HP-UX LAN Administrator's Guide

HP-UX 11i v2

Manufacturing Part Number : B2355-90796
E0903

United States

Legal Notices

The information in this document is subject to change without notice.

Hewlett-Packard makes no warranty of any kind with regard to this manual, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Hewlett-Packard shall not be held liable for errors contained herein or direct, indirect, special, incidental or consequential damages in connection with the furnishing, performance, or use of this material.

Warranty

A copy of the specific warranty terms applicable to your Hewlett- Packard product and replacement parts can be obtained from your local Sales and Service Office.

U.S. Government License

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

Copyright Notice

Trademark Notices

UNIX® is a registered trademark in the United States and other countries, licensed exclusively through The Open Group.

Intel® and Itanium® are registered trademarks of Intel Corporation.

Contents

Contents

Contents

About This Document

This document describes how to install, configure, and troubleshoot HP-UX 11i v2 LAN transport software.

The document printing date and part number indicate the document's current edition. The printing date will change when a new edition is printed. Minor changes may be made at reprint without changing the printing date. The document part number will change when extensive changes are made.

Document updates may be issued between editions to correct errors or document product changes. To ensure that you receive the updated or new editions, you should subscribe to the appropriate product support service. See your HP sales representative for details.

The latest version of this document can be found online at docs.hp.com, under the "Networking and Communications" topic, under "HP-UX LAN". Specifically, this is at: `docs.hp.com/hpux/netcom/index.html#HP-UX%20LAN`.

Intended Audience

This document is intended for system and network administrators responsible for installing, configuring, and managing HP-UX 11i v2 LAN transport software. Administrators are expected to have knowledge of Transmission Control Protocol/Internet Protocol (TCP/IP) networking concepts and network configuration. As well, it is helpful to have a working knowledge of operating system concepts, commands, and configuration.

This document is not a tutorial.

New and Changed Documentation in This Edition

This guide is for the HP-UX 11i v2 release only. It has been updated from *Installing and Administering LAN/9000 Software (Edition 10)*. Overall, this document has only had minor modifications to reflect changes specific to HP-UX 11i v2.

Publishing History

Table 1 Publishing History Details

Document Manufacturing Part Number	Operating Systems Supported	Publication Date
B2355-90796	11i v2	July 2003
B2355-90748	11i v1.6, 11i v1, 11.0, 10.x	March 2002

What's in This Document

This manual provides information for installing and administering the HP-UX LAN product. The HP-UX LAN product allows HP computers to connect to an IEEE 802.3 or Ethernet Local Area Network. An HP-UX LAN network can be further extended via bridges and routers into a Wide Area Network.

This manual also includes some information that may be useful for configuring other HP-UX link products, such as 10/100Base-T PCI. Specifically, this information includes:

- Manual configuration of links
- Network addressing
- LAN device and interface terminology

The information in this manual is intended for network managers or operators who install and administer HP-UX LAN on TCP/IP networks. It is assumed the reader is experienced with HP-UX and is familiar with the basics of local and wide area networking.

The manual is organized as follows:

Chapter 1 **Installing HP-UX LAN** describes how to install HP-UX LAN software.

Chapter 2 **Configuring HP-UX LAN Using SAM** describes the steps to configure HP-UX LAN software automatically using the System Administration Manager (SAM).

Chapter 3 **Manually Installing and Configuring HP-UX LAN** describes the steps to build the kernel, create device files, and manually configure HP-UX LAN using the `vi` editor.

Chapter 4 **Troubleshooting HP-UX LAN** provides flowcharts to help diagnose HP-UX LAN software and hardware problems. This chapter also describes LAN card status values and statistics returned by `lanadmin`.

Chapter 5 **LAN Resources** provides references to other useful tools for installing, configuring, and maintaining HP-UX LAN software.

Chapter 6 **Network Addressing** defines networking terms and explains network interface names, network addresses, names and subnets. The information in this chapter is applicable to configuring other HP-UX link products on TCP/IP networks.

Chapter 7 **LAN Device and Interface Terminology** defines terms used by the I/O system to identify LAN cards and device files.

IMPORTANT HP-UX 11i v2, by default, supports IPv6 transport. This guide does not include IPv6 transport information. Refer to *HP-UX IPv6 Transport Administrator's Guide* and *HP-UX IPv6 Porting Guide* for IPv6 transport information on HP-UX 11i v2.

NOTE This manual does not contain the procedures for adding and replacing PCI cards using OLA/R. OLA/R stands for On Line Addition and Replacement. This refers to the ability of a PCI I/O card to be replaced (removed and/or added) to an HP-UX computer system designed to support this feature without the need for completely shutting down, then rebooting the system or affecting other system components.

If you want to utilize the OLA/R feature that your system provides, refer to the *Interface Card OL* Support Guide*.

HP-UX Release Name and Release Identifier

Each HP-UX 11i release has an associated release name and release identifier. The *uname* (1) command with the `-r` option returns the release identifier. This table shows the releases available for HP-UX 11i.

Table 2 HP-UX 11i Releases

Release Identifier	Release Name	Supported Processor Architecture
B.11.23	HP-UX 11i v2	Intel® Itanium®

Table 2 HP-UX 11i Releases (Continued)

Release Identifier	Release Name	Supported Processor Architecture
B.11.22	HP-UX 11i v1.6	Intel® Itanium®
B.11.20	HP-UX 11i v1.5	Intel® Itanium®
B.11.11	HP-UX 11i v1	PA-RISC

Related Documents

HP Documentation

Additional information about HP-UX LAN can be found within *docs.hp.com* in the *networking and communications* collection under HP-UX LAN at:

`http://www.docs.hp.com/hpux/netcom/index.html#HP-UX%20LAN`

Additional information about HP-UX IPv6 transport software can be found within *docs.hp.com* in the *networking and communications* collection under IPv6 at:

`http://www.docs.hp.com/hpux/netcom/index.html#IPv6`

Related RFCs

This section provides a list of the protocols and standards on which the HP-UX LAN products are based. The IETF (Internet Engineering Task Force) RFCs listed below can be located at: `http://www.ietf.org/rfc.html`.

Table 3 LAN Protocols and Standards

For Information on:	Read:
Subnetting	RFC 950, RFC 1122
Classless Inter-Domain Routing	RFC1519
Requirements for IP Version 4 Routers	RFC 1812

Table 3 LAN Protocols and Standards (Continued)

For Information on:	Read:
Variable Length Subnet Table	RFC 1878
Protocols: Address Resolution Protocol (ARP)	RFC 826
Domain Requirements	RFC 920
Domain Name Server	RFC 1034, 1035, 1535
Internet Control Message Protocol (ICMP)	RFC 792
Internet Protocol (IP)	MIL-STD 1777; RFC 791
Standard for the Format of ARPA Internet Text Messages	RFC 822
Transmission Control Protocol (TCP)	MIL-STD 1788; RFC 793, 813, 814, 816, 817, 179, 889, 896, 1122, 1323, 2018, 2414, 2581, 2582
IP Multicast	RFC 1112
Path MTU Discovery	RFC 1191
Window Scaling	RFC 1323
Reverse Address Resolution Protocol (RARP)	RFC 903

HP Welcomes Your Comments

HP welcomes your comments concerning this document. We are truly committed to providing documentation that meets your needs.

Please send comments to: netinfo_feedback@cup.hp.com

Please include document title, manufacturing part number, and any comment, error found, or suggestion for improvement you have concerning this document. Also, please tell us what you like, so we can incorporate it into other documents.

New for the HP-UX 11i v2 Release

The HP-UX 11i v2 release has the following transport (IP, TCP, and UDP) changes:

- Transport support for IPv6

  IPv6 is the next generation Internet Protocol. IPv6 requires support from hosts and routers. HP-UX 11i v2 IPv6 transport software provides host support for IPv6.

  With only a few configuration steps, an IPv6 interface can be enabled, and you can use the functional features of IPv6 and IPv6-enhanced components for HP-UX 11i v2.

NOTE The following software currently offers IPv6 support: Transport, Internet Services, DCE, DLPI, FDDI, SAM-NNC, Libc, Commands, Desktop (CDE), X11R6-based applications, EMS, Online Diagnostics, SNMP, nettl, IPSec, Kerberos Client, Service Guard, Glance, HP-UX Secure Shell, Apache and JVM. Refer to product-specific documentation for more information.

  If you are not planning to use IPv6 software, there is no impact to you. Even though the software is included, other than a few new files and man pages, there are no actions required to “not use” IPv6.

  To learn more about IPv6 transport on HP-UX 11i v2, refer to the following documentation, all available at:http://docs.hp.com/hpux/netcom/index.html#IPv6

  — *HP-UX IPv6 Transport Administrator's Guide*

  — *HP-UX IPv6 Porting Guide*

  — “IPv6 Support” section of *HP-UX 11i Version 2 Release Notes*

NOTE This Guide only covers IPv4 related transport information. It does not cover IPv6 related transport information.

- Transport Support for HP-UX 11i v2 DLPI enhancements

  An OOP (Options negotiations and Out-of-Packet) header has been added to the front of all IP packets for both inbound packets from DLPI layer and outbound packets to DLPI layer. This change supports the HP-UX 11i v2 DLPI enhancements to the Streams Interface to provide a richer feature set for network stack feature options.

- Enhancements to CKO (Check-sum Offload) interfaces between transport and DLPI.

  This change makes the CKO interfaces more general with respect to the types of checksum offload hardware that HP will support, and adds sufficient flexibility to the interface that supports checksum offload for future transport protocols. This change incorporates the DLPI `OOP` header to carry the type of checksum offload and the offloaded checksum itself.

  The transport internal kernel header file, `net/cko.h` has been modified to support this enhancement. Kernel IP Stream modules that previously used options for fastpath negotiation features or looked into network data packets in previous versions must make changes and recompile for HP-UX 11i v2.

1 Installing HP-UX LAN

This chapter describes the manual procedures to load HP-UX LAN software and to install LAN hardware onto your system.

Overview

This chapter contains the following sections:

- Step 1: Checking LAN Installation Prerequisites
- Step 2: Loading LAN Software
- Step 3: Installing LAN Hardware

IMPORTANT If you have a system with HP-UX LAN pre-installed on it, you may skip this chapter and go directly to chapter 2. Execute the command `lanscan` to determine if the HP-UX LAN software and hardware have been pre-installed.

If, in addition, you have a preconfigured system, you may also skip chapter 2, "Configuring LAN Using SAM."

If you are unfamiliar with HP-UX LAN products or networking concepts, HP recommends that you read the *Networking Overview* manual, as well as Chapter 6, "Network Addressing," and Chapter 7, "LAN Device and Interface Terminology," of this guide before beginning HP-UX LAN installation.

Checking LAN Installation Prerequisites

Prior to loading the HP-UX LAN product onto your system, check that you have met the following hardware and software prerequisites:

- Check that `/usr/bin`, `/usr/sbin`, and `/sbin` are in your **PATH** using the command:

  ```
  echo $PATH
  ```

- The operating system should have been upgraded to HP-UX 11i v2 software. To obtain this information, execute the command:

  ```
  uname -a
  ```

- Prior to installing HP-UX LAN, HP recommends that you create a network map or update the existing map of your network. Refer to chapter 6 for a sample LAN network map and sample worksheet.
- You have the appropriate cabling for your LAN card.
- You have the installation and service manual for your network card.
- You have an IP address, subnet mask (optional), and host name alias for your new LAN card.
- You have superuser capability.

Loading LAN Software

Follow the steps below to load the HP-UX LAN software using the HP-UX `swinstall` program.

1. Log in as root.
2. Check that `/usr/bin`, `/usr/sbin`, and `/sbin` are in your PATH.
3. Insert the software media into the appropriate drive.
4. Run the `swinstall` program using the command:

   ```
   swinstall
   ```

   This opens the Software Selection Window and Specify Source Window.
5. Change the Source Host Name if necessary, enter the mount point of the drive in the Source Depot Path field, and activate the **OK** button to return to the Software Selection Window. Activate the Help button for more information.

   The Software Selection Window now contains a list of available software to install.
6. Highlight the Networking software.
7. Choose `Mark for Install` from the "Actions" menu to choose the product to be installed.
8. Choose `Install` from the "Actions" menu to begin product installation and open the Install Analysis Window.
9. Activate the *OK* button in the Install Analysis Window when the Status field displays a Ready message.
10. Activate the *Yes* button at the Confirmation Window to confirm that you want to install the software.

    View the Install Window to read processing data while the software is being installed, until the Status field indicates Ready and the Note Window opens.

    `swinstall` loads the fileset, runs the control scripts for the fileset, and builds the kernel.
11. Activate the *OK* button on the Note Window to reboot the system.

    The user interface disappears and the system reboots.
12. When the system reboots, check the `swinstall` log file in `/var/adm/sw` to make sure that the installation was successful.

