working correctly, enter the `timedc` command with the `msite` option, as follows:

```
# timedc msite
```

   If TSP is working, the `timedc` command with the `msite` option returns output like the following:

```
master timedaemon runs on mango
```

   For information on monitoring the `timed` daemon and using the `timedc` command, see the `timedc`(8) reference page.

See Also

`ntp`(1), `ntp.conf`(5), `ntpd`(8), `ntpdc`(8), `timed`(8), `timedc`(8)

Introduction to Networking and Distributed System Services
Guide to Kerberos
RFC 1129—*Internet time synchronization: the Network Time Protocol*

System Setup 4

This chapter discusses the following system setup tasks:

- Adding users locally and in a distributed environment
- Adding devices
- Adding pseudoterminal devices
- Adding local area transport (LAT) devices
- Adding printers with `lprsetup`
- Establishing disk quotas

Overview

After your network is established, you should configure your system environment.

When setting up your system environment you must consider your system configuration, what system information you want to track, how much disk space you have, and what devices to attach to your system.

System Setup Tasks

Table 4-1 lists, in the order in which they are generally performed, the system setup tasks that are required or optional for both workstations and servers. The symbol [Yes] emphasizes an optional setup task that you would probably perform when setting up your system.

Table 4-1: System Setup Setup Tasks for Workstations and Servers

Setup Task	Workstation		Server	
	Required	Optional	Required	Optional
Adding users	Yes	–	Yes	–
Adding devices	No	[Yes]	No	[Yes]
Adding pseudoterminal devices	No	Yes	No	[Yes]
Adding LAT devices	No	No	No	[Yes]
Establishing disk quotas	No	Yes	No	Yes
Adding printers	No	[Yes]	No	[Yes]

Adding Users

Depending upon whether you are adding users locally or on a distributed system, you can use either the `adduser` command or add users by hand, respectively. The `adduser` command automates adding user accounts to the local system. It prompts you for information about the new user and then either creates the appropriate files or adds the information you provide to existing system files. If, for example, your LAN is not running YP or BIND/Hesiod or if you are a workstation user with `root` privilege who would like to set up an account for a user on your workstation, you would use the `adduser` command.

Note

The `adduser` command only adds users to the local system and should not be used in a distributed environment.

For each new user account, the `adduser` command creates a home directory with generic startup files copied from `/usr/skel` and a `bin` subdirectory.

You can also use `adduser` to add users to group entries in the `/etc/group` file.

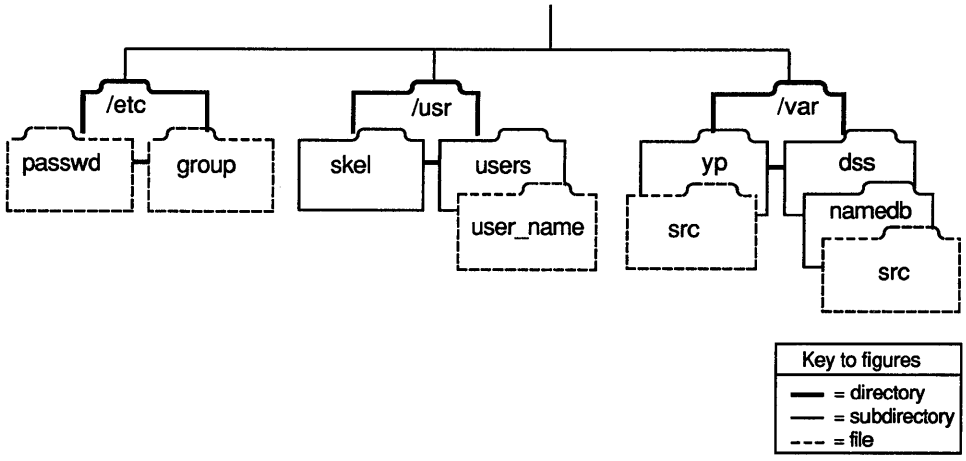
Adding user accounts in a distributed environment differs from adding user accounts to a single machine. To add user accounts if your LAN is distributing the `passwd` database with either Yellow Pages (YP) or BIND/Hesiod, you must edit and propagate the `passwd` database from the master server manually.

Note

If the master server for the `passwd` database is running at the UPGRADE or ENHANCED security level, you must use BIND/Hesiod to distribute the `passwd` and `auth` databases. Adding user accounts in an UPGRADE or ENHANCED environment is described in the *Security Guide for Administrators*.

The following figure depicts the system files that are created or changed when you add users to your system:

Figure 4-1: Adding Users



Gathering Prerequisite Information

Table 4-2 lists the prerequisite information that you will need to gather to add users to your system and shows whether the user or system administrator is able to provide that information. The term user refers to anyone who has `root` privilege on a workstation and is not a system administrator.

Note that all information that the user can determine can also be requested from the system administrator. If you have any doubts about being able to determine any of the information marked as User in the following table, request that information from your system administrator.

Table 4-2: Who Can Provide Prerequisite Information

New User Information	Adding Users With adduser	Adding Users in a Distributed Environment
New user's login name	Sys Admin/User	Sys Admin
New user's identification number (UID)	Sys Admin/User	Sys Admin
New user's group identification number (GID)	Sys Admin/User	Sys Admin
New user's real name, office number, telephone	Sys Admin/User	Sys Admin
User's startup directory	Sys Admin/User	Sys Admin
User's startup shell	Sys Admin/User	Sys Admin
In a distributed environment, which naming service you are using to distribute the `passwd` database.	–	Sys Admin
Your system's security level	–	Sys Admin

The following list describes in detail how to gather the necessary prerequisite information listed in Table 4-2. Before adding users to your system, you must determine the following:

- The new user's login name, user identification number (UID) (if the user has an account on another system in your environment), full name, login group, parent directory, and any other groups that are to include the new user

  The system uses the UID, not the login name, to determine the identity of a user. Therefore, the UID of a particular user must be the same on all systems in a networked environment. If you are in a networked environment, and the user you are adding has an account on another system, specify the same UID on this system as on the other. See the `passwd`(5) reference page for information on interpreting the fields in the `/etc/passwd` file.

  The login group determines the group identification number (GID) for processes and files created by the user. If permissions on a file are set allowing group read, write or execute, other users with the same GID can access those files.

The parent directory for the new user is the directory that contains the user's home directory. The user's home directory is the directory that the user logs in to. Users in different login groups can have their home directories under the same parent directory.

See for more information on user accounts and the `/etc/group` file. See the `chmod`(1) reference page for more information on setting file permissions.

- In a distributed environment, which naming service your LAN is using to distribute the `passwd` database

  The `passwd` database is located in the `/var/yp/src` directory if you are using YP, and in the `/var/dss/namedb/src` directory if you are using BIND/Hesiod.

- Your system's security level

  The default security level is BSD. If you want to run at a higher security level (UPGRADE or ENHANCED), special system setup is required.

  If your system is running at the UPGRADE or ENHANCED level, the `adduser` command asks security-related questions that it does not ask if your system is running at the BSD level.

  See the *Security Guide for Administrators* for information about system setup and adding user accounts on more secure systems.

Steps

These steps explain how to add users to your system. Read through these steps before adding users to your system to ensure that you have all of the information that you need and then refer to this section as you either add users locally with `adduser` or add users manually in a distributed environment.

Adding a New User Locally

To add a new user locally using the `adduser` command, follow these steps:

1. Log in as `root` or become superuser.
2. To invoke `adduser`, enter the following command:

   ```
   # adduser
   ```

3. Enter the new user's login name, UID, full name, and login group. The login name must be less than nine characters long, and cannot contain colons.

   If the new user has an account on another system in your environment, set the UID to match that of the account on the other system. You can also specify a particular UID if you have a numbering scheme for users in your environment. Otherwise, accept the default.

   The following example shows how to add a new user named John A. Laker, with a login name of `jal`, to the staff3 login group. In this example, the default UID is accepted:

   ```
   Enter login name for new user (initials, first or last name): jal
   Enter uid for new user [268]:
   Enter full name for new user: John A. Laker
   What login group should this user go into [ users ] ? staff3
   ```

The default login group is `users`, but you can specify any group as a login group. Note that the group name cannot contain colons. If you specify a group that does not exist, the `adduser` command indicates that the group is unknown, and then asks if you want to add it to the `/etc/group` file. If you choose to add the new group to the `/etc/group` file, the `adduser` command prompts you for a number for the new group. Either accept the default group number by pressing the Return key or specify another number.

The following example shows how to add the group `staff3` to the `/etc/group` file, and assign it the group number 79:

```
Unknown group: staff3. Known groups are:

system          daemon          uucp            rsrv3
bin             tty             kmem            authread
news            rsrv9           staff           ris
users           guest           operator        ingres

Do you want to add group staff3 to the /etc/group file? [yes]: Return
Adding new group to /etc/group file...

Enter group number for new group [79]: Return
```

4. Indicate other groups that are to include the new user.

   If you specify another group that does not exist, the `adduser` command goes through the sequence described in step 3.

5. Specify the parent directory for the new user.

   The default parent directory is `/usr/users`. To accept the default, press the Return key. If you want to specify a different parent directory, enter the directory pathname when the `adduser` command prompts you for it. If the parent directory you specify does not exist, the `adduser` command asks if you want to create it. The new user's home directory is automatically created as a subdirectory of the parent directory.

   The following example shows how to specify `/usr/staff/research` as the new user `jal`'s parent directory:

```
Enter parent directory for jal [/usr/users]: /usr/staff/research

/usr/staff/research not found, do you want to create it? [yes]: Return
```

6. Select a login shell for the new user.

   The `adduser` command displays a list of the supported login shells. The default shell is `/bin/csh`. Press the Return key to accept the default, or enter another shell.

```
The shells are:

/bin/sh          /bin/csh          /usr/bin/ksh      /usr/bin/sh5

Enter the users login shell name [/bin/csh]: Return
```

   Normally, the login shell you select appears as a field in the new user's entry in the `/etc/passwd` file. However, if you select the Bourne shell, `/bin/sh`, as the login shell, the field that designates the login shell is left blank.

Note

If you select a shell other than one that is listed in the prompt, unprivileged users cannot change their shell.

7. Enter and verify a password for the new user.

   Each user account added with the `adduser` command must have a password that is at least six characters long associated with it. The `adduser` command displays the following information, and then prompts you to enter and verify a password for the new user:

```
Adding new user ...
Creating home directory...
Until the password is set for jal they will not be able to login.
Enter new password:
Verify:
```

   After you have entered and verified the password, `adduser` exits.

Adding a New User in a Distributed Environment

To add a new user in a distributed environment, follow these steps:

1. Log in as `root` or become superuser.
2. Change your working directory to the directory where the `passwd` database is located.

   If you are distributing the `passwd` database with YP, it is located in `/var/yp/src`; if you are distributing the `passwd` database with BIND/Hesiod, it is located in `/var/dss/namedb/src`.

3. Edit the `passwd` database with an entry for the new user.

   The format for each user account is the same as the format in the `/etc/passwd` file:

```
login-name:password field:UID:GID:user-info:initial-working-directory:shell-program
```

   Set the *password* field to `Nologin`.

4. Rebuild the password database.

   If the `passwd` database is being distributed by BIND/Hesiod, change to the `/var/dss/namedb` directory and run the `make passwd` command. The following sequence of commands shows you how to rebuild the `passwd` database if you are using BIND/Hesiod to distribute the `passwd` database:

```
# cd /var/dss/namedb
# make passwd
```

   If the `passwd` database is being distributed by YP, change to the `/var/yp` directory and run the `make passwd` command. The following sequence of commands shows you how to rebuild the `passwd` database if you are using YP to distribute the `passwd` database:

```
# cd /var/yp
# make passwd
```

5. Set a password for the new user.

   A new user cannot log in if no password is set.

If the database is being distributed by BIND/Hesiod, use the `passwd` command to set a password for the new user. The following example shows how to set a password for the new user if you are using BIND/Hesiod to distribute the `passwd` database.

```
# passwd new_user
Changing password for new_user
Old password:
Enter new password:
Verify:
Your distributed password is updated
```

If the database is being distributed by YP, use the `yppasswd` command to set a password for the new user. The following example shows how to set a password for the new user if you are using YP to distribute the `passwd` database. The host `host1.cities.dec.com` is the YP master server.

```
# yppasswd new_user
Changing yp password for new_user
Old yp password:
New password:
Retype new password:
yellow pages passwd changed on host1.cities.dec.com
```

Note

If you are sharing the `/etc` source files for BIND/Hesiod and YP by using symbolic links, beware of changes to the `passwd` database. The `passwd` and `yppasswd` commands run independently and do not use a lock mechanism on the file. In simultaneous updates of the `passwd` database, this could result in the loss of one of the updates.

6. Create a home directory for the new user.

   You must create a home directory for each new user and populate it with the files in `/usr/skel`. You must also ensure that the correct permissions and individual and group ownerships are set on each of the files that you copy from the `/usr/skell` directory.

See Also

`chmod`(1), `chown`(8), `chgrp`(8), `mkdir`(1), `passwd`(1), `group`(5), `adduser`(8), `removeuser`(8), `vipw`(8)

Security Guide for Administrators
Introduction to Networking and Distributed System Services
Guide to the BIND/Hesiod Service
Guide to the Yellow Pages Service

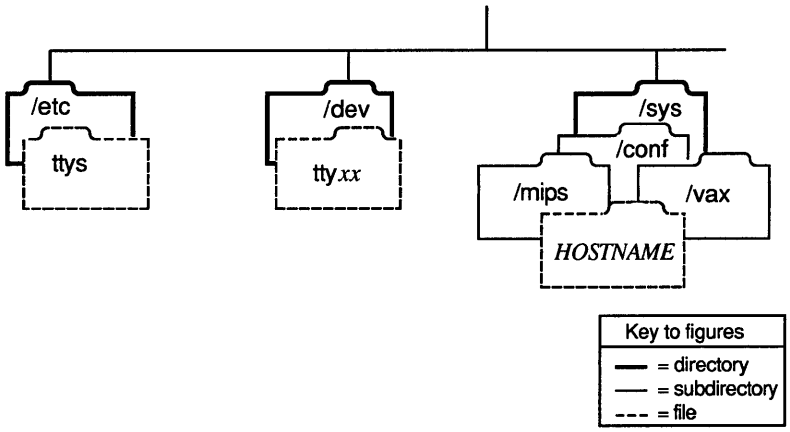
Adding Devices

If you are adding a new controller option to support additional devices (for example, disk drives, tape drives, or terminals), in most cases you have to put entries for the new controller and the new devices it supports in your system configuration file and make the controller and the new devices known to your system.

If you are adding additional devices to an existing controller option, you may not have to put any entries in your system configuration file, but you do have to create device special files and make the new devices known to your system. Terminals and modems, for example, do not require you to place entries in your system's configuration file, whereas some disks and tapes may.

The following figure depicts the system files that are created or modified when you add devices to your system:

Figure 4-2: Adding Devices



Steps

These steps explain how to add devices and controller options to your system. Read through these steps before adding new devices or controller options to ensure that you have all the information you that you need.

Adding a New Controller Option

If you are adding a new controller option to support additional disk, terminal, or tape devices, you must place entries for the option, as well as any new devices supported by the option, in your system's configuration file. You must then make device special files for the new devices and make the option and devices known to your system. After the new controller option is installed, configure it into your system by following these steps:

1. Log in as `root` or become superuser.
2. Copy `/genvmunix` to `/vmunix` by entering the following command:

```
# cp /genvmunix /vmunix
```

3. Shut down and halt your system.

   If your system is a standalone workstation, enter the w command to determine if any other users besides yourself are logged into your system, as follows:

   ```
   # w
   ```

   If you are the only user logged into your system, the w command will return data like the following, listing your login name in the first field:

```
eddie   p0 :0.0            1:04pm                      view fixXtm2d
eddie   p1 :0.0            1:04pm     3                -csh
eddie   p2 :0.0            1:04pm     2                -csh
eddie   qf                 1:03pm                      -
```

   If you are the only user logged into your system, reboot your system by entering the following command:

   ```
   # /etc/shutdown -h now
   ```

   If other users are logged into your system, the w command will return data like the following, listing the users' login names in the first field:

```
tony     p0 samsa        9:24am  3:31     10          -sh
tony     p1 samsa        9:24am  1:33      7          -sh
john     p2 badlands     12:47pm            2      1   -csh
eddie    p3 kafka        1:04pm                        w
```

   If other users are logged into your system, notify them that you are rebooting your system by entering the following command:

   ```
   # /etc/shutdown -h +15 Shutting down to add devices
   ```

   If your system is a server or a time-sharing system, notify the clients or users that you are rebooting the system by entering the following command:

   ```
   # /etc/shutdown -h +15 Shutting down to add devices
   ```

4. Reboot your system to single user mode. For information on how to boot your system to single user mode, see the *Guide to System Shutdown and Startup.*

5. When the system is up and running in single user mode, check the file system by running `fsck` with the `-p` option, as follows:

   ```
   # fsck -p
   ```

6. Set your host name by typing the following command:

   # hostname *system_name*

7. Start the error logger by entering the following command:

   ```
   # eclsd
   ```

8. Mount the `/usr` file system by entering the following command:

   ```
   # mount /usr
   ```

9. Run `doconfig` without any options to reconfigure your kernel. While your system is running with a generic kernel, the `doconfig` command places the proper configuration entries in your system's configuration file as well as make any special device files that may be required.

To run the doconfig program, enter the following command:

```
# doconfig
```

The doconfig program is menu-driven and asks you to supply the following information:

- Your system's name

  If a configuration file already exists for your system with your system's name, doconfig asks you the following question:

  ```
  A system with that name already exists. Replace it? (y/n) [n]:
  ```

  Answer y to the prompt.

- The date and time
- The time zone

The doconfig program then asks you if you want to edit the configuration file. Answer no to the prompt.