For additional information on the HP-UX `swinstall` program, refer to *HP-UX 11i Version 2 Installation and Update Guide*.

Installing LAN Hardware

Follow the steps below to prepare the system for installation of your network interface hardware.

1. At the HP-UX prompt, execute the command:

   ```
   shutdown -h
   ```

   Wait for the system to respond with a message indicating that the system has been halted.

2. Observe antistatic precautions by following the guidelines as described in the installation instructions in the hardware manual or the Antistatic Precautions Note.

3. Install your network interface card by referring to your hardware installation and service manual.

CAUTION HP recommends not rearranging any network interface cards installed in your system. If you move any existing network interface card in the system, you may need to reconfigure the IP address. See “Reconfiguring IP Addresses” section in the next chapter.

4. Power up the system to complete the process. The network interface card will run a self-test automatically. Any error messages will appear on the terminal display or system console.

5. Proceed to chapter 2, “Configuring HP-UX LAN Using SAM.”

2 Configuring HP-UX LAN Using SAM

This chapter describes how to configure LAN using SAM, the System Administration Manager. It contains the following sections:

- Overview of Configuration Using SAM

- Step 1: Configuring the Network Interface Card
- Step 2: Configuring Network Connectivity
- Step 3: Verifying the Installation
- Step 4: Reconfiguring IP Addresses

Overview of Configuration Using SAM

Once you have installed hardware and software, you can use SAM to automatically configure networking.

SAM stands for System Administration Manager, a menu-driven utility for system administration tasks, including configuration of networking software. SAM has two user interfaces, an X-Windows system interface and a text terminal interface. The primary components and functionality of SAM are the same for both interfaces. The differences are the screen appearance and the navigation methods.

You can access the SAM on-line help system using the following methods:

- Choose an item from the “Help” menu (located in the menu bar) for information about the current SAM screen, keyboard navigation within SAM, and the version of SAM on your system.
- Activate the HELP button from a dialog or message box for information about the attributes and tasks you can perform from the currently displayed window.
- Press the F1 key for context-sensitive information for the object at the location of the cursor.

Using SAM to configure HP-UX LAN can be divided into two procedures:

1. Configuring the Network Interface Card.
2. Configuring Network Connectivity.

Follow Step 1 to add the IP address, any alias names, and, if the LAN card is on a subnetwork, enter the subnet mask for your card. This procedure will automatically initialize the LAN link and attach your node to the local area network (LAN). Follow Step 2 to add remote system names and remote system IP addresses for network connectivity, and also to specify default gateway information.

Configuring the Network Interface Card

NOTE Make sure the LAN card and driver are installed in the system before you use SAM to configure the software.

Log in as `root` and do the following:

1. At the HP-UX prompt, type: `sam`
2. Select the `Networking and Communications` item of the SAM main window.
3. Select the `Network Interface Cards` item of the `Networking and Communications` window.
4. Double-click the LAN card that you want to configure from the object list.
5. Verify that the hardware path is correct for your LAN card (by comparing the hardware path on the Configure LAN Card window with the hardware path from the output of the commands `ioscan -fdbtlan` and `lanscan -v`).
6. If you are configuring the first logical interface for a card type (also called the "initial interface"), highlight the card and choose `Configure` from the "Actions" menu to open the `Configure LAN Card` window.

   If you are adding logical interfaces to a card type, choose `Add IP Logical Interface` from the "Actions" menu to open the `Configure LAN Card` window.

   a. Enter the information about the LAN card. To do so, press the *Tab* key to move through the data entry fields.

   NOTE SAM displays the Card Name, Hardware (H/W) Path, and Station Address fields with the appropriate values. These fields cannot be modified.

   b. Choose the Card Type of your LAN card. The default is Ethernet. SNAP802.3 can be selected.

   NOTE The Enable DHCP button specifies that the system is a Dynamic Host Configuration Protocol (DHCP) client. If you activate this button, the IP parameters for this system will be set using DHCP.

c. Enter the Internet address for your LAN card.

   Upon exiting the Internet Address field, SAM checks to make sure that the IP address you entered is correctly formatted and is not currently in use.

d. Specify whether your LAN card will be on a subnetwork or supernet.

   The Subnet Mask field can contain a fixed-length subnet mask, a variable-length subnet mask, or a supernet mask. Enter the appropriate type of mask for your network addressing scheme.

   Refer to the "Network Addressing" chapter for more information regarding network addressing schemes.

e. Optionally, enter comments about your LAN card.

f. Choose `Add Host Name Aliases`. You must complete this step if you have more than one LAN card installed in your system.

g. Add, modify, or remove alias host names for your logical interface.

h. Activate the *OK* button to perform the task and return to the `Configure LAN Card` window.

i. Optionally activate the *Advanced Options* button. This will allow you to modify the Station Address and configure an Internet Broadcast Address. The Maximum Transmission Unit (MTU) field is display only.

7. Activate the *OK* button at the `Configure LAN Card` window to enable your LAN card.

   If the software is correctly configured, SAM displays the Network Interface Cards object list with the status Enabled for your LAN card; otherwise, SAM displays an error message.

8. Choose `Exit` from the "File" menu.

9. At the `Networking and Communications` window, choose `Exit SAM` from the "File" menu.

NOTE If you have moved or removed any LAN cards from the system, HP recommends that you verify the IP address of every card in the backplane before leaving SAM.

---

10. IP accesses LAN devices through a single Data Link Provider Interface (DLPI) device file `/dev/dlpi`. The `/dev/dlpi` file is created automatically during installation. Verify that the `/dev/dlpi` file has been created correctly by executing the HP-UX command:

```
ls -al /dev/dlpi
```

    The output appears something like the following:

```
crw-rw-rw-  1  bin     bin    72 0x000077 Dec 4 11:14 /dev/dlpi
```

The major number, shown in the fifth column, must be 72 (decimal) and the minor number, shown in the sixth column, must be 77 (hexadecimal). If the major numbers or minor numbers are not correct, delete the device file and recreate it with the correct numbers using the mknod command. For example:

```
mknod /dev/dlpi c 72 0x77
```

Continue to the next section, “Configuring Network Connectivity” if you want to configure your system for network connectivity. Otherwise, continue to “Verifying the Installation”.

Configuring Network Connectivity

Your system may not be able to communicate with other systems (for example, PCs, workstations, servers, etc.) until you configure system-to-system connections. You can use SAM to do this automatically by completing the following steps:

1. At the HP-UX prompt, type: `sam`
2. Double-click on the `Networking and Communications` item of the SAM main window.
3. Double-click on `Hosts`. Double-click on the `Local Hosts File` item.

   SAM displays the remote system names and IP addresses that are already configured.
4. Choose `Add` from the "Actions" menu to open the `Add Host to File` window.

   Use the SAM on-line help system for information about adding remote host connections.

   a. Enter the Internet Address for the remote system.

      Upon exiting the Internet Address field, SAM checks to make sure you have entered a valid IP address. SAM also determines if a gateway is required for the connection (see step 4f).

   b. Enter the remote host name.

      Upon exiting the `Remote Host Name` field, SAM checks to make sure that connectivity has not already been configured for this system. If it has, SAM displays an error message.

   c. Optionally, choose `Configure Aliases` to open the `Configure Aliases` window for remote systems.

   d. Add, modify, or remove alias names for the remote system.

   e. Activate the *OK* button to perform the task and return to the `Add Host to File` window.

   f. Proceed to step 5 if a gateway is not required for this remote connection.

      SAM displays fields for entering gateway information if a gateway is required for this remote system connection. Use the SAM on-line help system for information about gateways.
5. Activate the *OK* button to enable your system to communicate with this system and return to the `Internet Addresses` window.

   SAM updates the object list to include the remote system you configured.

NOTE You can modify or remove remote systems and modify default gateways by highlighting the `Remote System Name` from the object list and choosing Modify, Remove, or Modify Default Gateway from the "Actions" menu.

6. Choose `Exit` from the "File" menu.
7. Then Exit SAM from the "File" menu again.
8. Verify remote system configuration.
   a. View the list of remote systems you can communicate with using a symbolic name by typing the following command at the HP-UX prompt (note that this file may be large):

      ```
      more /etc/hosts
      ```

   b. View the configured destinations reached through gateways and the gateways used to reach those destinations by typing the following command at the HP-UX prompt:

      ```
      netstat -r
      ```

To verify that you can communicate with a remote system using the HP-UX LAN product, continue to "Verifying the Installation".

Verifying the Installation

Once your HP-UX LAN software is installed, fully configured and running, you should execute the following commands to verify LAN hardware and software installation. See the man pages for complete descriptions of the commands listed below.

1. Check the state of all LAN hardware. To do so execute the `lanscan` command and verify that the Hardware State is `UP`.
2. Check the state of the network interface. To do so execute `ifconfig` on the interface you wish to verify. The example below will provide information about the `lan0` interface.

   ```
   ifconfig lan0
   ```

3. Verify link level loopback connectivity using the `linkloop` command with the PPA and station address of the interface you want to test. You can obtain the station address (typically 0x080009######) and PPA from the `lanscan` command output. The example below will test the connectivity of the LAN card with PPA 4 and station address 0x080009266C3F.

   ```
   linkloop -i 4 0x080009266C3F
   ```

4. Verify that the `/dev/lan` and `/dev/snap` files have been created and symbolically linked to the `/dev/dlpi` file.

   ```
   ls -l /dev/lan
   ls -l /dev/snap
   ```

5. To check that the your system can communicate with other systems, enter the `ping` command at the HP-UX prompt. In this example, 191.2.1.2 is the IP address of the remote system. Enter `[Ctrl]-C` to stop `ping`.

   ```
   ping 191.2.1.2
   ```

HP-UX LAN installation is verified if the steps above are successful. For information on troubleshooting HP-UX LAN configuration and operation, refer to the section“Diagnostic Flowcharts” in Chapter 4, “Troubleshooting HP-UX LAN.”

Reconfiguring IP Addresses

If you have rearranged any network interface cards in the system you may need to reconfigure the IP addresses. Follow the steps below:

1. At the HP-UX prompt, type: `sam`
2. At the main menu select the `Networking and Communications` item.
3. Select the `Network Interface Cards` item.
4. Verify the IP addresses of all the adapters in the system by reviewing the Card Name, Hardware Path, and Internet Address displayed in the Network Interface Cards window.
5. For adapters with incorrect IP addresses, follow the steps below:
   a. Select the adapter you wish to modify.
   b. Select `Modify` from the “Actions” menu.
   c. Modify the Internet Address and select `ok`.

3 Manually Installing and Configuring HP-UX LAN

This chapter provides information on manually configuring HP-UX LAN software. Manually configuring HP-UX LAN involves the following steps:

- Creating a New Kernel
- Verifying LAN Device Files
- Editing Driver Configuration Files
- Editing LAN Configuration Files
- Creating the `/etc/hosts` File
- Activating Optional Network Features

Creating a New Kernel

Before attempting this procedure, familiarize yourself with the system reconfiguration information in the `mk_kernel(1M)` man page and HP-UX system literature.

Refer to the System Administration manual for your system for complete instructions on how to create a kernel. The steps below contain a general outline of kernel configuration steps with some information for the HP-UX LAN drivers and parameter settings contained within that framework.

If the kernel was not created with the LAN driver in it during the `swinstall` procedure, you can create it manually. To determine whether the LAN driver is in the kernel, check the `/stand/system` file for the `lanX` keyword (where X=2 or 3). To determine whether the LAN driver is installed, execute `what /stand/vmunix` and check for references to LAN in the output.

If you used some other file to create the kernel previously, copy that file to `/stand/system` before following the steps below.

1. Ensure that you have superuser capabilities.
2. Change to the `/stand` directory.
3. Make a backup copy of your current configuration description file (which is commonly `system` or `build/system`).
4. Edit the `system` file to add drivers and/or change system parameters. (Refer to your specific Ethernet LAN documentation, as appropriate for your configuration, for complete information.)

   a. Verify or add the appropriate keywords:

      Some examples of keywords are:

      - btlan (PCI 10/100BT adapters)
      - gelan (Gigabit Ethernet add-on adapters)
      - igelan (Next Generation Core IO 10/100/1000 cards)

      If the keyword does not exist in the `/stand/system` file, add the appropriate lines to the file.

b. Depending on which filesets you have loaded, you may also need to add the following keywords to the `system` file:

Table 3-1 Keywords

Keyword	Comments
hpstreams	required for streams
dlpi	required
uipc	required for TCP/IP
inet	required for TCP/IP
nms	required for TCP/IP
nfs_core	
nfs_client	
nfs_server	
netdiag1	
tun	required for PPP

5. Make a copy of the existing kernel (default name `vmunix`).
6. Regenerate the kernel with `mk_kernel`, using the edited `system` file as input. `mk_kernel` creates the new hp-ux kernel (the default is `/stand/build/vmunix_test`).

   ```
   mk_kernel
   ```

   In this example a new kernel is created in the build directory called `vmunix_test`.