When the doconfig program finishes, it prints out information messages like the following, listing the path to the new kernel and to the log of device special files that it created in the /dev directory:

```
*** PERFORMING SYSTEM CONFIGURATION ***
    working ..... Thu Aug  1 09:11:45 EDT 1991
    working ..... Thu Aug  1 09:13:45 EDT 1991
    working ..... Thu Aug  1 09:15:46 EDT 1991
*** DEVICE SPECIAL FILE CREATION ***
    working ..... Thu Aug  1 09:16:35 EDT 1991


A log file listing Special Device Files is located in /dev/MAKEDEV.log

The new kernel is /sys/MIPS/SAMSA/vmunix
```

Record this information. You will need it in subsequent steps.

Note

If you are running BIND/Hesiod, edit the /etc/rc.local file and add the BIND extension to the hostname entry, because the doconfig program overwrites the existing hostname entry with the name of your system without its BIND extension.

10. Move /vmunix to /genvmunix by entering the following command:

    ```
    # mv /vmunix /genvmunix
    ```

11. Move the new kernel created by the doconfig program to the root partition.
12. If you are adding a new terminal controller, you will need to place entries for the new terminal lines in the /etc/ttys file.

Instead of editing the /etc/ttys file by hand, you can use the following Bourne shell command-line shell script to make the entries for you. Note that the symbol |----| indicates a tab stop. Also note that you must use two right angle brackets (>>) to append these entries to the /etc/ttys file. One right angle bracket (>) will overwrite the file.

```
# sh
# TTY_LIST="`grep tty /dev/MAKEDEV.log`"
# for tty in `echo $TTY_LIST`
> do
> echo $tty | grep -s tty
> if [ $? -eq 0 ];then
> grep -s $tty /etc/ttys
> if [ $? -gt 0 ];then
> echo "$tty|----|"/etc/getty std.9600"|----|vt100|----|on\
> |----|secure|----|#direct connect tty" >> /etc/ttys
> fi
> fi
> done
```

Exit sh by typing Ctrl/D.

The echo command will place the tty entries at the end of the /etc/ttys file. If necessary, you can move them to a more appropriate place with a text editor. For information on the meaning of each field in the /etc/ttys file and how to make edits to a tty entry, see Appendix B.

13. Reboot your system to make the new controller and devices known to your system by entering the following command:

```
# /etc/reboot
```

Adding New Devices To An Existing Controller Option

If you are connecting additional tape, disk, or terminal devices to your system that are supported by an existing controller option, you need to do the following to configure the new devices into your system:

1. Determine the device mnenumonic of the device you are adding by checking the Device Mnenumonic table in Appendix A.
2. Power down your system and connect the new peripheral device.
3. Boot your system to single-user mode. For information on how to boot your specific processor to single-user mode, see the *Guide to System Shutdown and Startup*.
4. Check the file system by running fsck with the -p option, as follows:

```
# fsck -p
```

5. Set your host name by issuing the following command, replacing the italic *system_name* with the name of your system:

```
# hostname system_name
```

6. Mount the `/usr` file system by entering the following command:

   ```
   # mount /usr
   ```

7. Start the error logger by entering the following command:

   ```
   # /etc/elcsd
   ```

8. To ensure that your system found the new device, check the error logger file by entering the following command:

   ```
   # uerf -R | more
   ```

   The error logger displays output like the following, showing the console messages that the system printed out at boot time:

```
     uerf version 4.0-009 (666)
******************************** ENTRY     1. ********************************
----- EVENT INFORMATION -----
EVENT CLASS                              OPERATIONAL EVENT
OS EVENT TYPE                    300.    SYSTEM STARTUP
SEQUENCE NUMBER                    0.
OPERATING SYSTEM                         ULTRIX 32
OCCURRED/LOGGED ON                       Thu Aug  1 09:29:25 1991 EDT
OCCURRED ON SYSTEM                       samsa.lr3.dec.com
SYSTEM ID                  x82012001     HW REV: x1
                                         FW REV: x20
                                         CPU TYPE: R2000A/R3000
PROCESSOR TYPE                           KN01
MESSAGE                                  ULTRIX V4.1 (Rev. 52) System #8: Thu
                                          _Aug  1 09:22:39 EDT 1991
                                         real mem = 20971520
                                         avail mem = 16420864
                                         using 512 buffers containing 2097152
                                          _bytes of memory
                                         KN01 processor - system rev 1
                                         cpu0 ( version 2.0, implementation 2 )
                                         fpu0 ( version 2.0, implementation 3 )
                                         dc0 at ibus0
                                         pm0 at ibus0
                                         sii0 at ibus0
                                         rz0 at sii0 slave 0 (RZ56)
                                    1    tz1 at sii0 slave 1 (TK50)
                                         ln0 at ibus0
                                         ln0: DEC LANCE Ethernet Interface,
                                          _hardware address: 08:00:2d:14:99:41
  .
  .
  .
```

9. Ensure that the system found the new device and record the unit number of the device. You need to know both the device name and the unit number to make the necessary device special files with `MAKEDEV`.

   For example, if you were adding a TK50 tape drive (mnemonic device `tz`), you would look for a `tz` entry in the list of startup events (1 in the preceding example). In the preceding example, the `tz` device was found by the system and was configured at logical unit number 1 (`tz1`).

10. Change your working directory to the `/dev` directory and make the necessary devices by entering the following command, substituting the device mnenumonic and unit number for the italic *device_mnenumonic_unit_number* in the example. Note that the output of the `MAKEDEV` command is directed to a file called

device_file by means of the tee command. If you are adding terminal devices you will use this file to include the necessary tty edits in the /etc/ttys file.

```
# cd /dev
# MAKEDEV device_mnenumonic_unit_number | tee device_file
```

The MAKEDEV command displays output like the following:

```
MAKEDEV: special file(s) for dmz3:
tty20 tty21 tty22 tty23 tty24 tty25 tty26 tty27 tty28 tty29
tty30 tty31 tty32 tty33 tty34 tty35 tty36 tty37 tty38 tty39
tty40 tty41 tty42 tty43
```

11. If you are adding terminal devices you must edit the /etc/ttys file and, using the device_file you created, add entries for each terminal device special file created by the MAKEDEV command. For example, to configure the terminal devices created by the MAKEDEV command in the preceding step, you would edit the /etc/ttys file to read similar to the following:

```
tty20   "/etc/getty std.9600"   vt100   on  secure      # direct connect tty
tty21   "/etc/getty std.9600"   vt100   on  secure      # direct connect tty
tty22   "/etc/getty std.9600"   vt100   on  secure      # direct connect tty
tty23   "/etc/getty std.9600"   vt100   on  secure      # direct connect tty
tty24   "/etc/getty std.9600"   vt100   on  secure      # direct connect tty
  .
  .
  .
```

Instead of editing the /etc/ttys file by hand, you can use the following Bourne shell command-line shell script to make the entries for you. Note that the symbol |----| indicates a tab stop. Also note that you must use two right angle brackets (>>) to append these entries to the /etc/ttys file. One right angle bracket (>) will cause the file to be overwritten.

```
# sh
# TTY_LIST="`grep tty /dev/MAKEDEV.log`"
# for tty in `echo $TTY_LIST`
> do
> echo "$tty|----|"/etc/getty std.9600"|----|vt100|----|on\
> |----|secure|----|#direct connect tty" >> /etc/ttys
> done
```

Exit sh by typing Ctrl/D.

The echo command places the tty entries at the end of the /etc/ttys file. If necessary, you can move them to a more appropriate place with a text editor. For information on the meaning of each field in the /etc/ttys file and how to make edits to a tty entry, see Appendix B.

12. Reboot your system to make the new devices known to your system by entering the following command.

```
# /etc/reboot
```

Note

If you are adding any other devices that require you to reboot your system (such as LAT and pseudoterminal devices), reboot your system only once, after you have configured in all of the devices.

See Also

pty(4), ttys(5), doconfig(8), MAKEDEV(8)

Adding Pseudoterminal Devices

Pseudoterminals enable users to access a system using the network. A pseudoterminal is a pair of character devices that emulates a hardware terminal connection to the system. Instead of hardware, however, there is a master device (`/dev/pty`*xx*) and a slave device (`/dev/tty`*xx*) .

The following processes use pseudoterminal lines: `xterm`, `dxterm`, `script`, `rlogind`, `dlogind`, `telnetd`, and `dgatewayd`.

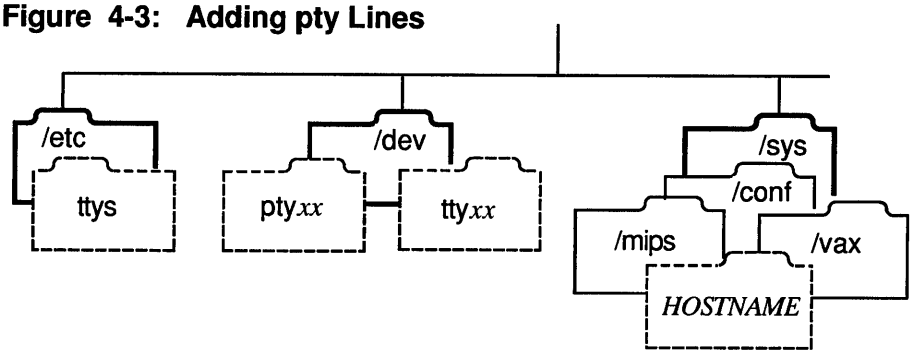
Pseudoterminal lines (`pty`) are created in sets of 16. The ULTRIX software provides a default of 32 `pty` lines, which corresponds to `pty0` and `pty1`.

For some installations, the default number of `pty` devices is adequate. However, as your user community grows, and each user wants to run multiple sessions on one or more timesharing machines in your environment, the machines may run out of available `pty` lines. The following error message is common, and results from too few `pty` devices being configured when a user tries establish a remote session using the `rlogin` command:

```
Insufficient network resources
```

The following figure depicts the system files that are created or changed when you add pty lines to your system:

Figure 4-3: Adding pty Lines



Steps

These steps explain how to add pseudoterminal devices to your system. Read through these steps before adding pseudoterminal devices to your system to ensure that you have all of the information that you need and then refer to this section as you configure `ptys`.

1. Log in as `root` or become superuser.
2. Edit the system configuration file.

   The pathname of the system configuration file is `/usr/sys/conf/vax` or `/usr/sys/conf/mips`, depending on whether yours is a VAX or RISC machine. The system configuration file name is is the host name of the machine in uppercase letters.

Search for the following line:

```
pseudo-device   pty
```

Enter the number of lines you want your system to support next to the pty entry in the pseudo-device section. The number, which must be a multiple of 16, represents the total number of pseudoterminal lines that your system supports.

For example, if you want your system to support 48 pseudoterminal lines, edit the configuration file to read as follows:

```
pseudo-device   pty     48
```

You can configure a maximum of 176 lines.

3. Change your working directory to the /dev directory and make the necessary pty devices by entering the following command. Note that the output of the MAKEDEV command is directed to a file named tty_file by means of the tee command. You will use this file to include the necessary tty edits in the /etc/ttys file.

```
# cd /dev
# MAKEDEV pty2 | tee tty_file
```

The MAKEDEV command displays output like the following:

```
MAKEDEV: special file(s) for pty2:
ptyr0 ttyr0 ptyr1 ttyr1 ptyr2 ttyr2 ptyr3 ttyr3 ptyr4 ttyr4
ptyr5 ttyr5 ptyr6 ttyr6 ptyr7 ttyr7 ptyr8 ttyr8 ptyr9 ttyr9
ptyra ttyra ptyrb ttyrb ptyrc ttyrc ptyrd ttyrd ptyre ttyre
ptyrf ttyrf
```

4. Edit the /etc/ttys file.

Add the tty*xx* half of the pseudoterminal device pair from the tty_file to the /etc/ttys file. For example, to configure the pseudoterminal devices created by the preceding MAKEDEV command, you would edit the /etc/ttys file to read similar to the following:

```
ttyr0   none            network
ttyr1   none            network
ttyr2   none            network
ttyr3   none            network
ttyr4   none            network
ttyr5   none            network
ttyr6   none            network
ttyr7   none            network
ttyr8   none            network
ttyr9   none            network
ttyra   none            network
ttyrb   none            network
ttyrc   none            network
ttyrd   none            network
ttyre   none            network
ttyrf   none            network
```

The first field indicates the name of the device special file. The second field indicates the command to be executed at startup. The third field is the type of terminal normally connected to the terminal special file.

Instead of editing the /etc/ttys file by hand, you can use the following Bourne shell command-line shell script to make the entries for you. Note that the symbol |----| indicates a tab stop. Also note that you must use two right angle brackets (>>) to append these entries to the /etc/ttys file.

One right angle bracket (>) will cause the file to be overwritten.

```
# sh
# TTY_LIST="`grep tty tty_file`"
# for tty in `echo $TTY_LIST`
> do
> echo "$tty |----|none|----||----|network" >> /etc/ttys"
> done
```

Exit `sh` by typing Ctrl/D.

The `echo` command will place the `tty` entries at the end of the `/etc/ttys` file. If necessary, you can move them to a more appropriate place with a text editor.

5. Reconfigure your kernel.

Note

If you will be adding any other devices that require you to edit your system's configuration file, such as LAT and peripheral devices, reconfigure your kernel after you have placed edits for all of these devices in your system's configuration file. Otherwise, reconfigure your kernel now.

To reconfigure your kernel, enter the following command, replacing the italic *MACHINE_NAME* with the name of your machine in capital letters:

```
# doconfig -c MACHINE_NAME
```

The `doconfig` command allows you to edit the configuration file. The following prompt appears immediately after you invoke the `doconfig` command with the `-c` option:

```
Do you want to edit the configuration file (y/n) [n]?
```

Answer `no` to this prompt.

The `doconfig` program then displays the following message as it begins to rebuild your kernel:

```
*** PERFORMING SYSTEM CONFIGURATION ***
 .
 .
 .
```

6. Make a backup copy of your kernel and then move the new kernel to `vmunix` by entering the following commands, replacing the italic *MACHINE_NAME* with the name of your machine in capital letters:

If your system is a RISC machine, enter the following commands:

```
# cp /vmunix /vmunix.orig
# mv /sys/MIPS/MACHINE_NAME/vmunix /
```

If your system is a VAX machine, enter the following commands:

```
# cp /vmunix /vmunix.orig
# mv /sys/VAX/MACHINE_NAME/vmunix /
```

See the doconfig(8) reference page for more information.

7. Shutdown and reboot your system to make the new pseudoterminal devices known to your system.

Note

If you are adding any other devices that require you to reboot your system (such as LAT and peripheral devices), reboot your system only once, after you have configured in all of the devices.

If your system is a standalone workstation, enter the w command to determine if any other users besides yourself are logged into your system, as follows:

```
# w
```

If you are the only user logged into your system, the w command will return data like the following, listing your login name in the first field:

```
eddie   p0 :0.0            1:04pm                      view fixXtm2d
eddie   p1 :0.0            1:04pm     3                -csh
eddie   p2 :0.0            1:04pm     2                -csh
eddie   qf                 1:03pm                      -
```

If you are the only user logged into your system, reboot your system by entering the following command:

```
# /etc/shutdown -r now
```

If other users are logged into your system, the w command will return data like the following, listing the users' login names in the first field:

```
tony     p0 samsa        9:24am  3:31     10          -sh
tony     p1 samsa        9:24am  1:33      7          -sh
john     p2 badlands     12:47pm            2      1   -csh
eddie    p3 kafka        1:04pm                        w
```

If other users are logged into your system, notify them that you are rebooting your system by entering the following command:

```
# /etc/shutdown -r +15 Quick Reboot
```

If your system is a server or a time-sharing system, notify the clients or users that you are rebooting the system by entering the following command:

```
# /etc/shutdown -r +15 Quick Reboot
```

See Also

pty(4), ttys(5), doconfig(8), MAKEDEV(8)

Adding Local Area Transport (LAT) Devices

The `lta` terminal driver provides support for remote terminals using the Local Area Transport (LAT) protocol. The LAT protocol allows users to access hosts on a local area network (LAN).

You can also set up printers to use the LAT to queue jobs. For information on setting up LAT printers, see the *Guide to Ethernet Communications Servers.*

LAT devices are created in sets of 16. When you install the ULTRIX operating system, if you choose to have your system configured for LAT, the appropriate entries are placed in your system's configuration file and the initial 16 LAT devices are created (`tty0- tty15`, which correspond to `lta0`) . However, you must still place the appropriate entries in the `/etc/ttys` file to activate the LAT lines (for more information, see step 6 in the subsection "Steps" in this section).

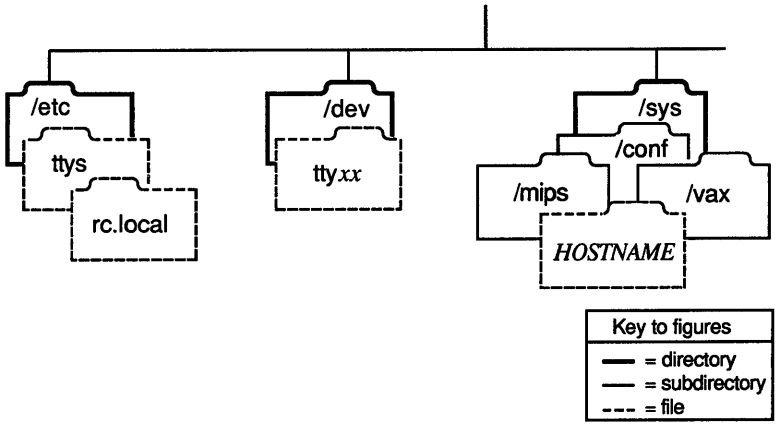
For some installations the default number of LAT devices is adequate. However, as your user community grows, and each user wants to run multiple sessions on one or more timesharing machines in your environment, the machines may run out of available LAT devices. When a user tries to connect using the LAT protocol to a timesharing machine that does not have enough LAT devices configured, the timesharing system returns the following error message:

```
Insufficient network resources
```

When users begin reporting this error, you must configure additional LAT devices.