   ```
   mk_kernel -s /stand/system -o /stand/vmunix
   ```

   This second example automatically moves the kernel to the `/stand` directory and makes a backup (`/stand/vmunix_prev`) if the file `/stand/vmunix` already exists.
7. If you did not use the `-o` option with the `mk_kernel` command, execute `kmupdate` and this will copy the new kernel to `/stand/vmunix`. (Or you can manually copy the new kernel to `/stand/vmunix`.)
8. Reboot the new kernel. If the new kernel fails to boot, boot the system from the backup kernel and repeat the process of creating a new kernel. To do so, follow the instructions in the *Managing System and Workgroups: A Guide for HP-UX System Administrators* manual.

9. Proceed to the next section “Verifying LAN Device Files,” in this chapter to continue configuring HP-UX LAN manually. To configure your system with the SAM utility, proceed to Chapter 2, “Configuring HP-UX LAN Using SAM.”

Verifying LAN Device Files

All IP access to LAN devices is done through the files /dev/lan and /dev/snap, which are symbolically linked to the Data Link Provider Interface (DLPI) device file /dev/dlpi. The /dev/dlpi file is automatically created at installation time. Verify that the /dev/lan and /dev/snap files have been created and symbolically linked to /dev/dlpi by entering:

ls -l /dev/snap output will look something like:

```
lrwxr-xr-x 1 root sys  9 Mar 15 11:59 /dev/snap -> /dev/dlpi
```

ls -l /dev/lan output will look something like:

```
crw-rw-rw- 1 root sys 72 0x000077 Mar 15 11:59 /dev/lan
```

ls -l /dev/dlpi output will look something like:

```
crw-rw-rw- 1 root sys 72 0x000077 Mar 15 11:59 /dev/dlpi
```

The major number, shown in the fifth column, must be 72 (decimal) and the minor number, shown in the sixth column, must be 77 (hexadecimal). If the information differs significantly, delete the file and recreate it correctly with the mknod command:

```
mknod /dev/dlpi c 72 0x77
```

or

```
insf -d dlpi
```

Editing Driver Configuration Files

If needed, edit appropriate configuration files. Refer to your specific Ethernet LAN documentation, as appropriate for your configuration, for complete information.

Below are some examples of software driver names, configuration files, and init scripts for Fast Ethernet and Gigabit Ethernet cards/adapters.

Table 3-2 Fast Ethernet Examples

Fast Ethernet	driver name	configuration file	init script
Core PCI 10/100BT cards	btlan	hpbtlanconf	hpbtlan

Table 3-3 Gigabit Ethernet Examples

Gigabit Ethernet	driver name	configuration file	init script
Add-On 10/100/1000 Adapters (A4926A, A4929A)	gelan	hpgelanconf	hpgelan
Core 10/100/1000 Cards	igelan	hpigelanconf	hpigelan
Add-On 10/100/1000 Cards (A6847A, A6825A)	igelan	hpigelanconf	hpigelan

Editing LAN Configuration Files

To configure and initialize LAN manually, you must edit the `/etc/rc.config.d/netconf` file and execute the HP-UX LAN initialization script. To do so, you must be logged on as `root`. The script reads the information in the edited `netconf` file when the system reboots and performs the following:

- Configures the network interface with an IP address and optional subnet mask.
- Configures the network routing table if your node is a gateway or on a LAN with a gateway.

NOTE You must initialize HP-UX LAN to use HP-UX NFS or HP-UX Internet Services. Refer to "Executing the Network Configuration Script" section below for options available to make configuration changes active.

You may also want to modify the default station address. To modify the station address, you must edit the (driver specific) configuration file in the directory `/etc/rc.config.d`. Refer to Table 3-2 and Table 3-3 for some examples, and to your specific Ethernet LAN documentation, as appropriate for your configuration, for complete information.

Modifying the Hewlett-Packard Station Address (optional)

This step is optional and most customers do not need to modify the station address of their Hewlett-Packard LAN cards.

Modifying the station address of an active card will probably destroy existing connections and traffic resulting in data loss. Modifying the station address can also cause temporary confusion in the network because all nodes that were communicating with the local node will detect an error when using the existing station address. The confusion is temporary and all nodes may eventually recover from the situation. Problems caused by modifying a station address during a communication session are difficult to troubleshoot.

Editing the Driver Configuration File in /etc/rc.config.d/

Editing the driver configuration file (refer to Table 3-2 and Table 3-3 for some examples, and to your specific Ethernet LAN documentation, as appropriate for your configuration, for complete information) will modify the station address each time the system reboots. The modified station address is not permanent. Each system reboot causes the preset station address to be overwritten when a value is supplied to the configuration parameters.

CAUTION Each HP LAN card has a preset station address, for example 080009xxxxxx. This address must be modified with caution. Modifying the station address of an active card will probably destroy existing connections and traffic resulting in data loss.

The station address configuration parameters have an index value, [x], that groups the station address parameters together. The index value must be different for each additional interface.

Following is a sample `hpbtlanconf` entry:

```
HP_BTLAN_INTERFACE_NAME[0]="lan1"
HP_BTLAN_STATION_ADDRESS[0]="0x022345678901"
```

Editing /etc/rc.config.d/netconf

Editing the `netconf` file allows you to identify the network interface name, IP address, and subnet mask of your LAN card, and add entries to the network routing table. As the `netconf` file has read-only permission, you must have superuser capability to make modifications to this file.

The steps to add the hostname, loopback address, internet configuration information and routing configuration information to the netconf file are listed below.

NOTE The `netconf` file and the script that is executed are shell programs; therefore, shell programming rules apply.

1. Verify that the hostname and loopback address are set. Upper layer software often requires loopback. Be sure that loopback is also enabled in the `netconf` file.

   ```
   HOSTNAME="nameofyoursystem"
   LOOPBACK_ADDRESS="127.0.0.1"
   ```

2. Add internet configuration information.

   In this step you will assign an IP address and subnet mask, and configure network interface parameters. The internet configuration parameters have an index value, [x], that groups the configuration parameters together. Following is a sample of the internet configuration information of the `netconf` file entry:

   ```
   INTERFACE_NAME[0]="lan0"
   IP_ADDRESS[0]="192.6.1.1"
   SUBNET_MASK[0]="255.255.224.0"
   ```

```
BROADCAST_ADDRESS[0]=""
INTERFACE_STATE[0]=""
DHCP_ENABLE[0]="0"
```

If you have more than one interface to configure, you must have a complete set of internet configuration parameters for each interface. The index value must be different for each additional interface. The network interface name in `lanscan` corresponds to the `INTERFACE_NAME[`*x*`]` in the internet configuration parameters.

You can configure multiple logical interfaces for a physical interface. You must configure the initial interface for a card/encapsulation type before you can configure other logical instances of the same card/encapsulation type. For example, you must configure lan0:0 (or lan0) before you configure lan0:1 and lan0:2. In the `netconf` entry shown previously, lan0 is equivalent to lan0:0, the initial interface for lan0. The following is a sample `netconf` entry for a second logical interface on lan0:

```
INTERFACE_NAME[1]="lan0:1"
IP_ADDRESS[1]="192.6.3.3"
SUBNET_MASK[1]="255.255.224.0"
BROADCAST_ADDRESS[1]=""
INTERFACE_STATE[1]=""
DHCP_ENABLE[1]="0"
```

For more information about specifying interface names for multiple logical interfaces, see chapter 6, "Network Addressing."

3. Add routing configuration information.

   If you intend to use your system as a gateway or to communicate with gateways, add the route destination, gateway address, and hop count parameter information. The routing configuration parameters have an index value, [x], that groups the routing parameters together. Following is a sample `netconf` entry:

   ```
   ROUTE_DESTINATION[0]="default"
   ROUTE_GATEWAY[0]="192.6.1.2"
   ROUTE_COUNT[0]="1"
   ROUTE_MASK[0]=""
   ROUTE_ARGS[0]=""
   ```

   The index value must be different for each additional route.

Executing the Network Configuration Script

Once you have edited the `netconf` file, you need to activate this configuration. After adding the LAN and routing configuration information into the `netconf` file, you can either reboot your system, execute the `ifconfig`, and `route` commands manually, or re-execute the `/sbin/init.d/net.init` and `/sbin/init.d/net` scripts. This section discusses each of these options.

Option 1: Reboot your system. HP recommends that you reboot your system to activate any changes you made in your `netconf` file. A reboot is the cleanest method for executing the network script because the reboot handles any other network initialization dependencies.

Option 2: Execute the `ifconfig` and `route` commands at the HP-UX prompt. HP recognizes that system reboots are disruptive to end users. To delay or schedule the reboot, but still make your configuration changes active, you may execute the `ifconfig` and `route` commands with the appropriate values for your network. When you reboot, the values in your `netconf` file will be used. Refer to the `lanscan(1M)`, `ifconfig(1M)`, and `route(1M)` man pages for information on command usage.

Option 3:

If you made changes to the station address, execute the init script in the directory /sbin/init.d/. Refer to Table 3-2 and Table 3-3 for some examples, and to your specific Ethernet LAN documentation, as appropriate for your configuration, for correct init script. For a 100BT PCI card on an rx2600 system, here's an example:

```
/sbin/init.d/btlan start
```

After executing the above command, execute the command:

```
/sbin/init.d/net start
```

These commands will source the contents of the `netconf` and `hpbtlanconf` files. Executing these scripts will not necessarily properly re-initialize any other networking subsystems. For example, if you are running an Internet Service over the link you have just configured manually, the service may not work with your new configuration. The system boot sequence initializes networking subsystems and products in the correct order. Initializing a specific subsystem alone may cause network problems.

Creating the /etc/hosts File

You must edit the `/etc/hosts` file to add an IP address and hostname for the LAN card that you are installing.

NOTE If you are using a naming service (DNS or NIS), you will need to modify the `/etc/hosts` file to add the IP address and host name to the appropriate databases on the name server system. Refer to *Internet Services Administrator's Guide* and *NFS Services Administrator's Guide* for more information on naming services.

The `/etc/hosts` file associates IP host addresses with mnemonic host names and alias names. It contains the names of other nodes in the network with which your system can communicate. HP-UX LAN diagnostics `netstat` and `ping` use `/etc/hosts`. If you install HP-UX Internet Services or HP-UX NFS, those products also use the `/etc/hosts` file.

You can create an `/etc/hosts` file three ways:

- From scratch, entering the known nodes in the format shown below.
- By copying the file from another node.
- By copying the official host database maintained at the Network Information Control Center (NIC) for ARPA Internet networks, if you are installing HP-UX Internet Services.

If you copy an `/etc/hosts` file from another host, you may need to update the file by adding unofficial aliases or unknown hosts, including your own host.

Network and System Names

A system is known by several names, each with its own purpose:

System name: Used for cluster configuration and UUCP communication.

Host name and aliases: Used for most network communication. This might be a fully qualified domain name including the DNS domain. For example: `turtle.bnio.nmt.edu`

HP recommends that you try to keep these names as consistent as possible, within their limitations. This will help to minimize confusion.

The examples below show how a system with the name, `host3`, might be referenced in the `/etc/hosts` and other system and networking files and commands:

System name in Install screen:

```
                host3
```

`/etc/rc.config.d/netconf` file:

```
                HOSTNAME=host3
```

`/etc/hosts` file:

```
                192.6.1.1 host3 host3.site2.region4
```

```
uname -S host3
hostname host3
```

NOTE When you first install a system, the `netconf` HOSTNAME entry, the `/etc/hosts` entry, the `hostname`, and `uname -S` are set for you automatically.

/etc/hosts

Each node has a one line entry in the `/etc/hosts` file. Each entry in the file takes the following form:

Syntax

`IP_address host_name` [*`alias`*]...

Parameters

`IP_address`	The IP address that uniquely identifies the node. *`IP_address`* must be in internet "dot" notation. Refer to Chapter 6, Network Addressing, for more information on IP addresses.
`host_name`	Name of the node. Host names can contain any printable character except spaces, newline, or the comment character (#). Naming Convention: the first nine characters should be unique for each network host.
`alias`	Common name or names for the node. An alias is a substitute for *`host_name`*. Alias names are optional. Naming Convention: the first nine characters should be unique for each network host.

/etc/hosts Format

When creating the `/etc/hosts` file, follow these rules:

- Lines cannot start with a blank or tab character.
- Fields can have any number of blanks or tab characters separating them.

- Comments are allowed and designated by a pound sign (#) preceding the comment text.
- Trailing blank and tab characters are allowed.
- Blank line entries are allowed.
- Only one host entry per line is allowed.

/etc/hosts Permissions

HP recommends that the `/etc/hosts` file be owned by user `root` and have `0x444` (`-r--r--r--`) access permission. For more information on `/etc/hosts`, refer to the `hosts(4)` man page in the HP-UX Reference Pages.

NOTE HP highly recommends that you limit access to the `/etc/hosts` file by setting the permission to `0x444` (`-r-—r-—r--`) for read access only.

/etc/hosts Example

The following `/etc/hosts` entry contains an IP address, hostname, and alias names (host3.site2.region4 and grace).

```
192.6.1.1      host3      host3.site2.region4     grace
```

Activating Optional Network Features

To activate special network features, you may also want to configure /etc/services, /etc/networks, and /etc/protocols. Each of these steps is optional.

NOTE If you are using NIS, modifications to the /etc/services, /etc/networks and /etc/protocols files should only be made on the NIS Master Server. Refer to the *NFS Services Administrator's Guide* for more information

Modifying the /etc/services File

The /etc/services file associates port numbers with mnemonic service names and alias names. The /etc/services file contains the names, protocol names, and port numbers of all services known to your local host. The netstat diagnostic uses the /etc/services file.

If you install HP-UX Internet Services or HP-UX NFS, these products will also use the /etc/services file.

NOTE You can modify this file if you have special requirements, but it is properly configured when you receive HP-UX LAN.

/etc/services

Each service has a one line entry in the /etc/services file. Each entry in /etc/services file takes the following form:

Syntax

service_name port_num/protocol [*alias*]...

Parameters

service_name Name of the service. Service names can contain any printable character except spaces, newline, or the comment character (#).

port_num / *protocol* *port_num* is the protocol port number assigned to this service. All requests for this service must use this port number. *protocol* is the protocol name, as listed in /etc/protocols, that the service uses.

alias Common name or names for the service. An alias is a substitute for *service_name*. Alias names are optional.

/etc/services Format

- Lines cannot start with a blank or tab character.
- Fields can have any number of blanks or tab characters separating them.
- Comments are allowed and designated by a pound sign (#) character preceding the comment text.
- Trailing blank and tab characters are allowed.
- Blank line entries are allowed.
- Only one entry per line is allowed.

/etc/services Permissions

The `/etc/services` file should be owned by user `bin`, group `bin`, and it should have `0x444` (`-r--r--r--`) access permission.

Refer to the `/etc/services` file for examples of actual format and contents. For more information on `/etc/services`, refer to the `services(4)` man page in the HP-UX Reference Pages.

/etc/services Example

The following `/etc/services` entry contains the service name, port number, protocol name, and alias name for the `shell` service.

```
shell 514/tcp cmd #remote command, no passwd used
```

Creating the /etc/networks File

The `/etc/networks` file associates network addresses with mnemonic names and alias names. The `/etc/networks` file contains the name and address of known internet networks with which your host can communicate. The HP-UX LAN diagnostic `netstat` and the `route` command use the `/etc/networks` file. You must configure this file for your host if you want `route` or `netstat` to use symbolic network names instead of addresses.

You can create an `/etc/networks` file two ways:

- From scratch, entering the known nodes in the format shown below.
- By copying the file from another node.

If you copy an /etc/networks file from another host, you may need to update the file by adding unofficial aliases or unknown networks, including your own network.

/etc/networks

Each network has a one line entry in the /etc/networks file. Each entry in /etc/networks file takes the following form:

Syntax

network_name network_address [*alias*]...

Parameters

network_name Name of the internet network. Network names can contain any printable character except spaces, newline, or the comment character (#).

network_address Network address that uniquely identifies the network. This address can be a subnet or supernet address. It may also contain the netmask translation. *network_address* must be in dot notation. See Chapter 6 for details on network addresses.

alias Common name or names for the network. An alias is a substitute for *network_name*. Alias names are optional.

/etc/networks Format

- Lines cannot start with a blank or tab character.
- Fields can have any number of blanks or tab characters separating them.
- Comments are allowed and designated by a pound sign (#) character preceding the comment text.
- Trailing blank and tab characters are allowed.
- Blank line entries are allowed.
- Only one entry per line is allowed.

/etc/networks Permissions

The /etc/networks file should be owned by user bin, group bin, and it should have 0x444 (-r--r--r--)access permission.

For more information on /etc/networks, refer to the networks(4) man page in the HP-UX Reference Pages.

/etc/networks Example

The following /etc/networks entry contains a network name, network address, and an alias name for the network neta.

```
neta 192.6.1 testlan
```

Modifying the /etc/protocols File

The /etc/protocols file associates port numbers with mnemonic names and alias names. The /etc/protocols file contains the names and protocol numbers of all protocols known to your local host. The netstat diagnostic uses the /etc/protocols file. If you install HP-UX Internet Services or HP-UX NFS, those products will also use the /etc/protocols file.

NOTE You can modify this file if you have special requirements, but it is properly configured when you receive the HP-UX LAN software.

/etc/protocols

Each protocol has a one line entry in the /etc/protocols file. Each entry in /etc/protocols file takes the following form:

Syntax

protocol_name protocol_num [*alias*]...

Parameters

protocol_name	Name of the protocol. Protocol names can contain any printable character except spaces, newline, or the comment character (#).
protocol_num	Protocol number that identifies this protocol.
alias	Common name or names for the protocol. An alias is a substitute for *protocol_name*. Alias names are optional.

/etc/protocols Format

- Lines cannot start with a blank or tab character.
- Fields can have any number of blanks or tab characters separating them.
- Comments are allowed and designated by a pound sign (#) character preceding the comment text.
- Trailing blank and tab characters are allowed.

- Blank line entries are allowed.
- Only one entry per line is allowed.

/etc/protocols Permissions

The `/etc/protocols` file should be owned by user `bin`, group `bin`, and it should have `0x444` (`-r--r--r--`) access permission.

Refer to the `/etc/protocols` file for examples of actual format and contents. For more information on `/etc/protocols`, refer to the `protocols(4)` man page in the HP-UX Reference Pages.

/etc/protocols Example

The following `/etc/protocols` entry contains a protocol name, protocol number and an alias name.

```
tcp 6 TCP
```

4 Troubleshooting HP-UX LAN

This chapter provides guidelines for troubleshooting H-UX LAN. It contains the following sections:

- Troubleshooting Overview

- Troubleshooting Q and A
- LAN Interface Card Statistics
- 100Base-T Checklist
- Diagnostic Flowcharts

Troubleshooting Overview

Troubleshooting LAN problems can be difficult because a variety of hardware and software components may be involved and because the problem impacting your system may originate in another part of the network.

The “Troubleshooting Q and A” section in this chapter provide quick fix solutions to common network problems by providing answers to the most frequently asked troubleshooting questions. Look through the problems identified to see if your problem fits into any of these categories. If so, you may be able to quickly identify and recover from the problem without any further investigation.

If, however, you are unable to identify your problem, proceed to the “100Base-T Checklist” to have a general view of your system and its possible problems. Then if you are still unable to identify your problem, proceed to the troubleshooting flowcharts. The troubleshooting flowcharts provide a logical sequence of steps to follow when troubleshooting HP-UX LAN. Using the diagnostic flowcharts provided in this chapter, identify whether the problem is with HP-UX LAN, any of the connectors, or if it is in some other part of the LAN network. Verify your assumptions and, it if is limited to HP-UX LAN software and hardware, correct the problem.

Troubleshooting Q & A

New system disrupts LAN: I tried to attach a new system to our site LAN. To make the installation process faster, I copied over an `/etc/rc.config.d/netconf` file from another system on the site LAN and used it on the new system. When I booted up the system, the site LAN went down.

Answer: You probably didn't assign a new IP address and host name prior to rebooting the system. If any two systems on the LAN have the same IP address and host name, the LAN will go down. Check the IP address in the `/etc/rc.config.d/netconf` file against the IP address of your system and other systems on your network map to be sure that no duplicate IP addresses exist on the LAN.

Related Documentation: Refer to the `ifconfig(1M)` man page.

Determining interface name: How do I determine the name of the interface to be configured?

Answer: Use the `lanscan` command to determine the hardware path of the interface card that you want to configure. Then use the value displayed for the “Net-Interface Name PPA” field as the interface name.

Multiple LAN interfaces, intermittent failures: I have been having problems getting the two LAN interfaces on my system to operate at the same time. Occasionally the ethernet cards stop communicating with remote systems. When this happens, the remote system also cannot communicate with the local system.

Answer: Check that the two interfaces on your system do not have the same network number or, if you are subnetting, the same subnet address. If both LAN interfaces have the same value in the network (subnet) address portions of the IP address, the cards may not be enabled simultaneously (although they may both run separately.)

Related Documentation: Refer to Chapter 6, “Network Addressing”.

New system, can’t reach some subnets: I recently tried to add a new system onto a subnet on our site LAN, and I am not able to communicate successfully with all LANs on the network.

Answer: Check the routing table to make sure the route for the LAN you are trying to communicate with has been properly configured. Execute `netstat -rvn` on both ends. Verify the subnet address, netmask and gateway.

Related Documentation: Refer to “Assigning Subnet Addresses” in Chapter 6.

Configuring address 127.0.0.1: I tried to add the IP address, 127.0.0.1, and the system won't accept it.

Answer: Addresses with the format 127.n.n.n are reserved as loopback addresses. Select another IP address. You can obtain Class C addresses that are unique within the ARPANET by contacting Government Systems, Inc.

Related Documentation: Refer to "Assigning an Internet Address" in Chapter 6.

Displaying station address: How do I locate the station address of my LAN card?

Answer: Use the `lanscan` command to display the station addresses of all LAN cards in the system:

```
lanscan
```

Related Documentation: Refer to the `lanscan(1M)` man page.

Resetting LAN card: How do I reset the LAN card?

Answer: Run the `lanadmin` diagnostic by entering the following sequence of commands, where *x* is the Physical Point of Attachment (PPA) of the interface you want to reset. (Use the `lanscan` command to determine the PPA of the interface on the system.)

```
lanadmin
lan
ppa
x
reset
quit
```

Related Documentation: Refer to the `lanadmin(1M)` man page.

Tracing: What's the best way to obtain and format tracing information when I am using the `nettl` utility?

Answer: HP field engineers recommend the following commands:

To begin LAN and loopback tracing, execute:

```
nettl -tn pduin pduout -e ns_ls_driver -f filename
```

To end LAN tracing, execute:

```
nettl -tf -e all
```

To format your entire LAN trace (no filtering), execute:

```
netfmt -Nnl -f filename.TRC1 > fmt1
netfmt -Nnl -f filename.TRC0 > fmt0
```

The file, *filename*`.TRC0` is the most recent trace file. If this file does not contain the trace information you are looking for, check the *filename*`.TRC1` file.

To format your LAN trace using a filter file, execute:

```
netfmt -c filterfile -N -f filename.TRC0 > fmt0
```

`nettl` appends TRC0 or TRC1 to the name you give the raw trace file.

Related Documentation: Refer to Chapter 5, “LAN Resources.”

Intermittent networking problems: I'm experiencing intermittent networking problems on my computer. What should I check to ensure proper operation of my networking software?

Answer: Upper layer software often requires loopback. Check `/etc/rc.config.d/netconf` to be sure that the loopback entry is correct. The line in `netconf` file should read:

```
LOOPBACK_ADDRESS=127.0.0.1
```

Related Documentation: Refer to chapter 6, “Network Addressing”.

Performance: I've noticed a significant drop in system response time and performance. What steps can I take to improve it?

Answer: Performance may be affected by many different factors. Sometimes removing pseudo drivers from the kernel for networking software that you may not be using improves performance. The problems may also be in the upper layer software (`ftp` or `telnet`).

Also, it is possible that too little memory is allocated to hold fragmented messages in the IP layer. IP messages may be fragmented into smaller parts when the message is sent through the system. The fragments must be held in memory for some time so that the entire message can be reassembled because the fragments arrive at the destination at different times and possibly out of order. Normally, fragmentation reassembly memory is limited arbitrarily so that incomplete messages do not consume all of memory, which could cripple the system. During stressful networking activity, some fragments might never be delivered because they are typically dropped in transit, for example, due to a collision or resource limitations on an intermediate system. However, fragments might also not be delivered (“dropped”) if there is insufficient fragmentation reassembly memory on the destination system during periods of high network activity. This can degrade performance due to retransmissions of data. If the problem is due to a high number of fragments dropped after time-out (see the output from the command `netstat -sp ip`), you might want to increase the size of the fragmentation reassembly memory by changing the `ip_reass_mem_limit` value using the `ndd` command. (The default is 2 MB for the system.) Enter the command `/usr/bin/ndd -h` to display `ndd` parameters and their use.

Deferred transmissions/collisions: Why is there a significant increase in the number of deferred transmissions and collisions on my network?

Answer: On IEEE802.3/Ethernet networks, a collision occurs when two or more stations try to transmit data simultaneously. A deferred transmission occurs if the network is busy when a station attempts to transmit data. The number of collisions and deferred transmissions on a node is directly related to the network load. As the network load increases, the number of collisions and deferred transmissions also increase.