The following figure depicts the system files that are created or modified when you add LAT lines to your system:

Figure 4-4: Adding LAT Devices



Before You Start

Before adding LAT lines to your system, you should verify that `LAT` is in the options section of your system configuration file, and that `lat` and `lta` are in the pseudo-device section of your system configuration file

The pathname of the system configuration file is `/usr/sys/conf/vax` or `/usr/sys/conf/mips`, depending on whether yours is a VAX or RISC machine.

The system configuration file name is the host name of the machine in uppercase letters. Include the following lines in the file, if they are not already there, substituting the number of LAT lines you are configuring (in multiples of 16) for the italic *n* in the example:

```
options         LAT
   .
   .
   .
pseudo-device   lat
pseudo-device   lta n
```

Steps

These steps explain how to add LAT devices to your system. Read through these steps before adding LAT devices to your system to ensure that you have all the information you that you need.

If you are adding LAT devices to your system, complete all of the following steps. If the default number of LAT devices is adequate for your site, complete steps 6 through 10 only.

1. Log in as `root` or become superuser.
2. Change your working directory to the directory where your system's configuration file is located.

   For VAX systems, enter the following command:

   ```
   # cd /usr/sys/conf/vax
   ```

   For RISC systems, enter the following command:

   ```
   # cd /usr/sys/conf/mips
   ```

3. Edit your system's configuration by following these steps. Note that the the name of your system's configuration file is the name of your system in capital letters.

   a. Search for the following line:

      ```
      pseudo-device   lta
      ```

   b. Next to the `lta` entry in the `pseudo-device` section, enter the number of lines you want your system to support. The number, which should be a multiple of 16, represents the total number of LAT lines that your system supports.

      For example, if you want your system to support 32 LAT devices, edit the configuration file to read as follows:

      ```
      pseudo-device   lta     32
      ```

      You can configure a maximum of 256 lines.

   c. When you have ensured that the entry is correct, write and quit the file.

4. Change your working directory to the `/dev` directory and make the necessary `lta` devices by entering the following command. Note that the output of the

MAKEDEV command is directed to a file named `tty_file` by means of the `tee` command. You will use this file to include the necessary `tty` edits in the `/etc/ttys` file.

```
# cd /dev
# MAKEDEV lta1 | tee tty_file
```

5. The MAKEDEV command displays output like the following:

```
MAKEDEV: special file(s) for lta1:
tty16 tty17 tty18 tty19 tty20 tty21 tty22 tty23
tty24 tty25 tty26 tty27 tty28 tty29 tty30 tty31
```

6. Edit the `/etc/ttys` file and, using the `tty_file` you created, add entries for each terminal device special file created by the MAKEDEV command. For example, to configure the terminal devices created by the preceding MAKEDEV command, you would edit the `/etc/ttys` file to read similar to the following:

```
tty16   "/etc/getty std.9600" network on nomodem  # lat terminal
tty17   "/etc/getty std.9600" network on nomodem  # lat terminal
tty18   "/etc/getty std.9600" network on nomodem  # lat terminal
tty19   "/etc/getty std.9600" network on nomodem  # lat terminal
tty20   "/etc/getty std.9600" network on nomodem  # lat terminal
tty21   "/etc/getty std.9600" network on nomodem  # lat terminal
tty22   "/etc/getty std.9600" network on nomodem  # lat terminal
tty23   "/etc/getty std.9600" network on nomodem  # lat terminal
tty24   "/etc/getty std.9600" network on nomodem  # lat terminal
tty25   "/etc/getty std.9600" network on nomodem  # lat terminal
tty26   "/etc/getty std.9600" network on nomodem  # lat terminal
tty27   "/etc/getty std.9600" network on nomodem  # lat terminal
tty28   "/etc/getty std.9600" network on nomodem  # lat terminal
tty29   "/etc/getty std.9600" network on nomodem  # lat terminal
tty30   "/etc/getty std.9600" network on nomodem  # lat terminal
tty31   "/etc/getty std.9600" network on nomodem  # lat terminal
```

Instead of editing the `/etc/ttys` file by hand, you can use the following Bourne shell command-line shell script to make the entries for you. Note that the symbol `|----|` indicates a tab stop. Also note that you must use two angle brackets (>>) to append these entries to the `/etc/ttys` file. One angle bracket (>) will overwrite the file.

```
# sh
# TTY_LIST="`grep tty tty_file`"
# for tty in `echo $TTY_LIST`
> do
> echo "$tty|----|"/etc/getty std.9600"|----|network|----|on\
> |----|nomodem|----|#|----|lat|----|terminal" >> /etc/ttys
> done
```

Exit `sh` by typing Ctrl/D.

The `echo` command will place the `tty` entries at the end of the `/etc/ttys` file. If necessary, you can move them to a more appropriate place with a text editor. For information on the meaning of each field in the `/etc/ttys` file and how to make edits to a `tty` entry, see Appendix B.

7. Edit the /etc/rc.local file by following these steps:

   a. Search for the following sendmail section:

```
[ -f /usr/lib/sendmail ] && {
        (cd /usr/spool/mqueue; rm -f lf*)
        /usr/lib/sendmail -bd -q1h -om& echo -n ' sendmail'
>/dev/console
}
```

   b. Place an entry similar to the following immediately after the sendmail section:

```
[ -f /etc/lcp ] && {
      /etc/lcp -s & echo -n ' starting LAT'      > /dev/console
}
```

   c. When you have ensured that the entry is correct, write and quit the file.

8. Reconfigure your kernel.

Note

If you will be adding any other devices that require you to edit your system's configuration file, such as pseudoterminal and peripheral devices, reconfigure your kernel after you have placed edits for all of these devices in your system's configuration file. Otherwise, reconfigure your kernel now.

To reconfigure your kernel, enter the following command, replacing the italic *MACHINE_NAME* with the name of your machine in capital letters:

```
# doconfig -c MACHINE_NAME
```

The doconfig command allows you to edit the configuration file. The following prompt appears immediately after you invoke the doconfig command with the -c option:

```
Do you want to edit the configuration file (y/n) [n]?
```

Answer no to this prompt.

The doconfig program then displays the following message as it begins to rebuild your kernel:

```
*** PERFORMING SYSTEM CONFIGURATION ***
 .
 .
 .
```

9. Make a backup copy of your kernel and then move the new kernel to vmunix by entering the following commands, replacing the italic *MACHINE_NAME* with the name of your machine in capital letters:

   If your system is a RISC machine, enter the following commands:

```
# cp /vmunix /vmunix.orig
# mv /sys/MIPS/MACHINE_NAME/vmunix /
```

If your system is a VAX machine, enter the following commands:

```
# cp /vmunix /vmunix.orig
# mv /sys/VAX/MACHINE_NAME/vmunix /
```

See the doconfig(8) reference page for more information.

10. Shutdown and reboot your system to make the new LAT devices known to your system.

Note

If you are adding any other devices that require you to reboot your system (such as pseudoterminal and peripheral devices), reboot your system only once, after you have configured in all of the devices.

If your system is a standalone workstation, enter the w command to determine if any other users besides yourself are logged into your system, as follows:

```
# w
```

If you are the only user logged into your system, the w command will return data like the following, listing your login name in the first field:

```
eddie   p0 :0.0          1:04pm                    view fixXtm2d
eddie   p1 :0.0          1:04pm     3              -csh
eddie   p2 :0.0          1:04pm     2              -csh
eddie   qf               1:03pm                    -
```

If you are the only user logged into your system, reboot your system by entering the following command:

```
# /etc/shutdown -r now
```

If other users are logged into your system, the w command will return data like the following, listing the users' login names in the first field:

```
tony     p0 samsa        9:24am  3:31     10          -sh
tony     p1 samsa        9:24am  1:33      7          -sh
john     p2 badlands     12:47pm            2      1   -csh
eddie    p3 kafka        1:04pm                        w
```

If other users are logged into your system, notify them that you are rebooting your system by entering the following command:

```
# /etc/shutdown -r +15 Quick Reboot
```

If your system is a server or a time-sharing system, notify the clients or users that you are rebooting the system by entering the following command:

```
# /etc/shutdown -r +15 Quick Reboot
```

See Also

lta(4), ttys(5), doconfig(8), lcp(8), MAKEDEV(8)

Guide to Ethernet Communications Servers

Adding Printers with lprsetup

The `lprsetup` command allows you to add local and remote printers and PrintServers to your system. The `lprsetup` command tailors the information it prompts you for depending on the type of printer you specify. With the information you provide, the `lprsetup` command then does the following:

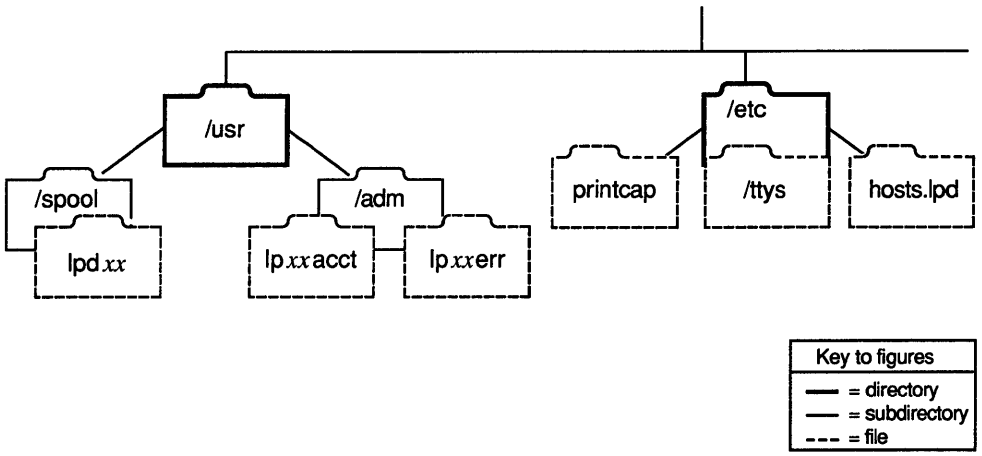
- Creates an `/etc/printcap` entry
- Creates a spooling directory
- Creates accounting files
- Creates error log files
- Modifies the `/etc/ttys` file

Note

If you did not perform an advanced installation and select the optional software subset Printer Support Environment, you must do so before attempting to use the `lprsetup` command. See the *Guide to Installing ULTRIX* and the `setld`(8) reference page for information on adding software subsets.

The following figure depicts the system files that are created or changed when you are running the `lprsetup` command:

Figure 4-5: Adding Printers



Before You Start

Before attempting to use the `lprsetup` command, you must have an understanding of the printer parameters that you are setting and the format of the `/etc/printcap` file. For an explanation of the printer parameters, see and the `printcap`(5) reference page. For sample `/etc/printcap` entries, see the `/etc/printcap.examples` file.

Gathering Prerequisite Information

Table 4-3 lists the prerequisite information that you will need to gather to complete `lprsetup` and shows whether the user or system administrator is able to provide that information. The term user refers to anyone who has `root` privilege on a workstation and is not a system administrator.

Note that all information that the user can determine can also be requested from the system administrator. If you have any doubts about being able to determine any of the information marked as User in the following table, request that information from your system administrator.

Table 4-3: Who Can Provide Prerequisite Information

lprsetup Information	Setting up a Printer
The type of printer you are adding	Sys Admin/User
The type of connection to your system and the appropriate values for the mandatory printer parameters	Sys Admin/User

The following list describes in detail how to gather the necessary prerequisite information listed in Table 4-3. Before connecting a printer to your system, you must determine the following:

- The type of printer you are adding

  The `lprsetup` command provides default settings based on the type of printer you specify. To determine the type of printer you are adding to your system, refer to the documentation that shipped with your printer. For a discussion of configurable printer parameters, see

- The type of connection to your system and the appropriate values for the mandatory printer parameters

  Printer connections fall into the following categories:

  – Device—local connection to a serial or parallel port

    If the printer is directly connected to a serial or parallel port, you must define a device for it to open for its output, **lp** in the `/etc/printcap` file. The default that `lprsetup` provides is usually adequate.

  – LAT—connection to a LAT port

    If the printer is connected to a LAT port, you must define a device for it to open for its output, **lp** in the `/etc/printcap` file, and an entry in the name field for LAT port characteristics, **op** in the `/etc/printcap` file. You must also enter the LAT terminal server node name, **ts** in the `/etc/printcap` file, and the default object service parameter, **os** in the `/etc/printcap` file. The default settings that `lprsetup` provides for **lp** and **os** are usually correct. No default settings are provided for **op** and **ts**.

    See the *Guide to Ethernet Communications Servers* for information on setting up a printer using a LAT terminal server.

- Remote—submits jobs to a remote machine

  If the printer is a remote printer you must specify the name of the machine from which it can be accessed, **rm** in the `/etc/printcap` file, and a name by which the remote machine knows the printer, **rp** in the `/etc/printcap` file. The **lp** parameter is left null by default. No default settings are provided for **rm** and **rp**.

- Network—submits jobs to a PrintServer using the network

  For PrintServers you must define the appropriate output filter **of** in the `/etc/printcap` file and the PrintServer's node name. For PrintServers running TCP/IP the output filter is `of=/usr/lib/lpdfilters/iplpscomm`.

  The **Dl** parameter is automatically set to `Dl=/usr/lib/lpdfilters/lps_v3.a`, which is correct for Version 3.0 of the PrintServer supporting host software. If the PrintServer supporting host software is Version 2.0 or 2.1, you must set the **Dl** parameter to `Dl=/usr/lib/lpdfilters/lps_40.a`.

Steps

These steps explain how to connect a printer to your system. Read through these steps before running `lprsetup` to ensure that you have all of the information that you need and then refer to this section as you execute the program.

Note

To terminate `lprsetup` with no modifications to system files press Ctrl/C.

1. Log in as `root` or become superuser.
2. To access the `lprsetup` program, enter the following command:

   ```
   # lprsetup
   ```

3. Select the `add` option from the menu:

   ```
   ULTRIX Printer Setup Program
   Command  < add modify delete exit view quit help >: add
   Adding printer entry, type '?' for help.
   ```

4. Enter the name of the printer at the prompt:

   ```
   Enter printer name to add [3] :
   For more information on the specific printer types
   Enter 'printer?'
   ```

   The `lprsetup` command uses an internal numbering scheme and, by default, assigns the next available number to the printer you are adding. It then automatically assigns the default number and `lp` *default-number* as printer names.

5. After you have entered the printer name, you are prompted to enter additional synonyms by which you want the printer known:

```
Enter printer synonym:
```

   Entering synonyms for the printer you are adding is optional. If you do not want to assign your printer a synonym, simply hit the Return key.

6. The `lprsetup` program prompts you for a series of printer parameters. Accept the defaults or specify the appropriate values for the various printer parameters when `lprsetup` prompts you.

   The prompts look like this:

```
Set device pathname 'lp' [/dev/tty03] ?

Set printer baud rate 'br' [9600] ?

Set accounting file 'af' [/usr/adm/lp3acct] ?

Set spooler directory 'sd' [/usr/spool/lpd3] ?

Set printer error log file 'lf' [/usr/adm/lp3err] ?

Set printer connection type 'ct' [dev] ?

Enter the name of the printcap symbol you wish to modify.  Other
valid entries are:
        'q'     to quit (no more changes)
        'p'     to print the symbols you have specified so far
        'l'     to list all of the possible symbols and defaults
The names of the printcap symbols are:

 af  br  cf  ct  df  dn  du  fc  ff  fo  fs  gf  ic  if  lf  lo
 lp  mc  mx  nc  nf  of  op  os  pl  pp  ps  pw  px  py  rf  rm
 rp  rs  rw  sb  sc  sd  sf  sh  st  tf  tr  ts  uv  vf  xc  xf
 xs  Da  Dl  It  Lf  Lu  Ml  Nu  Or  Ot  Ps  Sd  Si  Ss  Ul  Xf

Enter symbol name:
```

   The final prompt, `Enter symbol name:`, allows you to specify additional printer parameters (for information on the various printer parameters that can be set, see

   If you have no other printer parameters to set, enter `q`.

7. When you have entered all of the printer parameters, you are shown your choices and asked to confirm them, as follows:

```
        Printer #3
        ----------
Symbol  type  value
------  ----  -----
  af    STR    /usr/adm/lp3acct
  br    INT    9600
  ct    STR    dev
  fc    INT    0177777
  fs    INT    03
  if    STR    /usr/lib/lpdfilters/ln03rof
  lf    STR    /usr/adm/lp3err
  lp    STR    /dev/tty03
  mc    INT    20
  mx    INT    0
  of    STR    /usr/lib/lpdfilters/ln03rof
  pl    INT    66
```

```
pw      INT     80
rw      BOOL    on
sd      STR     /usr/spool/lpd3
uv      STR     4.0
xc      INT     0177777
xf      STR     /usr/lib/lpdfilters/xf
xs      INT     044000

Are these the final values for printer 3 ? [y]
```

If the parameters are correct, enter `y`. If the parameters are not correct, enter `no`. You are then prompted for the the name of the incorrect parameter so that you can change it.

8. You are asked if you want to place comments in the printer entry as follows:

```
Adding comments to printcap file for new printer, type '?' for help.
Do you want to add comments to the printcap file [n] ? :
```

9. After the `lprsetup` program completes its work, it presents you with the following message:

```
Set up activity is complete for this printer.
Verify that the printer works properly by using
the lpr(1) command to send files to the printer.
Command  < add modify delete exit view quit help >:
```

If you are finished adding printers, answer `quit` to the prompt.

10. For each printer connected to your system, you must create a device special file in the `/dev` directory. By default, a device special file may have been created for some printers during the installation process; however, some device special files must be created using the `MAKEDEV` command. A device special file name is specified as follows:

- `/dev/lp`*n* for printers attached by parallel interfaces
- `/dev/tty`*nn* for printers attached by all serial interfaces

The *n* arguments specify the port (physical connection) of each local printer connected to your system. To determine the port number of a printer, refer to the *Site Management Guide* prepared by the Digital field service person who installed the printer.

To create a device special file in the `/dev` directory for each printer, set your default directory to the `/dev` directory, then type the `ls` command to determine which device special files exist. For example, to determine if a device special file exists for `lp1`, type the following commands:

```
# cd /dev
# ls -l lp1
```

If the file exists, and the printer is attached by a parallel interface, you do not have to make a device for it.

If the file does not exist, use the `MAKEDEV` command to create the device special file.
For example, to create a device special file for `lp1`, enter the following command:

```
# MAKEDEV lp1
```

11. To enable other hosts to access your printer on the network, you must edit the `/etc/hosts.lpd` file and place in the file the name of each host that you want to access your printer.

See Also

`tty`(4), `printcap`(5), `lpd`(8), `lprsetup`(8)

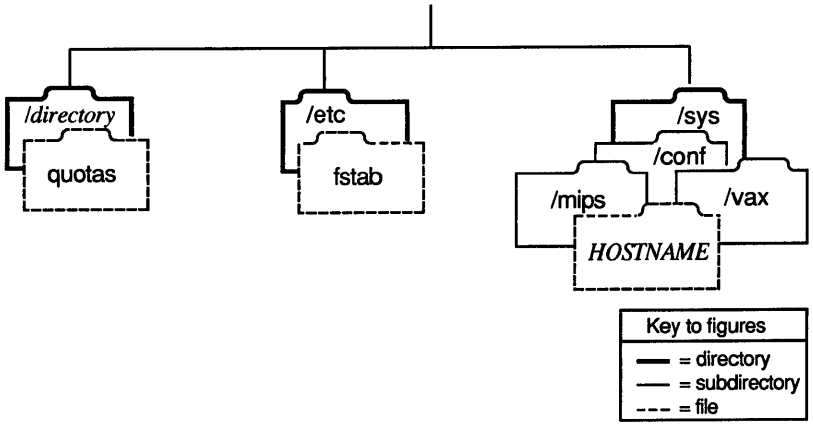
Guide to Ethernet Communications Servers

Establishing Disk Quotas

You can control the number of files and amount of disk space that each user can access by establishing disk quotas. The `edquota` command allows you to establish and modify disk quotas on individual users, and to use a prototypical user to establish the same quotas on a group of users.

The following figure depicts the system files that are created or changed when you establish disk quotas on your system:

Figure 4-6: Establishing Disk Quotas



Before You Start

Before running the `edquota` command, you should do the following:

- Verify that the QUOTA option is in your system's configuration file

  The pathname of the system configuration file is `/usr/sys/conf/vax` or `/usr/sys/conf/mips`, depending on whether yours is a VAX or RISC machine. The system configuration filename is the host name of the machine in uppercase letters.

  The QUOTA option is included in your system configuration by default. If it is not already there, include the following line in the options section of your system configuration file:

  ```
  options         QUOTA
  ```

  If you edit the system configuration file you must rebuild your kernel. Use the `doconfig` command and the `-c` option. The `-c` option specifies the use of the existing system configuration file to rebuild the kernel. The following command rebuilds the kernel of a system called `host1`:

  ```
  # doconfig -c HOST1
  ```

  See the `doconfig`(8) reference page for more information.

- Verify that the `quotaon -a` and `quotacheck -a` commands are in your `/etc/rc.local` file

Both commands are included in the default /etc/rc.local file. Look for the following entries:

```
echo -n 'check quotas: '              >/dev/console
        /etc/quotacheck -a
echo 'done.'                          >/dev/console
/etc/quotaon -a
```

- Edit the /etc/fstab file and change the mount type on the file systems for which you are establishing quotas to rq

  You can establish quotas only on file systems that are mounted locally and marked read-write.

  For example, the following /etc/fstab entry indicates that the /usr file system is read-write (rw):

  ```
  /dev/ra5h:/usr:rw:1:3:ufs::
  ```

  If you plan to establish quotas on the /usr file system, you must edit this entry to read as follows:

  ```
  /dev/ra5h:/usr:rq:1:3:ufs::
  ```

- Determine how much disk space and how many inodes you want to allot to each user

  The du command allows you to analyze disk usage. If you are establishing user partitions for the first time, you need to experiment with what amount of space is appropriate for each user. You can set quota on the number of blocks and the number of inodes. For more information, see the du(1) reference page.

Steps

These steps explain how to establish quotas on your system. Read through these steps before establishing quotas to ensure that you have all of the information that you need.

Complete step 1 if you are establishing quotas for the first time. If you are modifying quotas on a system where they are already established, start with step 2.

1. Bring your system to single-user mode.

   If your system is a standalone workstation, enter the w command to determine if any other users besides yourself are logged into your system, as follows:

   ```
   # w
   ```

   If you are the only user logged into your system, the w command will return data like the following, listing your login name in the first field:

```
eddie   p0 :0.0          1:04pm                    view fixXtm2d
eddie   p1 :0.0          1:04pm     3              -csh
eddie   p2 :0.0          1:04pm     2              -csh
eddie   qf               1:03pm                    -
```

   If you are the only user logged into your system, reboot your system by entering the following command:

   ```
   # /etc/shutdown +15 Shutting down to set up quotas
   ```

If other users are logged into your system, the `w` command will return data like the following, listing the users' login names in the first field:

```
tony     p0 samsa        9:24am  3:31     10          -sh
tony     p1 samsa        9:24am  1:33      7          -sh
john     p2 badlands     12:47pm            2      1   -csh
eddie    p3 kafka        1:04pm                        w
```

If other users are logged into your system, notify them that you are rebooting your system by entering the following command:

```
# /etc/shutdown +15 Shutting down to set up quotas
```

If your system is a server or a time-sharing system, notify the clients or users that you are rebooting the system by entering the following command:

```
# /etc/shutdown -r +15 Shutting down to set up quotas
```

2. Run the `quotacheck` command with the `-f` option.

   You must have a file named `quotas` at the top level of the file system in which you are establishing quotas. The `-f` option to the `quotacheck` command creates the `quotas` file.

   For example, if you are establishing quotas on the `/usr/staff/documentation` file system, create a `quotas` file as follows:

   ```
   # /etc/quotacheck -f /usr/staff/documentation
   ```

3. Establish disk quotas with the `edquota` command.

   To establish or modify disk quotas for each user enter the following command, substituting the login name of the user for the italic *login_name* in the example:

   ```
   # edquota login_name
   ```

   The `edquota` command places you in a temporary file that looks like this:

```
fs /usr/staff/documentation blocks (soft = 0, hard = 0) inodes (soft = 0, hard = 0)
```

   Edit this file by replacing the zeros (0) with the quotas that you want to apply to each user (a zero denotes that the user has no quota). The range of the quota is between the two values `soft` and `hard`. Once the user reaches the `soft` limit the system begins issuing warnings. After receiving three warnings or if the user reaches the hard limit, no more writes or inodes can be created unless the user deletes some files.

   When you have ensured that the edits are correct, write and quit the file.

   If you are modifying quotas that have already been established, just complete step 2 and step 3, as necessary.

4. Apply the same quotas that you established for the first user to a group of users.

   You can use the first user as a prototype for establishing quotas for a group of users as follows:

   ```
   # edquota -p user1 user2 user3 user4 user5
   ```

   The quotas that you established for `user1` are now also established for `user2`, `user3`, `user4`, and `user5`. See the `edquota`(8) reference page for more information.

5. Reboot your system to establish the quotas by entering the following command:

```
# shutdown -r now
```

See Also

du(1), doconfig(8), edquota(8), quotacheck(8), quotaon(8), quotarep(8)

"Disk Quotas in a UNIX Environment," *Supplementary Documents, Volume 3: System Manager*

Device Mnemonics A

This appendix identifies and defines the mnemonics that are used to attach any hardware or software device to your system. The mnemonics are used by the `/dev/MAKEDEV` shell script to create the character or block special files that represent each of the devices. The mnemonics also appear in the system configuration file, as described in the *Guide to Configuration File Maintenance.*

Table A-1 lists the mnemonics in nine categories: generic, systems, consoles, disks, tapes, terminals, modems, printers, and others. The generic category lists the mnemonics of a general nature and includes memory, null, trace, and tty devices. The systems category lists the mnemonic for DECstation system setup. The consoles category lists the system console devices that the ULTRIX operating system uses. The disks, tapes, terminals, modems, and printers categories identify the appropriate mnemonics for those devices. The others category lists the mnemonic for DECstation devices.

The description heading in Table A-1 identifies the corresponding device name. It does not define the mnemonic's use. For detailed information on the use of each mnemonic in relation to both the `MAKEDEV` script and the system configuration file, refer to the reference pages in Section 4 of the *ULTRIX Reference Pages.* If on-line reference pages are available, you can also use the `man` command. For instance, enter the following command at the system prompt to display the reference page for the Small Computer System Interconnect (SCSI) disk controller driver:

```
% man rz
```

Where appropriate, the SYNTAX section of the reference page defines the device's syntax as it should appear, in the `config` file. Refer to `/dev/MAKEDEV` for additional software device mnemonics that `MAKEDEV` uses. Refer to `MAKEDEV(8)` in the *ULTRIX Reference Pages* for a description of the `MAKEDEV` utility.

Table A-1 uses the convention of an asterisk (`*`) beside a mnemonic and a question mark (`?`) beside a device name to mean a variable number. The value of the variable number is dependent on the particular device.

Table A-1: Devices Supported by MAKEDEV

Category	Mnemonic	Description
Generic	boot*	Boot and std devices by cpu number; for example, boot750
	audit	Audit log device
	drum	Kernel drum device
	mvax*	All MicroVAX setups; for example, mvax2000
	vaxstation*	A VAXstation 2000 setup; for example, vaxstation2000
	std	Standard devices with all console subsystems
	errlog	Error log device
	kUmem	Kernel Unibus/Q-bus virtual memory
	kmem	Virtual main memory
	mem	Physical memory
	null	A null device
	trace	A trace device
	tty	A character terminal device
	local	Customer-specific devices
Systems	DECstation	A DECstation setup (for example, a DECstation 3100)
Consoles	console	System console interface
	crl	Console RL02 disk interface for VAX 86?0
	cs*	Console RX50 floppy interface for VAX 8??0
	ctu*	Console TU58 cassette interface for VAX 11/725/730/750
	cty*	Console extra serial line units for VAX 8??0
	cfl	Console RX01 floppy interface for 11/78?
	ttycp	Console line used as auxiliary terminal port
Disks	hp*	MASSBUS disk interface for RM?? drives and RP?? devices
	ra*	UNIBUS/Q-bus/BI/HSC/DSSI MSCP disk controller interface
	rb*	UNIBUS IDC RL02 disk controller interface for RB?? drives
	rd*	VAXstation 2000 and MicroVAX 2000 RD type drives
	rz	SCSI disks (for example, the RZ56)
	rk*	UNIBUS RK?? disk controller interface
	rl*	UNIBUS/Q-bus RL?? disk controller interface
	rx*	VAXstation 2000 and MicroVAX 2000 RX type drives
	fd	Floppy Disk (for example, the RX26)
Tapes	mu*	MASSBUS magtape interface (for example, the TU78)
	tms*	UNIBUS/Q-bus/BI/HSC/DSSI TMSCP tape controller interface
	rv*	UNIBUS/Q-bus/BI TMSCP optical disk
	ts*	UNIBUS/Q-bus TS11/TS05/TU80 magtape interface
	tu*	TE16/TU45/TU77 MASSBUS magtape interface
	st*	VAXstation 2000 and MicroVAX 2000 TZK50 cartridge tape
	tz*	SCSI tapes (for example, the TZU50)
Terminals	cxa*	Q–bus cxa16
	cxb*	Q–bus cxb16
	cxy*	Q–bus cxt08
	dfa*	Q–bus DFA01 comm multiplexer
	dhq*	Q–bus DHQ11 comm multiplexer
	dhu*	UNIBUS DHU11 comm multiplexer
	dhv*	Q–bus DHV11 comm multiplexer
	dmb*	BI DMB32 comm multiplexer including dmbsp serial printer/plotter

Table A-1: (continued)

Category	Mnemonic	Description
	dhb*	BI DHB32 comm multiplexer
	dmf*	UNIBUS DMF32 comm multiplexer including dmfsp serial printer/plotter
	dmz*	UNIBUS DMZ32 comm multiplexer
	dz	UNIBUS DZ11 and DZ32 comm multiplexer
	sh*	MicroVAX 2000, 8 serial line expansion option
	ss*	VAXstation 2000 and MicroVAX 2000 basic 4 serial line unit
	fc*	VAXstation 60 basic 4 serial line unit
	dzq*	Q–bus DZQ11 comm multiplexer
	dzv*	Q–bus DZV11 comm multiplexer
	lta*	Sets of 16 network local area terminals (LAT)
	pty*	Sets of 16 network pseudoterminals
	qd*	Q–bus VCB02 (QDSS) graphics controller/console
	qv*	Q–bus VCB01 (QVSS) graphics controller/console
	sm*	VAXstation 2000 monochrome bitmap graphics/console
	sg*	VAXstation 2000 color bitmap graphics console
	lx	VAXstation 8000 color high-performance 3D graphics
	fg*	VAXstation 60 color bitmap graphics/console
Modems	dfa*	DFA01 integral modem communications device.
Printers	dmbsp*	BI DMB32 serial printer/plotter
	dmfsp*	UNIBUS DMF32 serial printer/plotter
	lp*	UNIBUS LP11 parallel line printer
	lpv*	Q–bus LP11 parallel line printer
Packet filter	pfilt	Packet filter devices; set of 64
Other	pm*	mono/color bitmap graphics/mouse/modem /printer/terminals for DECstation 3100

System Files B

The ULTRIX system files perform a number of functions. For example, they enable log ins, the setup of mail aliases, and the display of a login message. Most system files are created during the system installation; however, you can create or modify files after the system has been installed.

This chapter contains descriptions, sample entries, and instructions for modifying the following system files:

- Password File (`/etc/passwd`)
- Group File (`/etc/group`)
- Terminal Initialization File (`/etc/ttys`)
- File System Table (`/etc/fstab`)
- Mail Aliases File (`/usr/lib/aliases`)
- Clock Table Daemon (`/usr/lib/crontab`)
- Message-of-the-Day File (`/etc/motd`)

B.1 The Password File

The password file, `/etc/passwd`, is a data file that contains an entry for every user who has login privileges on your system. Each entry in the `/etc/passwd` file has fields that specify the following information:

- User login name
- User password
- User identification number
- Group identification number
- A description of the user
- The pathname of the user home directory
- The pathname of the default shell or command to be executed immediately following login

Each entry in the `/etc/passwd` file must contain at least the following fields: name, encrypted password, user identification number, and group identification number. Entries that do not include the preceding fields are ignored, or may introduce a security problem on some systems. See the *Security Guide for Administrators* for more information.

Each field in the password file is separated by colons. The format for each entry is as follows:

name:[password]:user-id:group-id:[description]:home-directory:[shell]

name — The first field contains the user's login name (1 to 8 characters). The system uses this name to establish login permission.

password — The second field contains the user's encrypted password. An entry in this field is optional. The system uses this password to verify login permission.

user-id — The third field contains the user's identification number (user ID). The system uses this number to determine a user's identity. Once login permission is granted, the system internally translates the login name to this user ID number and uses it to identify the user's processes and to determine owner access permission to files. This number must be unique for each user and be less than 32000.

group-id — The fourth field contains the primary group identification number (group ID). The system uses this number to determine a user's default group classification. You can set up the permissions for any file so that users with the same *group-id* numbers can access the file, but those users with different *group-id* numbers cannot. Once login permission is granted, the system internally establishes the user's group ID number and uses it in determining group access permission to files. This number must be unique for each group and be less than 32000.

A user can belong to a maximum of 32 groups. The `/etc/passwd` file lists only the user's primary group. You can see the user's secondary groups by displaying the `/etc/group` file.

description — The fifth field contains additional user information. For example: user name, office location, office phone number, and home phone. The user name can be an ampersand (&), which means that the login name and the user name are the same. User's can maintain the *description* field with the `chfn` command. See `chfn`(1) for more information.

home-directory — The sixth field contains the absolute pathname to the user's home (initial working) directory. After establishing the appropriate user and group identification, the system uses this pathname to place the user in the named directory.

shell The seventh field contains the absolute pathname to the command that is to be executed immediately upon conclusion of the login process. This is normally a version of the shell (command interpreter) such as `/bin/csh` or `/bin/sh`. It can also be used to allow the user limited access to the system. For example, by replacing a shell version with the full pathname of a particular command, you can log in using the name of the command. The shell will then run the command and log the user out once the command has been executed.

An entry in this field is optional. If nothing is specified, the system automatically invokes `/bin/sh`.

Following is a sample password file:

```
root:R,r97fsje2oss:0:1:System PRIVILEGED Account:/:/bin/csh
field:Pa9rek3.1l5e:0:1:F S PRIVILEGED Account:/usr/field:/bin/csh
operator:sruF3.9ir,ePw:0:28:Operator PRIVILEGED Account:/opr:/opr/ops
guest:n3Rel9s22:10:33:Guest account:/tmpguest:/bin/date
jjd::34:10:John Joseph Doe:/usr/staff/jjd:/bin/csh
jws::24:10:John Walter Smith:/usr/staff/jws:/bin/csh
```

If you are using the Yellow Pages service on your system, see the *Guide to the Yellow Pages Service* for more information on the password file. While the format of the file is similar, differences exist. If you are using the BIND service, see the *Guide to the BIND/Hesiod Service.*

B.1.1 Modifying the Password File

The system automatically creates a generic `/etc/passwd` file during the installation, but you can modify this file to include site and user specific information. By default, the `/etc/passwd` file is read only. Only the superuser can modify any field in the password file using system commands. Once logins are enabled, registered users can modify only their password, description, and shell fields using system commands.