When high-performance systems are placed on a LAN with lower-performance systems (HP or non-HP systems), it is possible for the high-performance systems to use a higher percentage of the LAN bandwidth with network traffic intensive applications. High-performance systems generate network traffic at a 10Mbits/s link rate, and lower-performance systems cannot match this rate. Heavily loaded LAN networks can result in lower throughput performance on lower-performance systems.

In general, the short term average load on an IEEE802.3/Ethernet LAN should not exceed more than 70% of the total bandwidth of the LAN. When it does exceed 70% of the total bandwidth, network performance begins to degrade due to an increase in collisions and deferred transmissions. When it consistently exceeds 70% of the total bandwidth, you may need to reconfigure the systems on your LAN. If you notice throughput/performance degradation on your system, contact your local HP Representative for additional assistance and consultation.

"No such interface": After I booted my server, I found that networking failed. I found the following error in the `/var/adm/rc.log` file:

```
ifconfig lan0: no such interface
```

How do I resolve this problem?

Answer: This problem is caused by the LAN driver software disabling the LAN card because it was not connected to the LAN, or the LAN was down. Use `lanadmin` to reset the LAN card and run `ifconfig` to bring the card up.

"No such interface": When I configure an interface, `ifconfig` returns the error "no such interface." What should I do?

Answer: The numeric portion of the interface name is incorrect. Run the `lanscan` command to obtain a list of interface names.

"Plumbing error": When I configure an interface, `ifconfig` returns a "Plumbing error" message. What should I look for?

Answer: The interface name specified in the `ifconfig` run string is not defined in the `/dev` directory or is not a streams driver. The network device files `/dev/ip` and `/dev/tcp` are not defined.

Can't communicate outside local supernet: I recently tried to set up a supernet on my LAN. The systems in the supernet can communicate with one another, but they cannot communicate with systems outside the supernet.

Answer: Check the routing table on your system and the node you want to communicate with. If the system you want to communicate with does not support supernetting, you will have to configure a network route for each of the networks in the supernet. If the system you want to communicate with supports supernetting, you will only need to add a network route for the supernet.

LAN Interface Card Statistics

This section contains descriptions of the RFC 1213 MIB II statistics fields for LAN interface cards which are displayed on the screen with the `display` command in `lanadmin` LAN Interface Test Mode. A description of each field follows the display.

```
                    LAN INTERFACE STATUS DISPLAY
                     Mon, May 19,2003 11:16:22
Network Management ID               = 1
Description                        = lan0 HP PCI Core I/O 1000Base-T Release B.11.23.00.01
Type (value)                        = ethernet-csmacd (6)
MTU Size                            = 1500
Speed                               = 10000000
Station Address                     = 0x80009266c3f
Administration Status (value)       = up(1)
Operation Status (value)            = up(1)
Last Change                         = 5917
Inbound Octets                      = 1862255299
Inbound Unicast Packets             = 18765
Inbound Non-Unicast Packets         = 7644729
Inbound Discards                    = 48847
Inbound Errors                      = 0
Inbound Unknown Protocols           = 3877238
Outbound Octets                     = 1304687
Outbound Unicast Packets            = 18721
Outbound Non-Unicast Packets        = 19
Outbound Discards                   = 0
Outbound Errors                     = 0
Outbound Queue Length               = 0
Specific                            = 655367

Ethernet-like Statistics Group

Index                               = 0
Alignment Errors                    = 0
FCS Errors                          = 0
Single Collision Frames             = 0
Multiple Collision Frames           = 0
Deferred Transmissions              = 0
Late Collisions                     = 0
Excessive Collisions                = 0
Internal MAC Transmit Errors        = 0
Carrier Sense Errors                = 0
Frames Too Long                     = 0
Internal MAC Receive Errors         = 0
```

RFC 1213 MIB II STATISTICS

Description — A textual string containing information about the interface. This string includes the name of the manufacturer, the product name and the version of the hardware interface.

Type (value) — The type of interface, distinguished according to the physical/link protocol(s) immediately below the network layer in the protocol stack. It will have one of the following values:

Table 4-1 Type(value)/Comments

other (1)	None of the following
regular1822 (2)	
hdh1822(3)	
ddn-x25(4)	
rfc877-x25(5)	
ethernet-csmacd(6)	
iso88023-csmacd(7)	
iso88024-tokenBus(8)	
iso88025-tokenRing(9)	
iso88026-man(10)	
starLan(11)	
proteon-10Mbit(12)	
proteon-80Mbit(13)	
hyperchannel(14)	
fddi(15)	
lapb(16)	
sdlc(17)	
ds1(18)	T-1
el(19)	European equivalent of T-1

Table 4-1 Type(value)/Comments (Continued)

basicISDN(20)	
primaryISDN(21)	Proprietary serial
proPointToPointSerial(22)	
ppp(23)	
softwareLoopback(24)	
eon(25)	CLNP over IP [11]
ethernet-3Mbit(26)	
nsip(27)	XNS over IP
slip(28)	generic SLIP
ulta(29)	ULTRA technologies
ds3(30)	T-3
sip(31)	SMDS
frame-relay(32)	

MTU Size — The size of the largest datagram which can be sent/received on the interface specified in octets.

Speed — An estimate of the current bandwidth of the interface in bits per second. For interfaces which do not vary in bandwidth or for those where no accurate estimates can be made, this object contains the nominal bandwidth.

Station Address — The interface address at the protocol layer immediately below the network layer in the protocol stack. For interfaces which do not have such an address, such as serial line, this object contains an octet string of zero length.

Administration Status — The desired state of the interface It will have one of the following values:

Table 4-2 Administration Status

up(1)	Ready to pass packets
down(2)	Not operative

Table 4-2 Administration Status (Continued)

testing(3)	In test mode

Operation Status The current operational state of the interface. It will have one of the following values.

Table 4-3 Operation Status

up(1)	Ready to pass packets
down(2)	Not operative
testing(3)	In test mode

Last Change The value of SysUpTime at the time the interface entered its current operational state. If the current state was entered prior to the last reinitialization of the local network management subsystem, then this object contains a zero value.

Inbound Octets The total number of octets received on the interface, including framing characters.

Inbound Unicast Packets The number of subnetwork-unicast packets delivered to a high-layer protocol.

Inbound Non-Unicast Packets The number of non-unicast (subnetwork-broadcast or subnetwork-multicast) packets delivers to a higher-layer protocol.

Inbound Discards The number of inbound packets that were discarded even though no errors had been detected, to prevent their being delivered to a higher-layer protocol. One possible reason for discarding such a packet could be to free up buffer space.

Inbound Errors The number of inbound packets that contained errors preventing them from being deliverable to a higher-layer protocol.

Inbound Unknown Protocols The number of packets received via the interface which were discarded because of an unknown or unsupported protocol.

Outbound Octets The total number of octets transmitted out of the interface, including framing characters.

Outbound Unicast Packets The total number of packets that higher-level protocols requested be transmitted to a subnetwork-unicast address, including those that were discarded or not sent.

Outbound Non-Unicast Packets The total number of packets that higher-level protocols requested be transmitted to a non-unicast (a subnetwork-broadcast or subnetwork-multicast) address, including those that were discarded or not sent.

Outbound Discards The number of outbound packets that were discarded even though no errors had been detected to prevent their being transmitted. One possible reason for discarding such a packet could be to free up buffer space.

Outbound Errors The number of outbound packets that could not be transmitted because of errors.

Outbound Queue Length The length of the output packet queue (in packets)

RFC 1284 Ethernet-Like Interface Statistics

Index A value that uniquely identifies an interface to an Ethernet-like medium.

Alignment Errors A count of frames received on a particular interface that are not an integral number of octets in length and do not pass the FCS check.

FCS Errors A count of frames received on a particular interface that are not an integral number of octets in length and do not pass the FCS check.

Single Collision Frames A count of successfully transmitted frames on a particular interface for which transmission is inhibited by exactly one collision.

Multiple Collision Frames A count of successfully transmitted frames on a particular interface for which transmission is inhibited by more than one collision.

Deferred Transmissions A count of frames for which the first transmission attempt on a particular interface is delayed because the medium is busy. The count represented by an instance of this object does not include frames involved in collisions.

Late Collisions The number of times that a collision is detected on a particular interface later than 512 bit-times into the transmission of a packet.

Excessive Collisions A couple of frames for which transmission on a particular interface fails due to excessive collisions.

Internal MAC Transmit Errors A count of frames for which transmission on a particular interface fails due to an internal MAC sublayer transmit error.

Carrier Sense Errors The number of times that the carrier sense condition was lost or never asserted when attempting to transmit a frame on a particular interface.

Frames Too Long Account of frames received on a particular interface that exceed the maximum permitted frame size.

Internal MAC Receive Errors A count of frames for which reception on a particular interface fails due to an internal MAC sublayer receive error

100Base-T Checklist

In case of trouble with 100Base-T LAN links, you can use the following procedures to troubleshoot your network problems:

- *Verify Cabling*: make sure the connection is secured, UTP Category 5 is used, the card is well inserted. Also, assure the cable length is not within 35 - 41 meters. Check the cable running from the HP adapter to the Switch, and the Switch port, in case either is defective.

  — If the cable length is between 35 - 41 meters, (or 114 - 133 feet), then expand or reduce the length so that the cable is less than 35 meters or greater than 41 meters, keeping within 100Base-T specifications.

  — Have your site technician verify that the pair assignments and color codes of the RJ45 connector pins match the following recommended version:

    — Receive Signal: pin 1 = White and pin 2 = Orange

    — Transmit Signal: pin 3 = White and pin 6 = Green

  — Double-check your existing punch-down blocks in your networking environment. Punch-down blocks may affect the characteristics of the medium and therefore the problem seen with the 35-41 meter length cable may vary in length.

  — Some visible symptoms that might occur when the cable length is between 35 - 41 meters are:

    — Link Status is Down: LED light color turns amber because card negotiating with switch defaults to 10Mb/s instead of 100Mb/s. Or the LED lights are intermittent between 10Mb/s and 100Mb/s. They blink between 10Mb/s and 100Mb/s and keep doing that.

    — There is no traffic or there is high rate of packet loss.

    — To verify if the link is not yet established, format log file using the following command and syntax:

```
netfmt -LN -f /var/adm/nettt.LOG* > outfile
```

  Once the nettl log file is formatted, look for a string such as “...10/100Base-T driver detected bad cable connection between the adapter in slot # and the hub or switch.”

Or use the command `lanscan` to get the name of the Net Interface Name or ppa number or nmid number. Once you have the ppa number, for example lan17, you can issue these following commands one at a time:

```
lanadmin

lan

ppa

17

display
```

- Look for the value of `Operation Status`. It should say "`DOWN`".

- Ensure that the host system contains the correct patch level. To find out which version of the driver is currently installed, execute the command:

  ```
  what /stand/vmunix | grep btlan
  ```

  Or use the command:

  ```
  swlist -l product | grep 100
  ```

  This should display an ASCII string that indicates the Version and possibly Patch level of the driver. Many problems have been resolved with newer versions of the driver, which can be obtained from the latest patches for the 100Base-T cards along with any dependent patches. Work with the HP Response Center to determine if there is a newer driver that fixes the symptoms of the problem you are facing.

  There are three types of patches that are recommended:

  ```
  - Patches for software driver

  - Patch(es) for lanadmin - current lanadmin provides the
  lanadmin -x -X functionality.

  - Patch(es) for SAM - all systems need an updated SAM patch because SAM can "step on"
  the 100Base-T configuration file Duplex mode.
  ```

- Display the card's current speed and duplex, issue the command:

  ```
  lanadmin -x NMID/PPA (nmid/ppa comes from the lanscan command)
  ```

- If problems persist, if a switch is used, manually set the switch port to the desired speed and duplex.

- Manually set the card to the desired speed:

  ```
  - for 100Base-T hubs: 100 Half Duplex

  - for 10Base-T hubs: 10 Half Duplex
  ```

- Manual setting can be done *temporarily* with the command:

```
lanadmin -X speed/duplex nmid/ppa
```

- Manual setting can be *permanently* set by modifying the configuration file in `/etc/rc.config.d` or by using SAM (recommended way).
- If problems persist: connect the card to a switch or hub that is known to be good.
- If this connection works, contact the HP Response Center with the Switch/Hub information. This may be an IOP (Interoperability) problem.
- If this connection fails, the card may be bad and may need to be replaced.

Diagnostic Flowcharts

Below is a summary of the types of network tests in the diagnostic flowcharts. To diagnose your problem, first check the connections and configuration on your system (Flowcharts 1 through 5). If this does not solve your problem, use flowcharts 6 through 11 to test/verify connectivity with a remote system.

1, 2 & 3	Configuration Test
4 & 5	Network Level Loopback Test
6	Transport Level Loopback Test (using Internet Services)
7	Link Level Loopback Test
8	LAN Connections Test
9 &10	Gateway Remote Loopback Test
11	Subnet Test

Configuration Test: Verifies the configuration of the network interface on a host using the `lanscan`, and `ifconfig` commands.