Note

Avoid using a text editor to edit the password file. A text editor does not perform the necessary processing and locking to keep your password file secure. Use the `vipw` command to edit the password file.

The following commands enable you to modify the password file while performing the necessary protection and locking of the file:

`vipw`	Enables the superuser or root to edit any field in the password file.
`adduser`	Adds new accounts to the password file.
`removeuser`	Removes accounts from the password file.
`passwd`	Enables users to change the password field.
`chsh`	Enables users to change the login shell field.
`chfn`	Allows users to change the description field.

The following sections discuss these commands in detail. For additional information, see the *ULTRIX Reference Pages.*

B.1.1.1 Editing the Password File

To ensure proper locking and processing of the password file, use the `vipw` command to edit the password file. The `vipw` command enables you to edit any field in the password file. You must have root privileges to use the `vipw` command.

To use the `vipw` command, type the following:

```
# vipw passwd
```

The `vipw` command invokes the `vi` editor unless the environment variable EDITOR indicates an alternate editor. If the password file is being modified by another user, the following message is displayed:

```
vipw: password file busy
```

When you exit the editing session, the `vipw` command performs a number of consistency checks on the password entry for root, and does not enable a password file with a corrupted root entry to be installed.

B.1.1.2 Adding User Accounts to the Password File

The `adduser` command is an interactive facility that enables you to create accounts for new users on the local system. In addition to adding new users to the password and group files, the `adduser` command sets up a home directory with the generic startup files and creates a `bin` subdirectory.

To invoke the `adduser` command, type the following at the system prompt:

```
# /etc/adduser
```

See Chapter 4 for step-by-step instructions on using the `adduser` command. If you are using the Yellow Pages service, see the *Guide to the Yellow Pages Service* for information on adding users. If you are using the BIND services, see the *Guide to the BIND/Hesiod Service.* Additionally, the *Security Guide for Administrators* contains detailed information on the `passwd` file and the `/etc/auth` database.

B.1.1.3 Removing User Accounts from the Password File

The `removeuser` command is an interactive facility that removes user accounts from the password file, and optionally deletes the user's home directory and files. This command does not alter the group file; hence, you must edit the `/etc/group` file to remove a user from a group. For more information on the group file, see Section B.2.

Use the following steps to remove a user's account:

1. Invoke the `removeuser` command:

   ```
   # /etc/removeuser
   ```

2. Type the user's login name:

   ```
   Enter login name for user to be removed: tippet
   ```

   In this example, the login name `tippet` is entered. The `removeuser` command searches the password file for the user name.

If the user name exists, the password entry for that user name is displayed. For example, this is what the entry in /etc/passwd looks like:

```
tippet::543:15:Carl A. Tippet:/usr/user1/tippet:/bin/csh
```

If the user name is incorrect, the command facility displays a message stating that the user does not exist in the `/etc/passwd` file, then exits.

3. Indicate whether or not you want to delete the displayed entry:

```
Is this the entry you wish to delete?  y
Working ...

User tippet removed.
```

If you type `y` for yes, as in this example, the user account is removed. If you type an `n` for no, the command facility returns you to the system prompt and the password entry remains unchanged.

4. Determine whether you want to remove the user's home directory, subdirectory, and files:

```
Do you want to remove tippet's home directory,
all subdirectories and files (y/n)?  y

You should have backed up tippet's files if you do not wish to lose them.

Are you sure that you want to remove tippet's files (y/n)? y

Deleting /usr/user1/tippet
```

If you type `y` for yes, the command facility verifies that you want to remove the files, as in the previous example, then returns you to the system prompt. If you type `n` for no, the command facility saves the files, and returns you to the system prompt. The `removeuser` session is complete at this time.

B.1.1.4 Changing the User Password

If you are logged in as root, you can change any user's password field in the password file using either the `vipw` command or the `passwd` command. After logins are enabled, registered users can change their own password fields using the `passwd` command only.

The `passwd` command changes or adds a password field. A user password must contain between 6 and 16 characters comprised of at least 3 different characters. For example, the password `ababab` is not acceptable. To use this command facility, invoke the password facility, type the old password and the new password, then retype the new password verify its accuracy as follows:

```
% passwd
Old password:
Enter new password:
Verify:
```

The password is not echoed to the terminal screen for security reasons. If you use the `passwd` command on a hardcopy terminal, dispose of the printout when you have completed your session. If you are using the Yellow Pages Service, see the *Guide to the Yellow Pages Service* for information on changing the user password.

B.1.1.5 Changing the Login Shell

To change the login shell field listed in the /etc/passwd file, use the chsh command. This command checks the password file for your current login shell, displays the current shell, and allows you to type a different shell. For example:

```
% chsh
Changing login shell for tippet
Shell [/bin/csh]: sh
```

If your current shell or the new shell is not listed in the /etc/shells file, the entry remains unchanged.

B.1.1.6 Changing the Description Field

To change the description field in the password file, use the chfn command. The description field in the password file is used by the finger command, and other programs. The chfn command prompts you for the following information:

- User name
- Office number
- Office phone
- Home phone

The chfn command displays defaults for each entry in brackets. To accept the default, press the Return key. To leave an entry blank, type none at the prompt. Entries in the description field cannot contain colons, commas, or control characters; however, phone numbers may be entered with or without hyphens. To invoke the chfn command, type the following:

```
% chfn
Changing finger information for tippet
Name [Carl A. Tippet]: Return
Office number [GRR-13/DH]:
MMM-11/DH
Office phone []:
666-8888
Home phone [555-5555]:
none
```

If you are logged in as the superuser or root, you can change another users description by specifying the command and user's login name on the same command line. For example:

```
% chfn loginname
```

B.2 The Group File

The system group file, /etc/group, contains data for groups and group members. This data file allows users with different group identification numbers (group IDs) to access common files. The system uses the /etc/group file to establish access permissions to files created specifically by group members. A user can be assigned to a maximum of 32 groups. Each entry in the /etc/group file contains four fields. Each field is delimited by a colon, and the items in the fourth field are further delimited from each other by commas. A group file entry cannot exceed 1024 characters. The format of the /etc/group file is as follows:

group:password:group-id:name,name...

group	The name of the group.
password	The encrypted password. When this field is not used or when creating a new entry, place an asterisk (*) in this field to eliminate group password matching.
group-id	The group identification number. The system uses this number to determine group access permissions to files. The users' primary *group-id* is listed in the `/etc/passwd` file. The users' secondary groups is listed in the `/etc/group` file. The *group-id* must be unique and be less than 32000.
name	The login names of the current group members. Each name is delimited from the other by a comma. A group file entry can have as many as 200 members, provided that the entire entry does not exceed 1024. Member names can be continued on the next line.

The following example shows the contents of a group file:

```
clowns:*:25:brown,green,white
tigers:*:53:austin,wake,martinez
angels:*:47:howell,baskerville,tillman,wake
```

For more information on the group file, see the `group`(5) reference page in the *ULTRIX Reference Pages.*

B.2.1 Modifying the Group File

To modify the group file, `/etc/group`, you can use a standard text editor or, if you are adding only a new group, you can use the `addgroup` command. The next two sections discuss how to edit the group file using a standard text editor and how to use the `addgroup` command.

Note that the `adduser` command also allows you to add new groups to the group file when you are creating new user accounts in the password file. For more information on the `adduser` command, see Section B.1.1.2.

B.2.1.1 Editing the Group File

By editing the group file using a standard text editor, you can do the following:

- Add new users.
- Create a new group entry.
- Remove a member from a group.
- Remove a group entry.

To add a new entry to the group file, open the file with an editor, and add the information that defines the new group. For example, if you wanted to add a group called diners, you can create the entry as follows:

```
diners:*:79:lapin,johnson,howell
```

The group name is diners, the password field is marked with an asterisk (*) to eliminate password matching, the group id is 79, and the members of the group are lapin, johnson, and howell.

To add a new user to the group file, edit the `/etc/group` file and add the new name to the fourth field of the group entry. A comma must separate each entry in the

fourth field. For example, if you added `ellis` to the diners group, the entry would appear as follows:

```
diners:*:79:lapin,johnson,howell,ellis
```

To delete an existing group from the `/etc/group` file, open the file with an editor and remove the entry that defines the group you want removed. To remove a current member from a particular group, open the file with an editor, and delete the member's login name and delimiting comma from the group entry.

B.2.1.2 Adding a Group to the Group File

To add groups to the group file, you can either use a text editor, as described in Section B.2.1.1, or you can use the `addgroup` command. The `addgroup` command is an interactive facility that lets you add a new group and group ID to the password file. The new group name must be unique to the group file. The `addgroup` command automatically checks the group IDs and displays the next available group ID in brackets. To accept the default group ID, press the Return key. For example:

```
Enter group name for new group: moo
Enter group id for the new group [79]: Return
```

Using the information from the previous example, a new entry would be created as follows:

```
moo:*:79
```

To add members to the group, edit the group file as described in Section B.2.1.1.

B.3 The Terminal Initialization File

On ULTRIX systems, a terminal special file is created for each terminal connected to the system. Terminal special files are listed in the `/dev` directory. Each special file listed in the `/dev` directory has an entry in the terminal initialization file, `/etc/ttys`. The entry contains information used by various routines to initialize and control the use of the terminal. This file can be modified at any time.

Each entry in the terminal initialization file contains five fields separated by spaces or tabs. An entry cannot exceed 512 characters. A field that contains more than one word must be enclosed in quotation marks (""). Unspecified fields default to the empty string or zero, as appropriate. The format of each entry is as follows:

name command type status window="string" description

name This field contains the name of the terminal special file as listed in the `/dev` directory. All terminals except network pseudoterminals, workstation pseudoterminals, and modems use the following naming convention:

tty [0-9 | A-Z] [0-9]

Network pseudoterminals use the following naming convention:

tty [pqrstu] [0-9 | a-f]

The modem (dialup) lines have the following convention:

ttyd [0-9 | a-f]

command This field contains the name of the command to be executed each time the terminal is initialized. If the command has arguments, the command and arguments must be enclosed in double quotes.

The command, `/etc/getty`, is often listed in this field. This command performs such tasks as baud-rate recognition, reading the login name, and calling `login`(1). This field can also contain other commands, such as the startup command for a window system terminal emulator, or a command to maintain other daemon processes.

If the terminal is a pseudodevice, this field should contain the string `none`.

type This field contains the type of terminal connected to the terminal special file; for example, a vt100. The possible terminal types are listed in the third field of the `/etc/termcap` file on your system. If the terminal is a pseudodevice, specify the word `network` in this field.

status This field specifies the status for each terminal line. When terminals are initialized, these status values are used by the `init` command. See `init`(8) in the *ULTRIX Reference Pages*. A terminal line can have up to four status values. They are as follows:

`off` or **`on`**
The status bit, `on` enables logins for the terminal. The status bit `off` disables logins for the terminal. If this flag is not set, logins are disabled. For pseudodevices, do not specify this status flag.

`modem` or **`nomodem`**
If `modem` is specified, the terminal line recognizes modem signals (dial-in and dial-out). If `nomodem` is specified, the terminal line ignores modem signals. The default is `nomodem`.

`secure`
If `secure` is specified, logging in is enabled on this terminal line for root; the flag `on` must also be set. If `secure` is not specified, root cannot log in on this terminal line.

`shared`
If the terminal is `shared`, the terminal line can be used for both incoming and outgoing connections. If this status field is left blank, the line cannot be used for incoming and outgoing connections.

For example, if you specify **shared**, `login`, `tip`, and `uucp`, can use the same terminal, though not simultaneously. If this status field remains blank, logins require one terminal, and `tip` and `uucp` require a different terminal. See `tip`(1) and `uucp`(1c) in the *ULTRIX Reference Pages* for more information.

window="string"

This field is present on workstations running ULTRIX Worksystem Software. The *string* contains the name of the X server for your worksystem so that the X server will start up when you log into your workstation. For example:

```
window="/usr/bin/Xqvsm"
```

For more information, see X(1X) and Xqvsm(1X) in the *ULTRIX Reference Pages.*

description This field contains comments. Comments may appear anywhere in the terminal initialization file, as long as they are preceded by a number sign (#). The system ignores all comments.

Example B-1 displays the contents of a terminal initialization file with entries that correspond to the initial configuration. In this example, the initial configuration consists of 8 dmf0 lines, 8 dmfl lines, and 32 network pseudoterminal lines.

Example B-1: Contents of an /etc/ttys File

```
#                "@(#)ttys     4.1     (ULTRIX)        1/23/90"
#
#
#
# name  getty           type            status          comments
#
#
#  The dmf0 lines are here:
#
tty00   "/etc/getty 2" vt100    on nomodem secure       # direct connect tty
tty01   "/etc/getty 2" vt100    off nomodem secure      # printer
tty02   "/etc/getty 2" vt100    on nomodem              # direct connect tty
tty03   "/etc/getty 2" vt100    on nomodem              # direct connect tty
tty04   "/etc/getty 2" vt100    on nomodem              # direct connect tty
tty05   "/etc/getty 2" vt100    on nomodem              # direct connect tty
tty06   "/etc/getty 2" vt100    off nomodem             # unused - spare
tty07   "/etc/getty 2" vt100    off nomodem             # unused - spare
#
#  The dmf1 lines are here:
#
tty08   "/etc/getty 2" vt100    off nomodem             # unused
tty09   "/etc/getty 2" vt100    off nomodem             # unused
tty10   "/etc/getty 2" vt100    off nomodem             # unused
tty11   "/etc/getty 2" vt100    off nomodem             # unused
tty12   "/etc/getty 2" vt100    off nomodem             # unused
tty13   "/etc/getty 2" vt100    off nomodem             # unused
tty14   "/etc/getty 2" vt100    off nomodem             # unused
tty15   "/etc/getty 2" vt100    off nomodem             # unused
#
#  The network pty0 pseudodevice lines are here:
#
ttyp0   none            network
ttyp1   none            network
ttyp2   none            network
ttyp3   none            network
ttyp4   none            network
ttyp5   none            network
ttyp6   none            network
ttyp7   none            network
ttyp8   none            network
ttyp9   none            network
```

Example B-1: (continued)

```
ttypa   none            network
ttypb   none            network
ttypc   none            network
ttypd   none            network
ttype   none            network
ttypf   none            network
#
#  The network pty1 pseudodevice lines are here:
#
ttyq0   none            network
ttyq1   none            network
ttyq2   none            network
ttyq3   none            network
ttyq4   none            network
ttyq5   none            network
ttyq6   none            network
ttyq7   none            network
ttyq8   none            network
ttyq9   none            network
ttyqa   none            network
ttyqb   none            network
ttyqc   none            network
ttyqd   none            network
ttyqe   none            network
ttyqf   none            network
```

For further information on the `/etc/ttys` file, see `getttyent`(3), `gettytab`(5), `termcap`(5), `ttys`(5), and `getty`(8) in the *ULTRIX Reference Pages.*

B.3.1 Modifying the Terminal Initialization File

To modify any field in the terminal initialization file, `/etc/ttys`, use a standard text editor. In most cases, the terminal initialization file is modified for operations such as the following:

- Adding or removing a terminal
- Enabling or disabling logins to a specific terminal line
- Enabling or disabling modem recognition (dial in and dial out)
- Setting bit recognition for 7- or 8-bit mode
- Changing the baud rate for a terminal

The following sections discuss these topics in more detail. Additionally, describes how to process the terminal initialization file after changes have been made.

B.3.1.1 Adding or Removing an Entry

To add a new entry to the terminal initialization file, include the name of the terminal special file, the command that you want executed for that terminal, the type of terminal, and the status of that terminal line. For example:

```
tty01     "/etc/getty std.9600"     vt100     on
```

In the previous example, the terminal special file is named tty01, the `getty` command sets 7-bit recognition at a 9600 baud rate, the terminal type is a VT100, and login is enabled by the `on` status flag.

To remove a terminal line from the terminal initialization file, edit the file using a standard text editor and delete the entry from the file.

B.3.1.2 Enabling or Disabling Logins

To enable root login on a VT100 terminal, include the following entry:

```
tty02  "/etc/getty std.9600"  vt100 on secure
```

Both the `on` and `secure` status flags must be set to enable login for the root user.

To disable login, set the status bit to `off`.

B.3.1.3 Enabling or Disabling Modem Recognition

To allow modem (dial up) access at 1200 baud on a VT220 without root login privileges, include the following entry:

```
tty01  "/etc/getty std.1200"   vt220 on modem
```

To disable modem access, change the status bit to `nomodem` as follows:

```
tty01  "/etc/getty std.1200"   vt220 on nomodem
```

The baud rate is set by the `getty` command.

B.3.1.4 Setting Bit Recognition

To permit 8-bit recognition at 9600 baud on a VT100 with logins enabled for root, use the following format:

```
tty03  "/etc/getty 8bit.9600" vt100  on secure
```

Note, if a terminal is set up to operate in 8-bit mode and the command field does not specify an 8-bit entry, output to the terminal is displayed as multinational characters.

To permit 7-bit recognition at 9600 baud on a VT100 with logins enabled for root, use the following format:

```
tty03  "/etc/getty std.9600"  vt100  on secure
```

In both examples, 7-bit and 8-bit recognition are set by the `gettytab` entry.

B.3.1.5 Setting the Baud Rate

The baud rate is set by the `getty` command. For example, to set the baud rate to 9600 for a VT220 terminal, use the following format:

```
tty03  "/etc/getty std.9600"  vt220  on secure
```

To change the baud rate, for example, from 9600 to 1200, change the `getty` entry as follows:

```
tty03  "/etc/getty std.1200"  vt220  on secure
```

B.3.1.6 Processing the Terminal Initialization File

Changes made to the terminal initialization file can either be processed during the boot process or during multiuser mode. To process changes during multiuser mode, use the `kill` command as follows:

```
# kill -HUP 1
```

The `kill` command sends a hangup signal to the `init` command, which causes `init` to rescan the `/etc/ttys` file. If changes are found in the file, `init` processes those entries that have been modified. For more information, see the `kill`(1) and `init`(8) in the *ULTRIX Reference Pages*.