Network Level Loopback Test: Checks roundtrip communication between Network Layers on the source and target host using the `ping` diagnostic.

Transport Level Loopback Test: Checks roundtrip communication between Transport Layers on the source and target host using Internet Services `telnet` and `ftp` commands.

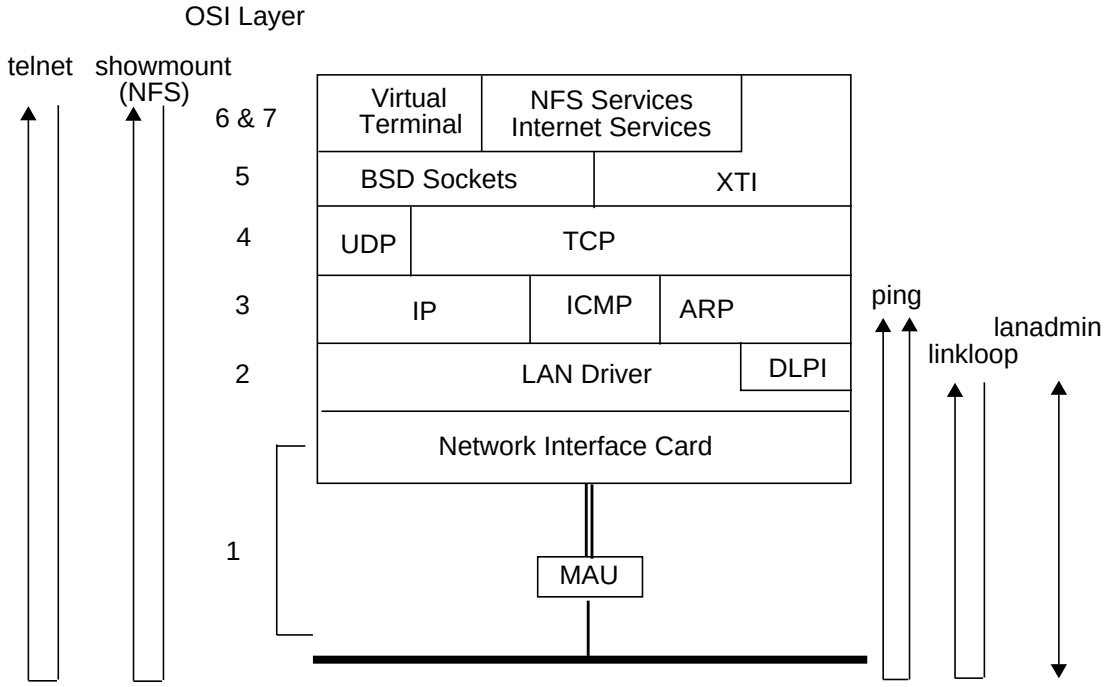
Link Level Loopback Test: Checks roundtrip communication between Link Levels on the source and target host using the `linkloop` diagnostic.

LAN Connections Test: Checks the connections between the LAN media and any component between the LAN media and individual LAN cards.

Gateway Remote Loopback Test: Checks general network connections through a gateway.

Subnet Test: Verifies the correct use of subnets.

Figure 4-1 Loopback Tests



The loopback tests shown in Figure 4-1 are used to isolate a network communication problem that may be software- or hardware-related. In any case, you should first have checked that the problem is not due to a recent configuration change.

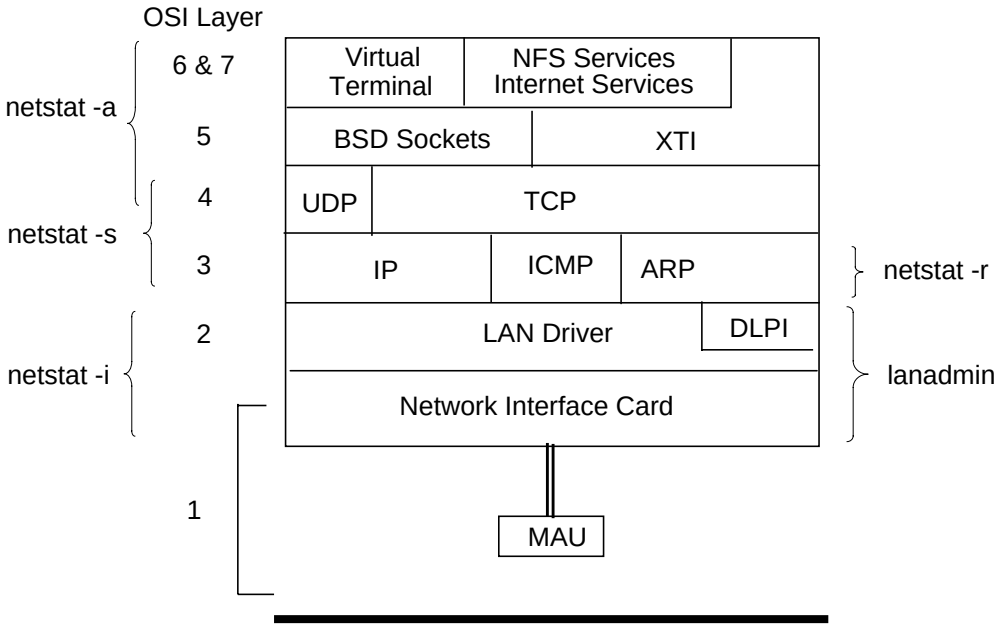
The network level loopback test, `ping`, commonly is tried first. It is fast, efficient, and it does not require superuser privileges. If the connection passes this test, you know the problem is at OSI Layer 4 or above. Go on to the transport level loopback test.

The Transport Level Loopback Test can be implemented using Internet Services. In this case, you use `telnet` and `ftp` to systematically focus on a problem.

If the network level loopback test failed, the problem is in OSI Layer 3 or below. In this case, continue with the link level loopback test, using `linkloop`, to isolate a problem to OSI Layer 2 or below.

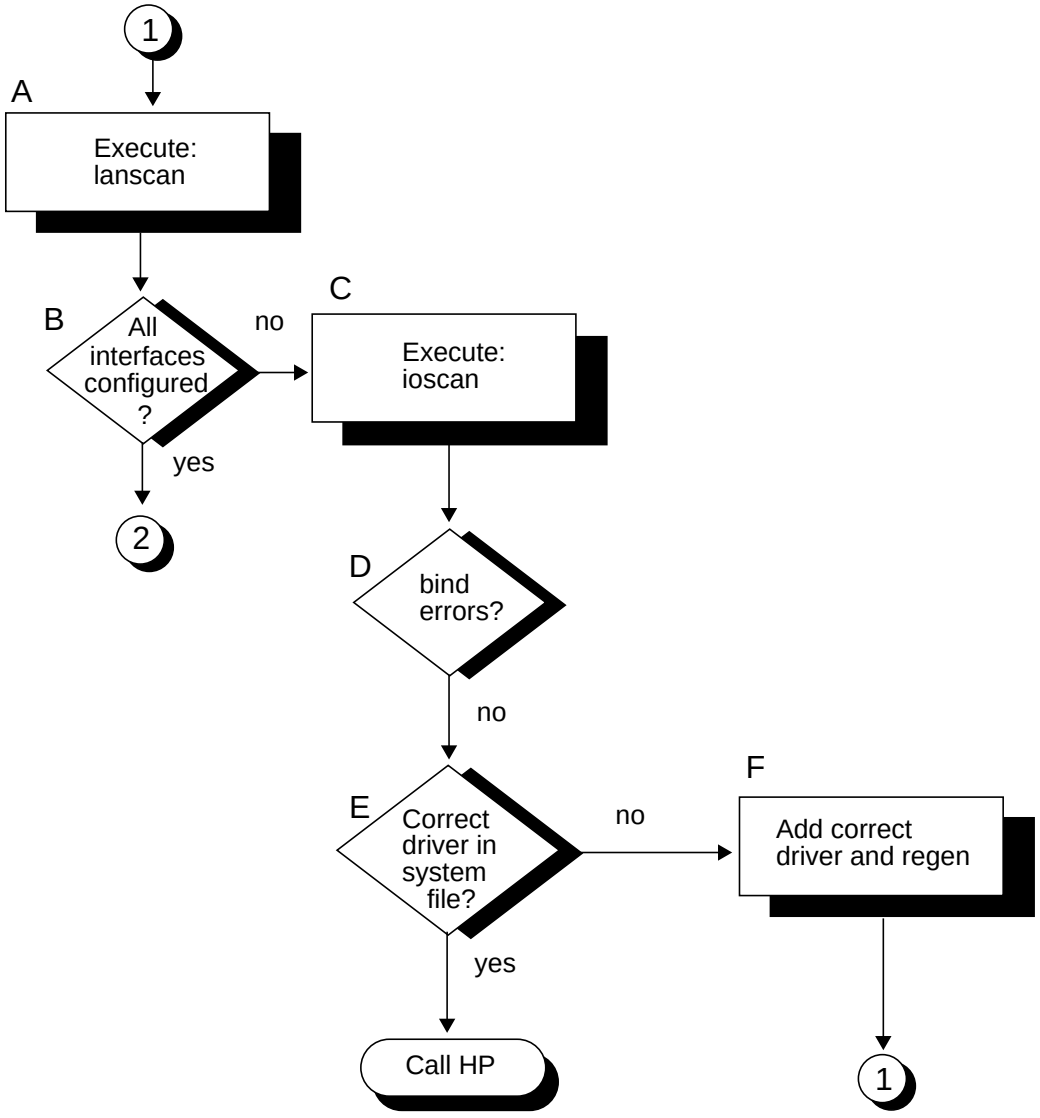
The `netstat` utility reports network and protocol statistics regarding traffic and the local LAN interface. As shown in Figure 4-2, there are many options to `netstat`. The options that are most useful are those which display information that is not available through other commands such as `ping` and `lanadmin`, for example, `-a` and `-r` options. You can also use the `lanadmin` command to obtain LAN driver statistics. For more detailed information on these diagnostics, refer to the `netstat(1M)` and `lanadmin(1M)` man pages.

Figure 4-2 Networking Software Statistics



Flowchart 1: Configuration Test

Figure 4-3 Flowchart 1



Flowchart 1 Procedures

A. *Execute: lanscan*. Execute `lanscan` to display information about LAN cards that are successfully bound to the system. For example, to check the cards on `/stand/vmunix`, enter:

```
lanscan
```

B. *All interfaces configured?* `lanscan` is successful if the output shows information about every card in the hardware backplane.

C. *Execute ioscan*. Execute the `ioscan` command to check for bind errors.

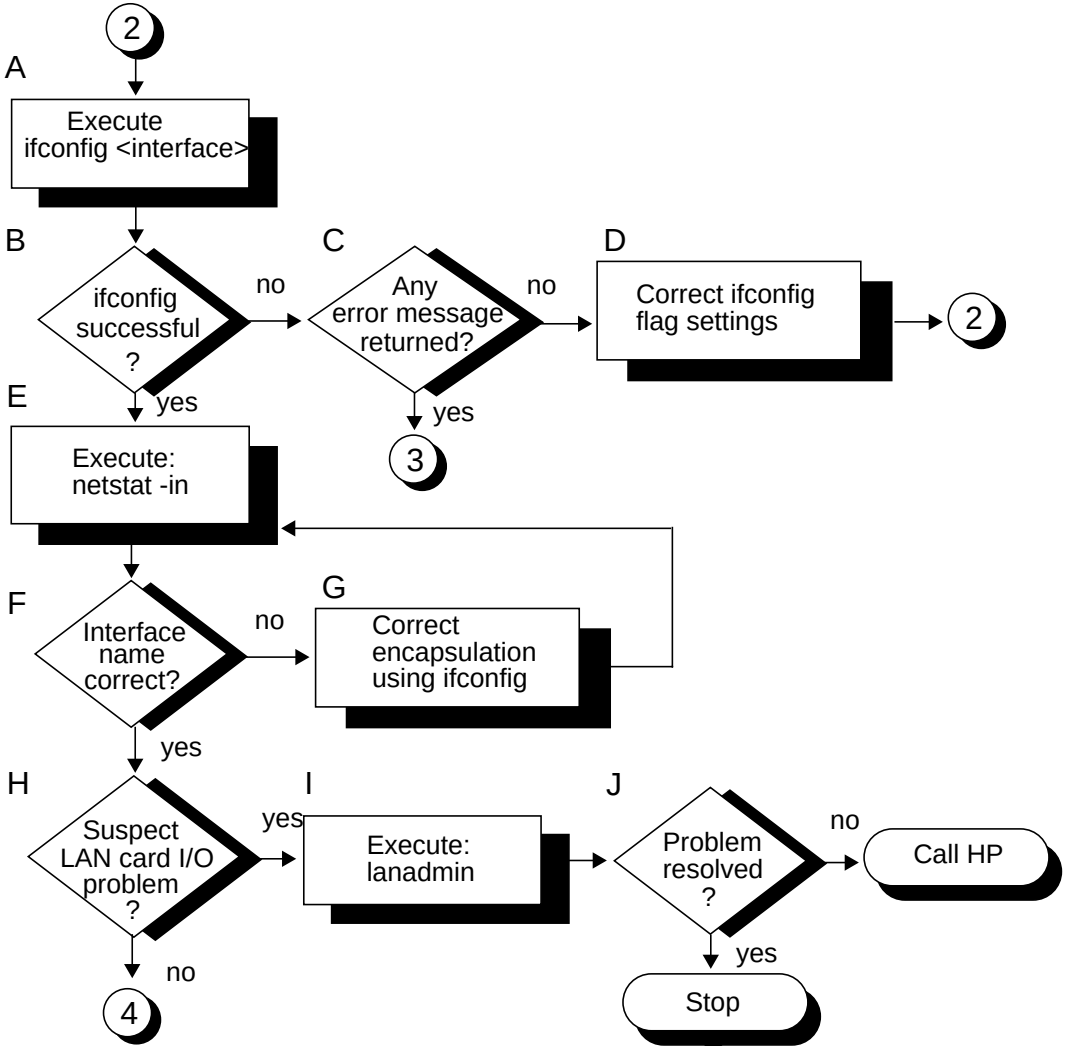
D. *bind errors?* If a bind error exists, check that the hardware has been properly installed.

E. *Correct driver in /stand/system file?* Refer to "Creating a New Kernel for HP-UX Systems" in Chapter 3 for a list of LAN drivers.

F. *Add correct driver and regen*. Edit the `/stand/system` file to contain the correct driver for your system type.

Flowchart 2: Configuration Test continued

Figure 4-4 Flowchart 2



Flowchart 2 Procedures

A. *Execute: ifconfig <interface>*. Execute `ifconfig` on the interface you want to test. For example, to check LAN interface `lan0`, enter:

```
ifconfig lan0
```

B. *ifconfig successful?* `ifconfig` is successful if the output shows the correct Internet address and the flags, typically: UP,BROADCAST,RUNNING.

C. *Any error message returned?* If `ifconfig` is not successful, and an error message appears, go to Flowchart 3. Flowchart 3 shows common error messages and what to do for each.

D. *Correct ifconfig flag settings.* If `ifconfig` returns an incorrect flag setting, re-execute the command with the proper setting. For more information, refer to the `ifconfig(1M)` man page. Start again with Flowchart 2, as necessary.

E. *Execute: netstat -i*. If `ifconfig` is successful, you know the network interface has been configured correctly. Using `netstat`, you can return statistics which show the interface is operational. Enter:

```
netstat -in
```

`netstat` statistics give a quick check of key operating parameters. For instance, if the `Opkts` value does not change after attempting the file transfer, packets are not being transmitted. Similarly, if the `Ipkts` value does not change, packets are either not being received by the local node or are not being sent by the remote node, which may not be receiving your transmissions.

F. *Interface name correct?* Verify that the name for your interface is what you expect. See the `ifconfig(1M)` man page.

G. *Correct encapsulation using ifconfig.* Use the `ifconfig` command to correct the encapsulation method of your interface. For more information, refer to the `ifconfig(1M)` man page. Go to E.

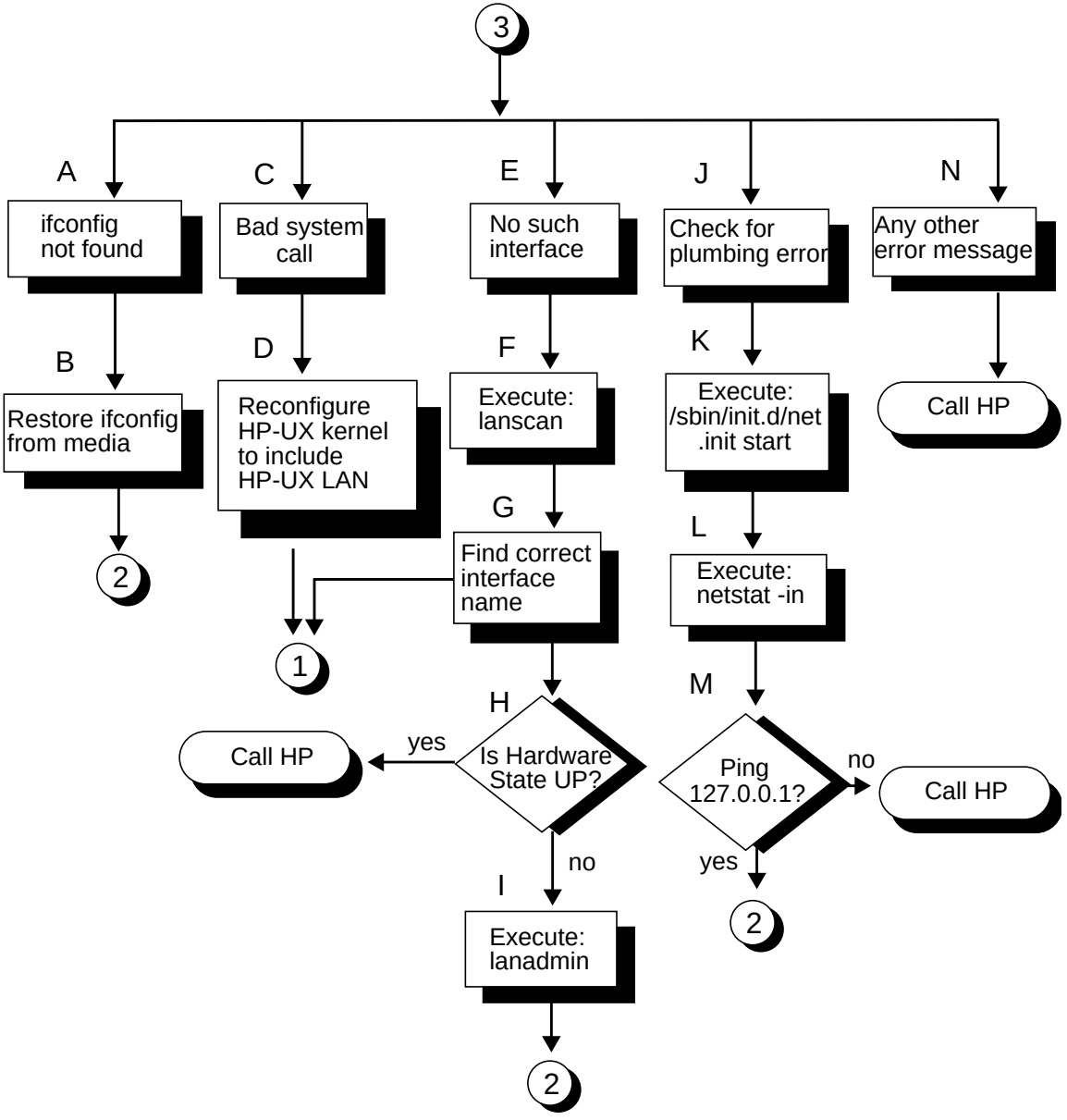
H. *Suspect LAN card I/O problems?* If the statistics indicate possible LAN card problems, go to I, otherwise go to Flowchart 4.

I. *Execute: lanadmin*. Use `lanadmin` to ensure the LAN card is operational. If the values of the `Ierrs` and `Oerrs` fields increase substantially during a file transfer attempt, this can indicate transmission or reception problems. Refer to “LAN Interface Card Statistics” in this chapter for more information.

J. *Problem resolved?* If you have found and corrected the LAN card problem, stop. If not, call your HP representative for help. Be prepared to discuss the problem as described in “Contacting your HP Representative” at the end of this chapter.

Flowchart 3: Configuration Test continued

Figure 4-5 Flowchart 3



Flowchart 3 Procedures

A. *ifconfig not found*. The command has been relocated on the system, deleted, or `/usr/sbin` is not in your PATH.

B. *Restore /etc/ifconfig from media*. You can restore `ifconfig` from the last good backup tape or your install/update tape. Go to Flowchart 2.

C. *Bad system call*. Networking is not configured into the HP-UX kernel.

D. *Reconfigure HP-UX kernel to include HP-UX LAN software*. Edit the `/stand/system` file to include HP-UX LAN software. Refer to chapter 3 for a list of LAN drivers.

E. *No such interface name*. The interface name passed to `ifconfig` does not exist on the system. Check spelling and names of interfaces on the system using `lanscan`.

If you have more than one LAN card, make sure the number of LAN cards has been configured into the kernel and that an `ifconfig` command has been executed for each.

F. *Execute lanscan*. Execute `lanscan` to display information about the LAN cards in your system.

G. *Find correct interface name*. Using the correct interface name, start again with Flowchart 1.

H. *Is Hardware State UP?* Verify the state of the hardware with the output from the `lanscan` command. If the Hardware State is up call your HP representative for help. Otherwise go to I.

I. *Execute lanadmin*. Use the `lanadmin` command to reset the LAN card. Go to Flowchart 2.

J. *Check for the plumbing error: Bad file number*. If you see this error, the TCP/IP stack has not yet been initialized.

K. *Initialize the TCP/IP stack*. Execute the command `/sbin/init.d/net.init start`.

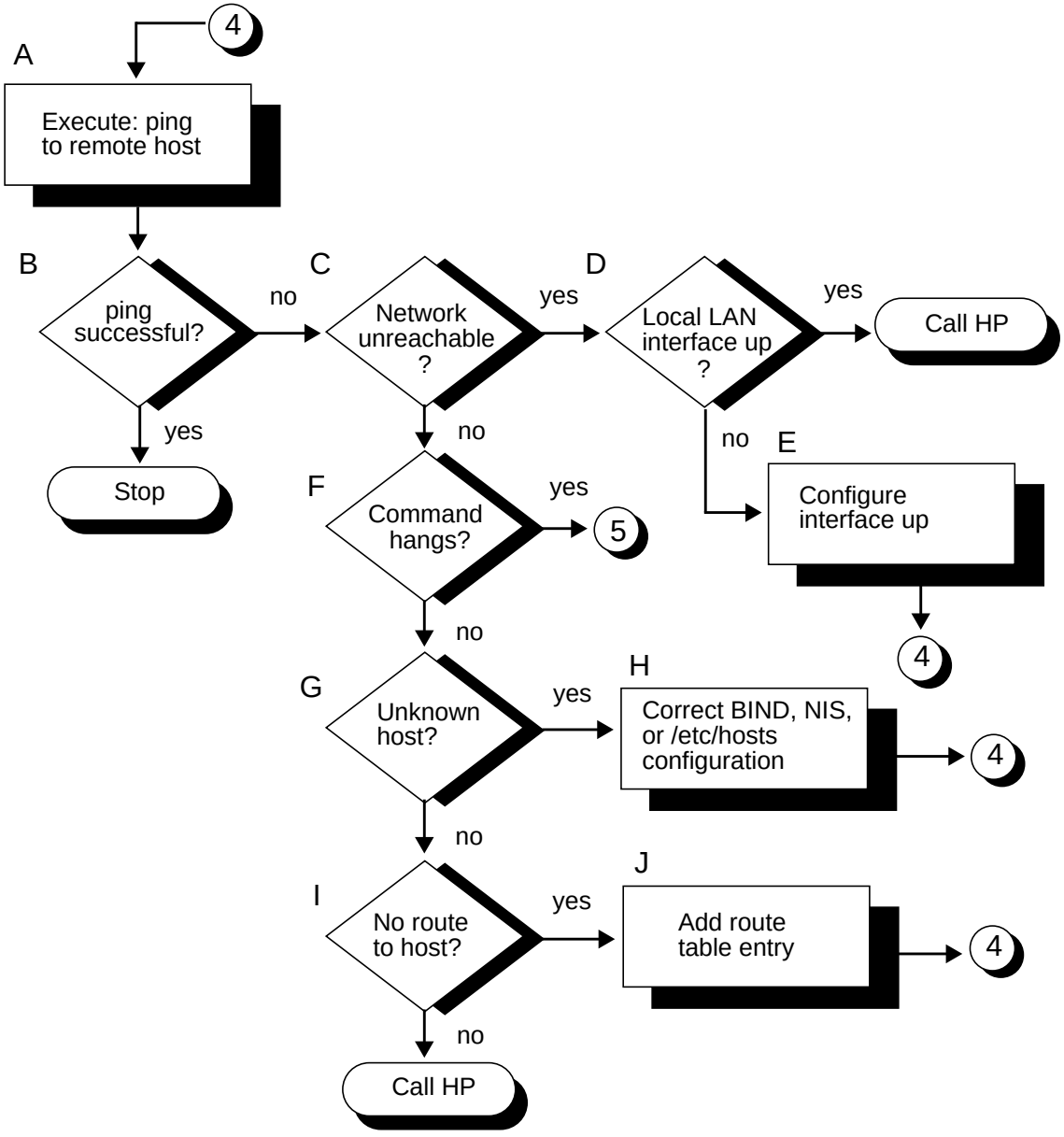
L. *Check the status of the network*. Use the command `netstat -in`. Make sure that the loopback interface `lo0` appears in the display.