B.4 The File System Table

The file system table, `/etc/fstab`, contains an entry for each known file system. These entries provide descriptive information on each file system. The order of the entries is important, because other programs (such as `dump`, `mount`, and `fsck`) must access this information in sequential order.

Each entry in `/etc/fstab` contains seven fields of information, delimited by colons. As defined in the `/usr/include/fstab.h` file, the format of the `/etc/fstab` file is as follows:

spec:file:type:freq:passno:name:opts:

spec This field defines either the file system's block special file (device) name, or a remote file system like Network File System (NFS). For example, `/dev/rz0a` defines a local system and `/usr/src@erie` defines a remote system.

file This field defines the absolute pathname to the directory on which the file system is mounted. The `mount` command uses this information as the default.

type This field specifies the file system mode: `rw` (read-write), `ro` (read only), `rq` (read-write with quotas), `sw` (swap), and `xx` (ignore).

If `sw` is specified and if the file system has been configured for such use, `swapon` (invoked by `/etc/rc`) makes that file system part of the system swap space. If `xx` is specified, the local file system is ignored, that is, it is not processed by `mount`, `dump`, or `fsck`.

freq This field specifies the file system dump frequency (every *n*th day). This is the default order for the `dump` command. For NFS entries, this field should contain a `0` value.

passno This field defines the file system pass number. This is used as the default order for the `fsck` command. Usually, only the root file system has a pass number of 1. The remaining file systems should be assigned higher pass numbers, which enables the `fsck` command to simultaneously check file systems in parallel.

Section B.4.1.2 discusses this field in more detail.

Note that NFS entries should have pass numbers of `0` so that local execution of certain commands, such as `fsck` and `dump` ignore these entries. This ensures that you do not interfere with remote file systems maintained by another site.

name This field specifies the type of file system you are mounting. Supported file systems are UFS and NFS.

opts This field defines file system-specific options that are being passed to the file system being mounted.

The following example displays the contents of an /etc/fstab file:

```
/dev/ra0a:/:rw:1:1:ufs::
/dev/ra1g:/usr:rw:1:2:ufs::
/usr@bigvax:/bigvax:rw:0:0:nfs::
/usr/uws2.0@bigvax:/usr/uws2.0:rw:0:0:nfs:soft,bg,nosuid:
/usr/dec@bigvax:/usr/dec:rw:0:0:nfs:bg,soft,nosuid:
/usr/pro/xyz@vax:/usr/pro/xyz:rw:0:0:nfs:bg,soft,intr,nosuid:
```

B.4.1 Modifying the File System Table

By convention, the /etc/fstab file is created and maintained as a read-only file. Consequently, only the superuser can modify it. Typically, this file is modified for the following reasons:

- To add a new file system
- To remove an obsolete file system
- To change the order in which file systems are loaded, dumped, or checked
- To import a file system with NFS

The following sections discuss these topics in more detail.

B.4.1.1 Adding or Removing a File System

To add a new file system to the /etc/fstab file, open the file with a standard text editor. You need the following information to create an entry for the new file system:

- Block special file (device) name
- Absolute pathname to the directory where the file system is located
- Protection mode of the file system
- File system dump number to determine the frequency of dumps
- Pass number supplied for the fsck command to determine how the file systems are checked
- The name of the supported file systems (UFS or NFS)

The following example shows an entry for a local file system:

```
/dev/ra0a:/:rw:1:1:ufs::
```

The block special file (device) name is /dev/ra0a. The absolute pathname is the root directory and the file system is mounted read-write (rw). This file system is dumped daily as signified by the number 1, and it is checked by the fsck command on the first pass as indicated by the number 1.

The following example shows an entry for a remote file system:

```
/usr/jwn@minn:/usr/jwn:ro:0:0:nfs:bg,soft:
```

The block special file (device) name is /usr/jwn@minn. The absolute pathname is /usr/jwn and the file system is mounted read only. The dump field and the pass number fields are defined as zero (0) so that locally executed commands ignore this entry in the /etc/fstab file.

To remove a file system entry from the /etc/fstab file, open the file with a standard text editor and delete the line. Make certain this does not affect the logical ordering of the other file systems.

B.4.1.2 Changing the Order of File Systems in the /etc/fstab File

The mount, dump, and fsck commands process /etc/fstab entries in order, according to the sequential listing of the entries and the pass field. Consequently, the order of the entries in the /etc/fstab table is important. This section describes how the mount, dump, and fsck commands use the information in the /etc/fstab file.

The fsck command performs a file consistency check on local file systems. If the fsck command notes any inconsistencies, it attempts to correct the file system before continuing. The fsck command makes a number of passes, often inspecting groups of disks in parallel. Hence, all file systems on a single disk should have a different pass number because the fsck command can check file systems on the different disks at the same time. To determine which file systems to check in each pass, you must supply a pass number in the /etc/fstab file. The root file system should be checked on pass 1, while other root file systems such as partition a should be checked on pass 2. Other small file systems can be checked on separate passes. For example, d file systems can be checked on pass 3, and e file systems can be checked on pass 4. Large user file systems should be checked on the final pass. Those file systems that are NFS mounted, or that you do not want checked, should have a pass number of zero (0). Any file system mounted with read-write (rw) or read only (ro) are not checked.

The mount command determines which and how to mount local and remote file systems from the entries in the /etc/fstab file. Entries in the /etc/fstab file are read sequentially; therefore, list the file systems as you want them mounted. For example, the mount command fails if it is directed to mount a file system on a directory that has not been mounted. Physically write-protected disks and magnetic tape file systems must be mounted read only (ro) or an error occurs at mount time. Mounting a corrupted file system can cause the system to crash.

In addition to specifying which file systems to mount, you can also specify options when creating /etc/fstab entries for remote file systems. The NFS options direct the mount command to retry a failed mount operation, allows hard mounted file systems to be interrupted, prevents binaries from being executed on a specified file system, and sets the size of the read and write buffers.

The dump command saves a copy of all files changed after a certain date. Using the /etc/fstab file, you can specify how frequently local file systems may be backed up. For example, the number one (1) in the *freq* tells the dump command to back up that particular file system on a daily basis; the number ten (10) tells the dump command to perform a back up every 10 days. For NFS file systems, the *freq* field should contain a zero (0).

See the *ULTRIX Reference Pages* for more information on fsck(8), mount(8), and dump(8).

B.4.1.3 Importing a File System

To mount remote directories or file systems each time your system enters multiuser mode, place an entry in the `/etc/fstab` file as described in Section B.4. By placing an entry in the `/etc/fstab` file, you can automatically mount remote file systems from any NFS server; however, you must also manually create the mount point. Chapter 2 describes how to use the `nfssetup` utility to import a file system. For detailed information on importing file systems, see the *Guide to the Network File System.*

B.5 The Aliases File

The aliases file, `/usr/lib/aliases`, contains information that the `sendmail` utility uses to route messages to a group of one or more users. Each entry in `/usr/lib/aliases` contains two fields of information separated by a colon. The format of the `/usr/lib/aliases` file entries is as follows:

alias:user,user...

alias This field contains the name of the group to which messages are routed.

user This field contains the list of group members (user login names), separated by commas. When an `alias` is supplied, all user login names included in the entry receive the same mail message. The list of group members can extend beyond one line.

To add comments to the aliases file, include a number sign (#) followed by the comment in first column of the file. The following example shows the contents of an aliases file:

```
# The friends alias is an exclusive group.
friends:barry,candida,martinez
# The team alias lists all project leaders
team:lisa,anthony,terry,alice
# The research alias lists all team members, including project leaders
research:martinez,waker,ellis,artis,wall,lisa,anthony,terry,alice
```

The following sections discuss modifying the `/usr/lib/aliases` file, and how to process new entries in the file.

B.5.1 Modifying the Aliases File

To modify the aliases file, use a standard text editor. You must edit this file when you want to add new members to a group, remove members from a group, create a new entry, or remove an entry.

To add a new member to a group or create a new entry, open the file using a standard text editor. You can then add the name of a new group member to an existing group, separating the user login name by a comma, or you can create a new entry using the format described in the Section B.5.

To remove a group member from a group, open the file using a standard text editor, then delete the name and separating comma from the entry. To remove an entire alias, open the file, then delete the entire entry.

B.5.2 Processing the Aliases File

To process the alias file, use the `newaliases` command. This command allows new additions to the aliases file to become a part of the `sendmail` aliases database. Use the `newaliases` command as follows:

```
# newaliases
```

This command reads the new information added to `/usr/lib/aliases` and rebuilds the `sendmail` aliases database.

For further information, see `newaliases`(1), `aliases`(5), and `sendmail`(8) in the *ULTRIX Reference Pages.*

B.6 The Clock Daemon Table

The clock daemon table, `/usr/lib/crontab`, is a symbolically linked file that contains routine commands which the system clock daemon, `cron`, executes at the specified dates and times. For example, `/usr/lib/crontab` might contain routine backup commands as well as commands that cause the automatic removal of outdated or unused temporary files.

Once invoked during multiuser startup, the system activates the system clock daemon, `cron`, every 60 seconds. In turn, the system clock daemon executes those commands listed in the `/usr/lib/crontab` file that are scheduled for that time. Each entry contains six fields of information separated by spaces. The format of the entries in the `/usr/lib/crontab` file is as follows:

minute hour day month weekday command

minute	The exact minute that the command sequence is to be executed. The *minute* variable can be 0 through 59.
hour	The hour of the day on which the command sequence is to be executed. The *hour* variable can be 0 through 23.
day	The day of the month on which the command sequence is to be executed. The *day* variable can be 1 through 31.
month	The month of the year on which the command sequence is to be executed. The *month* variable can be 1 through 12.
weekday	The day of the week on which the command sequence is to be executed. The *weekday* variable can be an integer from 1 to 7. Monday equals 1 and Sunday equals 7.
command	The command sequence that is to be executed. The *command* variable should contain the complete command sequence.

In addition, the first five fields may specify either a single time indicator, a multiple time indicator, a time range, or an asterisk. A single time indicator may consist of one or two consecutive digits such as 3 or 33. A multiple time indicator consists of a string of indicators separated by commas, such as 5,10,15,20. A time range consists of two indicators separated by a dash, such as 5-20. An asterisk field entry represents all times.

The following examples display the partial contents of a /usr/lib/crontab file:

```
# periodic things
0,15,30,45 * * * * (echo '^M' `date`; echo '') >/dev/console
0,15,30,45 * * * * /usr/lib/atrun

# daily stuff
5 4 * * * sh /usr/adm/newsyslog
15 4 * * * ( cd /usr/preserve; find . -mtime +7 -a -exec rm -f {} ; )
20 4 * * * find /usr/msgs -mtime +21 -a ! -perm 444 -a ! -name bounds
     -a -exec rm -f {} ;

# NOTE: The above line is wrapped.

# local cleanups
30 4 * * * find /usr/spool/mqueue -type f -mtime +5 -name df -exec rm {} ;
35 4 * * * find /usr/spool/mqueue -type f -mtime +5 -name tf -exec rm {} ;
40 4 * * * find /usr/spool/rwho -type f -mtime +21 -exec rm {} ;
#
```

The next two sections discuss the cron command and describe how to modify the /etc/usr/crontab file.

B.6.1 Specifying cron

The cron command executes at specified dates and times, according to the information in the /usr/lib/crontab file. The cron command never exits; hence, it should only be executed once to avoid using up system resources. For the best results, run the cron command from the initialization process by including it in the /etc/rc file. For more information, see the init(8) command in the *ULTRIX Reference Pages.*

B.6.2 Modifying the Clock Daemon Table

Each entry in /usr/lib/crontab contains information that specifies a time and command sequence that is to be executed regularly; hence, stagger the crontab entry times so that the processes are not running at together. When appropriate, and especially during anticipated periods of heavy user activity, include the nice command in your crontab file entries. The nice command tells cron to execute commands at a lower priority.

To change the clock daemon table, use a standard text editor to open and edit /usr/lib/crontab. For further information on the clock daemon and prioritizing tasks, see cron(8) and nice(1) in the *ULTRIX Reference Pages.*

B.7 The Message-of-the-Day File

The message-of-the-day file, /etc/motd, provides system users with information relevant to each day's operation. The message-of-the-day file is displayed on the terminal screen after each login.

The /etc/motd file does not have a special format; however, the first line of the message-of-the-day file is controlled by the /etc/rc.local file. This line contains the adjective ULTRIX, the current version of the operating system, revision number, and the day's date. To modify the motd file, use a standard text editor; you must have root privileges.

The system displays the same message after each login until you either modify or delete the contents of /etc/motd. The following example displays the contents of an /etc/motd file:

```
Ultrix 4.2A (Rev 162) System #8: Fri Jan 11  10:45:34 EST 1991

-------------------------------------------------------------------

          This machine is running ULTRIX Version 4.2A

-------------------------------------------------------------------

Tomorrow, October 31, 1991, this machine will be unavailable
from 12pm to 1pm.  Field Service is performing some maintenance
tests.

-------------------------------------------------------------------

        If you encounter problems or have questions
        concerning this machine, send mail to the
        admin account.
```

Support of the CI/HSC Hardware C

The CI is a high speed, dual-path bus that connects processors and intelligent I/O subsystems (HSCs) in a computer room environment. The HSC is a self-contained, intelligent mass storage controller that provides access to disks and tapes from multiple host nodes attached to the CI bus.

Note

The ULTRIX implementation has the following limitations:

- A maximum of four HSC controllers may be attached to a CI bus.
- A single CI bus may be attached to a host.
- Under no circumstances can an ULTRIX node participate as a VMS cluster member. A configuration which includes a VMS system and an ULTRIX system residing on the same CI is not supported.

This version of ULTRIX supports Digital's System Communication Architecture (SCA) for CI port adapters and HSC controllers. SCA implements port and class driver support, and standardizes the ways in which TMSCP (tms) and MSCP (ra) devices are handled. SCA separates functionality into different architectural layers, thus minimizing the effect that software changes to one layer have on other layers.

C.1 Hardware Setup and Restrictions

For information on physical components and setup, refer to the HSC hardware documentation and the hardware documentation for your processor and supported devices. Only processors with CI adapters can support HSC configurations.

When setting up the HSC hardware, you should attach a terminal to the HSC in order to use commands to get/set HSC parameters, monitor connections between remote systems, and identify the disk/tape status.

The maximum number of hosts on a CI is 16. The host number for any host on the CI must be between 0 and 15; however, if the broadcast address has been set to 0, then 0 cannot be a host number.

Note

Two parameters of particular importance are the system ID and the system name. Use the HSC `SET` command to specify these parameters. Do not duplicate any system identification or names of nodes on the star coupler.

C.2 Software Installation and Restrictions

The installation software assists you in identifying and configuring the components of your system. You should be familiar with the Basic Installation Guide for your processor before starting the actual installation.

During installation of the ULTRIX software, each accessible MSCP (ra) disk device must be uniquely identified by its unit plug number:

- The unit plug number must be between 0 and 254 inclusive.
- Each unit plug number must be unique. Two disks cannot have the same unit plug number even if the disks are on separate controllers. For example, if the unit plug number for a disk on controller A is 5, and the unit plug number for a disk on controller B is also 5, you must change one of the numbers.

After installation, the unit plug numbers can be between 0 and 254 inclusive, and they need not be unique in cases where the disks are on separate controllers.

The CI network device (`scs0`) is not configured by default. The network setup installation script gives you the option to install or not.

C.2.1 Hardware Revision Levels

The correct operation of the software subsystems is sensitive to the revision levels of the CI/HSC microcode. In particular, the following microcode levels should be installed if they are not already:

- HSC microcode level should be V3.9A or higher.
- HSC microcode level V500 or later is needed to support the TA90E and to use the exclusive access functionality provided by the `-e` and `-n` options of the `radisk`(8) utility. See the *ULTRIX Reference Pages* for more information on this utility.
- HSC tape interface boards should have microcode at level 26 or higher.
- HSC disk interface boards should have microcode at level 39 or higher.

Note

This version of ULTRIX, does not support the new CI CISCE 24-node upgrade. The CI microcode distributed by ULTRIX does not support rev. 20 link modules. As a result, the system will be unable to load and verify the CI functional microcode.

C.3 Configuration File Entries

The installation software ensures that your HSC components are configured into the kernel and included in the system configuration file, `/usr/sys/mips/conf/`*HOSTNAME*, for RISC systems, and `/usr/sys/vax/conf/`*HOSTNAME*, for VAX systems, where *HOSTNAME* is your system's name, in uppercase letters.

The *Guide to Configuration File Maintenance* provides information on the following entries that correspond to a CI/HSC configuration:

- Description of the `scs_sysid` parameter
- CI adapter specifications
- Controller and device specifications
- The `scsnet` pseudodevice definition

C.4 Booting an HSC Controller or an HSC Disk

If an HSC controller fails, any disks connected to that HSC become inaccessible. Attempts to access those disks will cause the accessing system to hang until the HSC reboots completely.

The ULTRIX software supports booting an HSC disk on all VAX processors with the exception of the MicroVAX class of system. The *Guide to System Shutdown and Startup* provides explicit instructions for booting an HSC disk on each processor that supports an HSC configuration.

C.5 Sharing Disk/Tape Units Among Several Hosts

Although an HSC can be shared among several hosts, there is no software interlocking mechanism to prevent concurrent writes to the same partition by multiple ULTRIX systems. The following restrictions must be observed:

- A disk unit can be shared by multiple readers only. Writable file systems cannot be shared.
- If a disk will be shared, it should be hardware write-protected.
- Each host must mount the file systems to be accessed with the read-only (`-r`) option of the `mount` command.
- Only a single host may mount a disk containing writable file systems.
- Use the Network File System (NFS) if multiple writers need to share partitions.

You should coordinate disk unit ownership among the hosts on the CI. For example, assign a range of disk unit numbers to each host. The HSC can also be directed to limit disk access to an exclusive host system. This protects the disk from accidental access by another host on the CI. For more information, see the `-e` and `-n` options for the `radisk`(8) utility in the *ULTRIX Reference Pages.*

Tape drives that are attached to an HSC can be shared. This feature is recommended and provides greater use of tape drives. Be aware that the access mechanism provides serial sharing of the drives, not simultaneous access.

C.6 CI Network Capabilities

ULTRIX also provides host to host communications over the CI through a network driver (`scsnet`). The driver can be accessed through the socket system interface in the same manner as the Ethernet drivers. For a description of the network interface, see `scs`(4) in the *ULTRIX Reference Pages.*

Currently the TCP/UDP/IP protocols are supported. The `scsnet` driver takes full advantage of the block mode capabilities of the CI and is therefore a good means for offloading NFS traffic from the Ethernet. Figure C-1 illustrates how the CI can be used to share disks via NFS.

Figure C-1: Typical CI Configuration

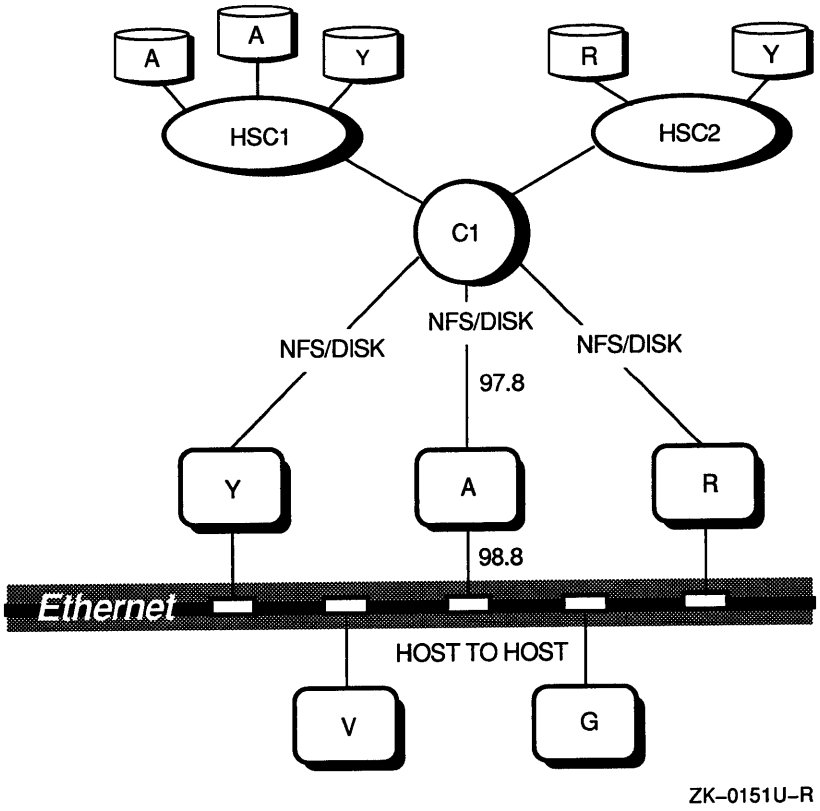


Figure C-1 displays five systems: hosts A, G, R, V, and Y. Hosts Y, A, and R connect to both the CI (net number 97.8) and the Ethernet (net number 98.8). Hosts V and G are connected to the Ethernet only. Note that a separate subset is required for the CI.

Each host shares disks on an HSC. Host A has mounted all of the partitions on the two disks labeled A. Because of the restrictions mentioned above no other system can directly mount A's disks for write access. The only way to obtain access to A's disks is through the NFS. For example hosts Y, and R, could NFS mount A's partitions specifying that the CI path be used instead of the Ethernet. The path is chosen by the `mount(8)` command. For example, the `/etc/hosts` file would have an entry for both paths to system A:

```
"A"      98.8    (for the Ethernet path)
"A-ci"   97.8    (for the CI path)
```

The system manager at system Y would NFS mount a disk partition by specifying host A-ci in the `mount` command instead of host A. For example, the following command would NFS mount the `/usr/users` directory from system A using the CI instead of the Ethernet:

```
# /etc/mount A-ci:/usr/users /mnt
```

The NFS traffic would now be routed over the CI and the other host to host traffic over the Ethernet. (Systems V and G would have to specify the Ethernet path because they are not connected to the CI).

A CI must be configured to all systems. The CI is normally configured at boot time; however, the entry for the CI in the `/etc/rc.local` file must be uncommented and modified to specify the correct broadcast address.

Managing the Print System D

This appendix discusses the makeup of the /etc/printcap file, known as the printer capability database, and explains the various printing parameters that can be defined in that file. This appendix also describes the commands available to maintain the print system.

D.1 The Printer Capability Database

The printer capability database, /etc/printcap, contains a descriptive entry for each printer available on your system. A generic version of the /etc/printcap file is created during the installation process; however, you must edit this file to define each printer's capabilities.

Entries in the /etc/printcap file consist of several fields separated by colons. An entry can span several lines; however, a backslash (\), used as a line continuation character, must appear at the end of each entry to signify that the line is continued. The first line of an entry usually specifies the printer's logical names. Subsequent lines define the capabilities of each printer. If you do not specify a capability, the default is used. Example D-1 shows a printcap entry:

Example D-1: A Printcap Entry

```
lp2|2|lab2|lg01 printer in lab:\        1
        :lp=/dev/lp2          2
        :rm=atlanta                   3
        :rp=lp2               4
        :sd=/usr/spool/lp2                  5
        :mx#0:                6
```

In the previous example, the printcap entries are defined as follows:

1. The first line specifies the logical name (lp) and number (2) of the printer as listed in the /dev directory. This line also contains any other logical names (lab2) for the printer and lists the location of the printer.
2. The **lp** symbol specifies the name of the special file to open for output.
3. The **rm** symbol specifies the machine name of the remote printer as this printer is not connected to the local system.
4. The **rp** symbol specifies the logical name for the printer on the remote system.
5. The **sd** symbol specifies the spooling directory where print requests are queued before printing.
6. The **mx** symbol specifies the maximum allowable file size in blocks. In this instance, zero (0) removes any file size restriction.

The ULTRIX operating system provides you with a file, `/etc/printcap.examples`, which contains examples of printer definitions. By using this file as a template, you can tailor it to suit the needs of your site. To display this file, type the following:

```
# more /etc/printcap.examples
```

The following section lists and describes the available printcap symbols and their defaults if applicable. The format for specifying a printcap entry is as follows:

symbolname=value

D.1.1 Printcap Symbols

This section describes the various printcap symbols that can be defined in the `/etc/printcap` file.

`af`

Specifies the accounting file that tracks the number of pages printed by each user for each printer. Each accounting file associated with a printer must have a unique name. This file must be owned by the print daemon. This symbol cannot be used for remote printer entries. When this symbol is specified, intermediate directories are automatically created as needed.

`br`

Specifies the baud rate for the printer. This symbol must be specified with tty devices (serial lines); however, it has no effect on printers connected to the console port or printers connected by a parallel port.

`cf`

Specifies the output filter for the cifplot data filter. For more information on using filter capabilities, see `lpd`(8) in the *ULTRIX Reference Pages*.

`ct`

Specifies the connection type. Valid connection types are *dev*, *lat*, *remote*, and *network*. The *dev* argument specifies a printer connected to the local system. The *lat* argument specifies a printer connected over the local area. The *remote* argument specifies a printer connected to another system running a compatible printer daemon. The *network* argument specifies a printer that does not use standard output, has its own cpu, address, and node name.

`Da`

Specifies the default data type used by the print daemon. Valid arguments are *ansi*, *ascii*, *postscript*, *regis*, and *tek*. If a data type other than *postscript* is specified, a translator is invoked by the daemon to convert the files in the job to *postscript*. This symbol is only valid with PostScript (TM) printers.

`df`

Specifies the output filter for the TeX data filter (DVI format). For more information on using filter capabilities, see `lpd`(8) in the *ULTRIX Reference Pages*.

`Dl`

Specifies the device control module library file. The default is `/usr/lib/lpdfilters/lps_v3.a`. This symbol is only valid with PostScript (TM) printers.

`dn`

Specifies the name of the `daemon` program that must be invoked when a print request is made of the printer. The default is `/usr/lib/lpd` which should not be changed. This symbol enables you to specify support for line printer daemons other than the default.

`du`

Specifies the daemon UID used by the print spooler programs. The default value is zero (0) which should not be changed. This symbol enables you to define other daemon UIDs for printer daemons other than `/usr/lib/lpd`.

`fc`

Specifies which terminal flag bits to clean when initializing the printer line. All bits should be cleared (fc=0177777) before calling the `fs` symbol. For more information, see the `fs` symbol in this table.

`ff`

Specifies the string to send as a form feed to the printer. The default is `\f`.

`fo`

Specifies whether a form feed is printed when the device is first opened. This is in addition to the normal form feed which is printed by the driver when the device is opened. To suppress all printer induced form feeds, specify this symbol with the `sf` symbol.

`fs`

Specifies which terminal flag bits to set when initializing the printer line. Before calling the `fs` symbol, all bits should be cleared using the `fc` symbol, then the `fs` symbol should be set to specific bits. For detailed information on these bits, see `tty`(4) in the *ULTRIX Reference Pages*. A brief description of each bit follows:

Flag	Value	Description
ALLDELAY	0177400	Delay algorithm selection
BSDELAY	0100000	Select Backspace delays (not implemented)
BS0	0	
BS1	0100000	
VTDELAY	0040000	Select form feed and vertical tab delays
FF0	0	
ff1	0100000	
CRDELAY	0030000	Select carriage return delay

Flag	Value	Description
CR0	0	
CR1	0010000	
CR2	0020000	
CR3	0030000	
TBDELAY	0060000	Select tab delays
TAB0	0	
TAB1	0002000	
TAB2	0004000	
XTABS	0006000	
NLDELAY	0001400	Select new-lines delay
NL0	0	
NL1	00E00400	
NL2	0001000	
NL3	0001400	
EVENP	0000200	Even parity allowed on input (most terminals)
ODDP	0000100	Odd parity allowed on input
RAW	0000040	Raw mode: wake up on all characters; 8–bit interface
CRMOD	0000020	Map CR into LF; echo LF or CR as CR-LF
ECHO	0000010	Echo (full duplex)
LCASE	0000004	Map upper case to lower on input
CBREAK	000000	Return each character as soon as it is typed
TANDEM	0000001	Automatic flow control

`gf`

Specifies the graph data filter (`plot`(3X) format). For more information on using filter capabilities, see `lpd`(8) in the *ULTRIX Reference Pages.*

`ic`

Specifies the driver that supports nonstandard ioctl to an independent printout.

`if`

Specifies the accounting filter. If an accounting filter is specified using the `af` symbol, the `if` symbol is ignored. For more information on using filter capabilities, see `lpd`(8) in the *ULTRIX Reference Pages*. Available print filters are listed below. For detailed information on the individual filters, see Section 8 of the *ULTRIX Reference Pages*.

Filter name	Description
`/usr/lib/lpdfilters/lpf`	Line printer filter (LP25, LP26, LP27, LP29, LG01, LA210, LQP02, LQP03)
`/usr/lib/lpdfilters/lqf`	Letter quality filter (LQP02, LQP03)
`/usr/lib/lpdfilters/ln01of`	LN01 Laser Printer filter
`/usr/lib/lpdfilters/ln03of`	LN03 Laser Printer filter
`/usr/lib/lpdfilters/ln03of`	LN03S Laser Printer filter
`/usr/lib/lpdfilters/lcg01of`	LCG01 Color Printer
`/usr/lib/lpdfilters/lj250of`	LJ250 DEColorwriter filter

`It`

Specifies the default input tray. Valid arguments are *`bottom`*, *`middle`*, *`lcit`*, and *`top`*. An error occurs if the specified input tray is not available on the printer. This symbol is only valid with Postscript (TM) Printers.

`lf`

Specifies the name of the error log file. The default is `/dev/console`. If more than one printer is connected to your system, each printer must have an error log file with a unique name. When the error log file is created, intermediate directories are automatically created as needed.

`Lf`

Specifies the Layup to PostScript (TM) translator program. The default is `/usr/lib/lpdfilters/layup`. This symbol is only valid with PostScript (TM) printers.

`Lu`

Specifies the layup definition file which contains information that can alter the appearance of output such as margins and borders. This symbol is only valid with PostScript (TM) printers.

`lo`

Specifies the name of the lock file used by the printer daemon to control the printing jobs in each spooling directory. The default is lock. This symbol enables you to define a lock file for use by other printer daemons.

`lp`

Specifies the name of the special file to open for output. The default is `/dev/lp` which specifies a parallel printer. Valid special file names include lp1, lp2, lp3, and so on. Serial printers should contain a special file name such as tty0, tty1, tty2, and so on. You must define this entry with a null argument for remote printers.

`mc`

Specifies the maximum number of copies that may be printed using the `lpr` command. See `lpr`(1) in the *ULTRIX Reference Pages.*

`Ml`

Specifies the action to take with user errors detected by the printserver. Valid arguments are *keep* and *ignore*. If you specify *keep*, the messages are recorded in a file and sent to you. If you specify *ignore*, the messages are not recorded. This symbol is only valid with PostScript (TM) printers.

`mx`

Specifies the maximum allowable file size (in BUFSIZE blocks) printable by each user. If this symbol is not specified, 1000 blocks is the maximum allowable file size. If you specify zero (mx=0), the file size restriction is removed.

`nc`

Does not allow control characters to appear in the output file.

`nf`

Specifies a ditroff filter. For more information on using filter capabilities, see `lpd`(8) in the *ULTRIX Reference Pages.*

`Nu`

Specifies the default number of pages on a single sheet. The argument can be a number from 1 through 100. This symbol is only valid with PostScript (TM) printers.

`of`

Specifies the output filter to be used with the printer. Output filters filter text data to the printer device when accounting is not enabled or when text data must be passed through a filter. If the `af` symbol is specified, the `of` symbol is ignored. For more information on using filter capabilities, see `lpd`(8) in the *ULTRIX Reference Pages.* Available output filters are listed below. For detailed information on the following filters, see Section 8 of the *ULTRIX Reference Pages.*

Filter Name	Description
`/usr/lib/lpdfilters/lpf`	Line printer filter (LP25, LP26, LP27, LP29, LG01, LA210, LQP02, LQP03)
`/usr/lib/lpdfilters/lqf`	Letter quality filter (LQP02, LQP03)
`/usr/li/lpdfilters/la75of`	LA75 Dot Matrix Printer filter
`/usr/lib/lpdfilters/ln01of`	LN01 Laser Printer filter
`/usr/lib/lpdfilters/ln03of`	LN03 Laser Printer filter
`/usr/lib/lpdfilters/ln03of`	LN03S Laser Printer filter
`/usr/lib/lpdfilters/lcg01of`	LCG01 Ink Jet Printer filter
`/usr/lib/lpdfilters/lg02of`	LG02 Ink Jet Printer filter
`/usr/lib/lpdfilters/lg31of`	LG31 Line Printer filter
`/usr/lib/lpdfilters/lj250of`	LJ250 Ink Jet Printer filter

`op`

Specifies the object port on a LAT terminal server.

`Or`

Specifies the manner by which pages are printed. Valid arguments are *portrait* and *landscape*. This symbol is only valid with PostScript (TM) printers.

`os`

Specifies the object service on a LAT server. (Not Used)

`Ot`

Specifies the output tray. Valid arguments are *face-up*, *lcos*, *lower*, *side*, and *top*. If the specified output tray is not available on the printer, an error occurs. This symbol is only valid with PostScript (TM) printers.

`pl`

Specifies the page length in lines. The default is 66 lines.

`pp`

Specifies the print command filter replacement. The available filter is `/usr/lib/lpdfilters/ln01pp` which is the LN01 laser printer filter. This filter replaces the `pr` filter request that is made when using the `lpr` command with the `-p` option. See `lpr`(1) in the *ULTRIX Reference Pages*. For more information on using filter capabilities, see `lpd`(8) in the *ULTRIX Reference Pages*.

`ps`

Specifies the mode in which the daemon runs. Valid arguments are *non_PS* (non-postscript printers) and *LPS* (supported postscript printers).

`Ps`

Specifies the page size. Valid arguments are *a*, *letter*, *a3*, *a4*, *a5*, *b*, *ledger*, *executive*, and *legal*. If the specified page size is not available on the printer, an error occurs. This symbol is only valid with PostScript (TM) printers.

`pw`

Specifies the page width in characters. The default page width is 132 characters; however, you should not specify more than 80 characters for a letter quality printer that uses 8 1/2 by 11 paper.

`px`

Specifies the page width in pixels.

`py`

Specifies the page length in pixels.

`rf`

Specifies the filter for printing FORTRAN style text files.

`rm`

Specifies the machine name for a remote printer. This symbol may only be used in printcap entries for remote printers. In addition to this symbol, you must also define the printcap symbols `lp`, `rp`, and `sd` to access a printer on the remote system. A remote printer cannot process a print request if the hostname of the requesting node does not appear in the `/etc/hosts.lpd` or `/etc/hosts.equiv` files of the local and remote systems. Note that an asterisk (`*`) at the start of any line in the `/etc/hosts.lpd` enables remote print requests from all systems.

`rp`

Specifies the remote print name argument. The name specified must be one of the logical names for the printer on the remote system. This symbol must be specified for a remote printer.

`rs`

Restricts the remote printer usage to those users with local accounts.

`rw`

Specifies that the printer has read and write access. Usually, a printer allows only write access.

`sb`

Specifies a one-line banner.

`sc`

Suppresses multipile copies. This is equivalent to setting the `mc` symbol to one (`1`).

`sd`

Specifies the spooling directory where all print requests are queued before they are printed. The default is `/usr/spool/lpd`. Each printer connected to the system should have a unique name. Both local and remote printcap entries must specify a spooling directory. If you use a directory other than the default, you must create the directory. When the spooling directory is created, the intermediate directories are automatically created as needed.

`Sd`

Specifies the default sheet size value. Valid arguments are *`a`*, *`letter`*, *`a3`*, *`a4`*, *`a5`*, *`b`*, *`ledger`*, *`b4`*, *`b5`*, *`executive`*, and *`legal`*. Unlike the `Ss` symbol, if a sheet size is specified that is not available, an error does not occur and the print job is printed on the paper available with the printer. If the `Ss` symbol is specified, the `Sd` symbol is ignored. This symbol is only valid with PostScript (TM) printers.

`sf`

Suppresses all printer induced form feeds, except those present in the file. The `sf` symbol, when used with the `sh` symbol, is useful if you want to print a letter on a single sheet of stationary.

`sh`

Suppresses printing of the burst page header. See the **sf** symbol for more information.

Si

Specifies the default sides option. Valid arguments are *one_sided_duplex*, *2*, *two_sided_duplex*, *tumble*, *two_sided_tumble*, *one_sided_duplex*, *one_sided_tumble*, or *two_sided_simplex*. An error occurs if the specified argument is not available on the printer. For a description of these arguments, see the -K argument to lpr(1) in the *ULTRIX Reference Pages*. This symbol is only valid with PostScript (TM) printers.

Ss

Specifies the physical sheet size. Valid arguments are *a*, *letter*, *a3*, *a4*, *a5*, *b*, *ledger*, *b4*, *b5*, *executive*, and *legal*. An error occurs if the specified sheet size is not availabe on the printer. This symbol overrides the Sd symbol. This symbol is only valid with PostScript (TM) printers.

st

Specifies the status file name. The default name is status. The status file is located in the spooling directory. The status of the printer is written to this file.

tf

Specifies a troff data filter. For more information on using filter capabilities, see lpd(8) in the *ULTRIX Reference Pages*.

tr

Specifies a trailing string to print when the spooling queue empties. This symbol resets the printer to a known state. The trailing string may be a series of form feeds or escape sequence.

ts

Specifies a LAT terminal server node name.

uv

Specifies the ULTRIX version. This symbol is used by the daemon to determine whether to expand percent (%) escapes when using filter capabilities. Valid arguments are 3.0 and 4.0. When using the ct symbol, you must use the uv symbol. For more information on the symbols ct and uv, see lpd(8) in the *ULTRIX Reference Pages*.

Ul

Specifies the default upper page limit value. This must be a value from 1 through 10,000. This symbol is only valid with PostScript (TM) printers.

vf

Specifies a raster image filter. The available raster filter that can be specified with is /usr/lib/lpdfilters/ln01vf which is the LN01 laser printer filter. Other raster filters can be specified with the if or of symbols. For detailed descriptions of filters, see Section 8 of the *ULTRIX Reference Pages*. For more information on using filter capabilities, see lpd(8) in the *ULTRIX Reference Pages*.

xc

Specifies the local mode bits to clear when the terminal line is first opened. You should clear all bits by specifying xc=0177777 before specifying the xs symbol. See the description of the xs symbol for more information.

`xf`

Specifies the pass-thru filter name. This routine is specified when output is preformatted and does not require special filtering. For more information on using filter capabilities, see `lpd`(8) in the *ULTRIX Reference Pages.*

`Xf`

Specifies the translator dispatch program. The default is `xlator_call`. This symbol is only valid with PostScript (TM) printers.

`xs`

Specifies the local mode bits to set when the terminal line is first opened. You should clear all bits by specifying the `xc` symbol before specifying the `xs` symbol. For a detailed description of the status bits, see `tty`(4) in the *ULTRIX Reference Pages.* A brief description of the status bits follow:

Flag	Value	Description
LCRTBS	0000001	Backspace on erase rather than echoing erase
LPRTERA	0000002	Printing terminal erase mode
LCRTERA	0000004	Erase character echoes as backspace-space-backspace
LTILDE	0000010	Convert a tilde (~) to an apostrophe (`) on output (for Hazeltine terminals)
LLITOUT	0000040	Suppress output translations for 8–bit
LTOSTOP	0000100	Send SIGTOC for background output
LFLUSHO	0000200	Output is being flushed
LNOHANG	0000400	Do not send hangup when carrier drops
LAUTOFLOW	0001000	Hardware responds to flow control characters. (See Flow control.)
LCRTKIL	0002000	BS-space-BS erase entire line on line kill
LPASS8	0004000	Allow 8–bit characters in input and output
LCTLECH	0010000	Echo input control characters as Ctrl/X; delete as Ctrl/?
LPENDIN	0020000	Retype pending input at next read or input character
LDECCTQ	0040000	Only Ctrl/Q restores output after Ctrl/S, like DEC systems
LNOFLSH	0100000	Flush output on receipt of suspend character

D.2 Controlling Print Jobs

Once known to your system, the print system software requires little maintenance. This section describes the commands and files you use to maintain the print system. They are as follows:

`/usr/lib/lpd`	This program, the line printer daemon, is a print spool handler. Normally, the program is invoked at boot time from the `rc` file. The daemon works with several system programs and files to coordinate and synchronize printer activity. The ULTRIX operating system supplies this file. While you do not modify the file, you can specify spooling, logging, and locking activities. You must have superuser privileges to access this program.
`/etc/lpc`	This program lets you control the operation of the line printer system. While most control functions are available only to the superuser, some functions can be accessed by general users.
`/usr/ucb/lpr`	This program lets you queue and submit files for printing.
`/usr/ucb/lpq`	This program lets you examine the status of jobs currently on the print queue.
`/usr/ucb/lprm`	This program lets you remove jobs from the print queue.
`/etc/pac`	This program generates accounting information about printer use at your site. You must have superuser privileges to use this program.

The following sections describe how to use these files and commands. For more information on these commands, see the *ULTRIX Reference Pages.*

D.2.1 The Line Printer Daemon

The line printer daemon, `lpd`, provides network communications of print requests. It also provides the selection and start of specific print filters for specific print requests. The print filters process the varying input formats into printer-specific output format.

The line printer daemon interface is a task that runs automatically and remains running, ready for input. This daemon is generally started at boot time from the `/etc/rc` file. The `lpd` command invokes the line printer daemon. When you submit a print job using the `lpr` command, the line printer daemon schedules jobs and notifies printers that have jobs waiting.

When signaled for inputs, the line printer daemon checks the spooler directory, `/usr/spool/lpd`, for the existence of a lock file. If the lock file exists, `lpd` knows another job is currently printing. If a lock file is not present, `lpd` creates one to reserve access to the printer for a particular print job. Once the daemon creates the lock file, it scans the directory of files beginning with `cf`. These files are control files which represent print jobs. For example:

```
% lpr memo.1
% ls -l /usr/spool/lpd
total 3
-rw-rw----   1 daemon        86 July 9 11:11 cfA024myvax
-rw-rw----   1 dmf         2358 July 9 11:11 dfA024myvax
-rw-r--r--   1 root           5 July 9 11:11 lock
-rw-rw-r--   1 root          52 July 9 11:11 status
```

The control file beginning with `cf` contains print instructions and the data file beginning with `df` contains the formatted text. The lock file contains the process ID of the currently running daemon, while the status file contains a line describing the current printer status.

D.2.2 Controlling Printer Activity

The `lpc` command enables you to control the activity of the line printers and spooler queues listed in `/etc/printcap`. Use the `lpc` command to do the following:

- Enable/disable a printer
- Enable/disable a spooler queue
- Alter order of queued jobs
- Display printer, queue, or daemon status

You must have superuser privileges to enable or disable a printer or queue, or to alter the order of queued jobs.

D.2.3 Printing a File

The `lpr` command queues and submits a job for printing. For example, if you have a file named memo.1, you can print the file using:

```
% lpr memo.1
```

The `print` command paginates the job before printing. To do this type the following:

```
% print memo.1
```

If you pipe the file to `lpr`, the file name is listed as standard input. For example:

```
% cat memo.1 | lpr
```

If a file is not specified, the standard input is read.

D.2.4 Checking the Print Queue

The `lpq` command displays the current contents of the line printer queue and lists the jobs that have not yet printed. For example:

```
% lpq
lp is ready and printing
Rank      Owner     Job   Files                Total Size
active    dmf       24    memo.1               23056 bytes
1st       dmf       25    (standard input)      6987 bytes
```

There are two jobs in the print queue belonging to user `dmf`. The active job is number 24, memo.1. The `lpq` command displays information in the order in which it is scheduled to print.

D.2.5 Removing a Job from the Queue

The `lprm` command allows you to remove a job from a queue. To locate the job number and then remove a print job, type:

```
% lpq
lp is ready and printing
Rank   Owner   Job   Files              Total Size
active dmf     24    memo.1             23056 bytes
1st    dmf     25    /etc/printcap       6987 bytes
% lprm 24
dfA024myvax dequeued
cfA024myvax dequeued
```

When used without arguments `lprm` deletes the currently active job, if it is owned by you or if you are the superuser. If invoked with a user's name, it removes all print jobs owned by that user.

D.2.6 Generating a Report of Printer Use

The `pac` command displays a report detailing number of pages printed, feet of paper consumed, and total estimated cost per user. To periodically generate a report of your printer activity, type the following:

```
# /etc/pac
```

Note that the `pac` command can be used only if you have specified an accounting file for each printer for which a report is wanted.

Monitoring and Managing System Accounting E

This appendix provides guidelines that you can use to monitor and to manage system accounting.

E.1 Generating System Accounting Information

There are two types of system accounting information: accumulated and archived. During daily operations, system accounting information is accumulated so that you can keep track of day-to-day operations such as the following:

- User logins
- Command usage
- Printer usage

The following sections describe the commands you can use to display archived system statistics. Note that you must install the optional software subset for accounting to use many of these commands.

E.1.1 Generating User Log-In Report

The system automatically maintains two log-in accounting files: `/etc/utmp` and `/usr/adm/wtmp`. The system records all active logins in `/etc/utmp` and accumulates a user log-in history in `/usr/adm/wtmp`.

You can generate a report of the system's login-history with the `ac` command:

```
# /etc/ac -p
```

Over time, `/usr/adm/wtmp` increases in size. After you generate a hardcopy of the file you should clear it. To clear the `/usr/adm/wtmp` file, use the `cp` command with the arguments as follows:

```
# cp /dev/null /usr/adm/wtmp
```

This command copies `/dev/null` to `/usr/adm/wtmp`. That is, it reduces `/usr/adm/wtmp` to a zero-length file.

Note

The system automatically enables log-in history, but it accumulates a log-in history only if `/usr/adm/wtmp` exists. To disable the system log-in history, remove `/usr/adm/wtmp`.

For further information, see `cp`(1) and `ac`(8).

E.1.2 Generating Command Usage Report

During multiuser startup, /etc/rc normally enables system process accounting. When process accounting is enabled, the system records information on each executed process in /usr/adm/acct. In some systems, system process accounting may be disabled to save disk space.

You can display the contents of the system's current process accounting file, /usr/adm/acct, using the sa command. For example:

```
# /etc/sa
```

This report shows which commands are being used most often on the system.

The file /usr/adm/acct increases in size depending upon your system's activity. To manage space on your /usr file system, you should condense the process accounting information as necessary. To condense /usr/adm/acct, use the sa command with the -s option specified. For example:

```
# /etc/sa -s
```

This command merges the current information in /usr/adm/acct into /usr/adm/savacct, the process history file.

Note

To disable process accounting immediately, type:

```
# /etc/accton
```

To disable process accounting the next time the system reboots, comment out this line in the /etc/rc file by putting a # in the first column of the line on which the statement appears. This makes the accton line a comment which is not executed. For example:

```
# /etc/accton /usr/adm/acct; echo -n ' accounting' > /dev/console
```

For further information, see sa(8) in the *ULTRIX Reference Pages.*

E.1.3 Generating Printer Usage Report

Your system records all printer information in the default accounting file named in /etc/printcap if a default accounting file is specified in the /etc/printcap file.

To generate a report of your printer usage, use the pac command. For example:

```
# /etc/pac
```

The pac command displays a report detailing usage per user: number of pages printed, feet of paper consumed, and total estimated cost. For further information, see printcap(5) and pac(8) in the *ULTRIX Reference Pages.*

E.1.4 Generating Active System Report

In addition to those commands that are used to display accumulated system accounting information, the system has a number of commands that you can use to display active system statistics.

`iostat(1)`	Displays a report of current I/O statistics.
`ps(1)`	Displays a report of the system's process status.
`uptime(1)`	Displays a report of how long the system has been up.
`vmstat(1)`	Displays a report of virtual memory statistics.
`w(1)`	Displays a report of currently active users and what they are doing.
`pstat(8)`	Displays various system tables.
`netstat(1)`	Displays network activity.
`nfsstat(8nfs)`	Displays activity on the Network File System (NFS).

For further information on these commands, see the *ULTRIX Reference Pages.*

Index

U

Y

How to Order Additional Documentation

Technical Support

If you need help deciding which documentation best meets your needs, call 800-343-4040 before placing your electronic, telephone, or direct mail order.

Electronic Orders

To place an order at the Electronic Store, dial 800-234-1998 using a 1200- or 2400-baud modem from anywhere in the USA, Canada, or Puerto Rico. If you need assistance using the Electronic Store, call 800-DIGITAL (800-344-4825).

Telephone and Direct Mail Orders

Your Location	Call	Contact
Continental USA, Alaska, or Hawaii	800-DIGITAL	Digital Equipment Corporation P.O. Box CS2008 Nashua, New Hampshire 03061
Puerto Rico	809-754-7575	Local Digital Subsidiary
Canada	800-267-6215	Digital Equipment of Canada Attn: DECdirect Operations KAO2/2 P.O. Box 13000 100 Herzberg Road Kanata, Ontario, Canada K2K 2A6
International	————	Local Digital subsidiary or approved distributor
Internal*	————	SSB Order Processing - WMO/E15 *or* Software Supply Business Digital Equipment Corporation Westminster, Massachusetts 01473

* For internal orders, you must submit an Internal Software Order Form (EN-01740-07).

Reader's Comments

ULTRIX
Guide to System and Network Setup
AA-ME88C-TE

Please use this postage-paid form to comment on this manual. If you require a written reply to a software problem and are eligible to receive one under Software Performance Report (SPR) service, submit your comments on an SPR form.

Thank you for your assistance.

Please rate this manual:	Excellent	Good	Fair	Poor
Accuracy (software works as manual says)	☐	☐	☐	☐
Completeness (enough information)	☐	☐	☐	☐
Clarity (easy to understand)	☐	☐	☐	☐
Organization (structure of subject matter)	☐	☐	☐	☐
Figures (useful)	☐	☐	☐	☐
Examples (useful)	☐	☐	☐	☐
Index (ability to find topic)	☐	☐	☐	☐
Page layout (easy to find information)	☐	☐	☐	☐

What would you like to see more/less of? _____

_____

_____

What do you like best about this manual? _____

_____

_____

What do you like least about this manual? _____

_____

_____

Please list errors you have found in this manual:

Page	Description
_____	_____
_____	_____
_____	_____
_____	_____
_____	_____

Additional comments or suggestions to improve this manual:

_____

_____

_____

_____

What version of the software described by this manual are you using? _____

Name/Title _____ Dept. _____

Company _____ Date _____

Mailing Address _____

_____ Email _____ Phone _____

Do Not Tear – Fold Here and Tape

digital™

NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL
FIRST–CLASS MAIL PERMIT NO. 33 MAYNARD MA

POSTAGE WILL BE PAID BY ADDRESSEE

DIGITAL EQUIPMENT CORPORATION
OPEN SOFTWARE PUBLICATIONS MANAGER
ZKO3–3/Y32
110 SPIT BROOK ROAD
NASHUA NH 03062–2698

Do Not Tear – Fold Here

Cut
Along
Dotted
Line

Reader's Comments

ULTRIX
Guide to System and Network Setup
AA-ME88C-TE

Please use this postage-paid form to comment on this manual. If you require a written reply to a software problem and are eligible to receive one under Software Performance Report (SPR) service, submit your comments on an SPR form.

Thank you for your assistance.

Please rate this manual:	Excellent	Good	Fair	Poor
Accuracy (software works as manual says)	☐	☐	☐	☐
Completeness (enough information)	☐	☐	☐	☐
Clarity (easy to understand)	☐	☐	☐	☐
Organization (structure of subject matter)	☐	☐	☐	☐
Figures (useful)	☐	☐	☐	☐
Examples (useful)	☐	☐	☐	☐
Index (ability to find topic)	☐	☐	☐	☐
Page layout (easy to find information)	☐	☐	☐	☐

What would you like to see more/less of? _____

_____

_____

What do you like best about this manual? _____

_____

_____

What do you like least about this manual? _____

_____

_____

Please list errors you have found in this manual:

Page	Description
_____	_____
_____	_____
_____	_____
_____	_____
_____	_____

Additional comments or suggestions to improve this manual:

_____

_____

_____

_____

What version of the software described by this manual are you using? _____

Name/Title _____ Dept. _____

Company _____ Date _____

Mailing Address _____

_____ Email _____ Phone _____

Do Not Tear – Fold Here and Tape

digital™

NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST–CLASS MAIL PERMIT NO. 33 MAYNARD MA

POSTAGE WILL BE PAID BY ADDRESSEE

DIGITAL EQUIPMENT CORPORATION
OPEN SOFTWARE PUBLICATIONS MANAGER
ZKO3–3/Y32
110 SPIT BROOK ROAD
NASHUA NH 03062–2698

Do Not Tear – Fold Here

Cut
Along
Dotted
Line