M. *Check the loopback interface*. Use the command `ping 127.0.0.1`. If you cannot ping the loopback interface, call HP. If the ping is successful, go to Flowchart 2.

N. *Any other error message.* If you received an error message not listed on this flowchart, interpret the message and take the appropriate action. If you need assistance, call your HP representative. Be prepared to discuss the problem as described in “Contacting Your HP Representative” in Chapter 5, “LAN Resources.”

Flowchart 4: Network Level Loopback Test

Figure 4-6 Flowchart 4



Flowchart 4 Procedures

A. *Execute: ping to remote host.* Using `ping`, send a message to the remote host with which you are having problems connecting. For example, suppose the remote host is known as 192.6.20.2. Enter:

```
ping 192.6.20.2
```

NOTE HP recommends using the IP address, rather than the hostname, as part of the problem may be an error in the `/etc/hosts` file or connectivity with a name server.

B. *ping successful?* A message is printed on `stdout` for each `ping` packet returned by the remote host. If packets are being returned, your system has network level connectivity to the remote host.

You may find it useful to note what percentage of the total packets are lost, if any. Losing ten percent or more may indicate the network or remote host is extremely busy. If, over a one-day period, `ping` reports a packet loss that you feel is unacceptable, yet connectivity remains, report this to your HP representative.

You may also find it useful to note the round-trip transmission times. Periodically high transmission times may indicate that the network or remote host is extremely busy. Consistently high transmission times may indicate the local host is extremely busy. Make sure that the network event logging masks are not set to values which can impair system performance (such as DEWRP).

C. *Network unreachable?* If so, check the status of the local LAN interface first.

D. *Local LAN interface up?* Execute `ifconfig` on the local interface to be sure it is configured up.

E. *Configure interface up.* If you find the local interface is not up, execute `ifconfig` with the appropriate flags set. Start again with Flowchart 4.

F. *Command hangs?* If a message is not returned after executing `ping`, go to Flowchart 5.

G. *Unknown host?* (Error= Unknown host hostname?) There is a problem with the configuration for the host in the `/etc/hosts` file or on the name server.

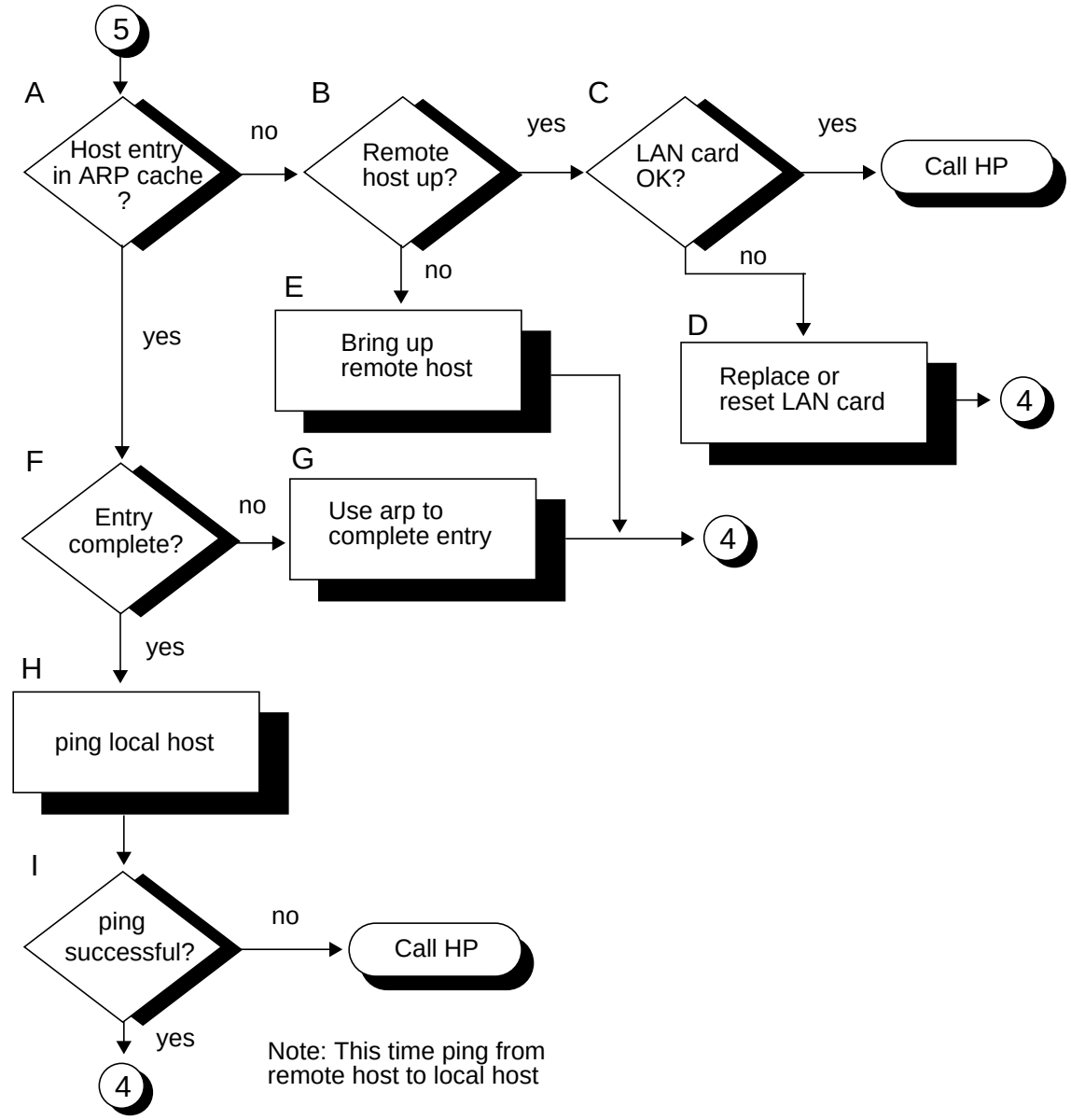
H. *Correct BIND, NIS or /etc/hosts configuration.* Add the missing host name and start again with Flowchart 4.

I. *No route to host?* (Error= Sendto: No route to host?) Use `netstat -rn` to check the routing table. If there is no route table entry for the host, go to J. Otherwise, call your HP representative for help. Be prepared to discuss the problem as described in "Contacting Your HP Representative" in Chapter 5, "LAN Resources".

J. *Add route table entry.* Using `route`, add a route table entry for that host.

Flowchart 5: Network Level Loopback Test continued

Figure 4-7 Flowchart 5



Flowchart 5 Procedures

A. *Host entry in ARP cache?* Using `arp`, check that an entry exists for the remote host in your system's ARP cache. For example, suppose the remote host is known as 192.6.20.2. Enter:

```
arp 192.6.20.2
```

B. *Remote host up?* If there is no ARP cache entry for the remote host, first check that the remote host is up. If not, the remote host has not broadcast an ARP message, and that likely is why there is no entry in the ARP cache.

C. *LAN card O.K.?* Use `lanadmin` to ensure the remote LAN card is operational.

D. *Replace or reset LAN card*. When the LAN card is operational, use `lanadmin` to reset. Refer to the `lanadmin` command description or sample output in this chapter.

E. *Bring-up remote host.* Have the node manager of the remote host bring that system up.

F. *Entry complete?* Perhaps there is an ARP cache entry, but it is wrong or not complete.

G. *Use arp to complete entry.* Using `arp`, enter the correct Station Address. For more information, refer to the `arp(1M)` man page.

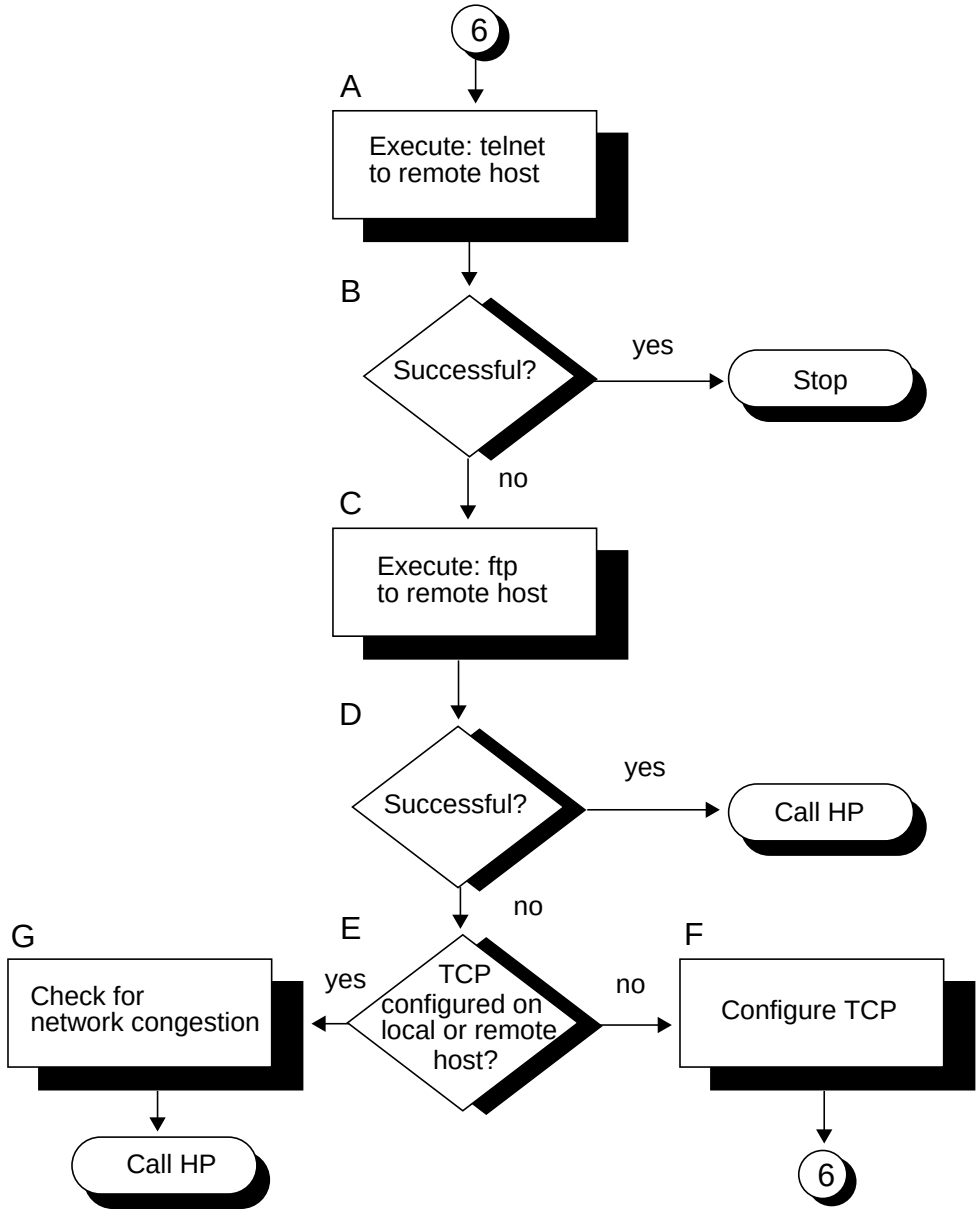
H. *ping local host.* Using `ping`, do an internal loopback on your own system. In other words, `ping` your own IP address. This will determine if the problem is on your end.

I. *ping successful?* If the internal loopback is successful, your system is operating properly to the Network Layer (OSI Layer 3). In addition, you know an ARP cache entry for the remote host exists on your system. If this is true, the network interface or software on the remote host is suspect. Start again with Flowchart 4, but this time `ping` from the remote host to your system.

If the `ping` was not successful, call your HP representative for help.

Flowchart 6: Transport Level Loopback Test (using Internet Services)

Figure 4-8 Flowchart 6



Flowchart 6 Procedures

A. *Execute: telnet to remote host.* Try to establish a `telnet` connection to the remote host.

B. *Successful?* If your `telnet` attempt was successful, stop. The connection is okay through the Transport Layer (OSI Layer 4).

C. *Execute: ftp to remote host.* Unlike `telnet`, `ftp` does not go through a pseudo-terminal driver (pty) on your system. This step tests to see if the pty is why `telnet` failed.

D. *Successful?* If `ftp` is successful, you likely have a problem with a pty on your system. Contact your HP representative.

E. *TCP configured on local or remote host?* Neither `telnet` or `ftp` will work if TCP is not configured on either side of the connection. Check the `/etc/protocols` file on both hosts to be sure TCP is installed and configured.

F. *Configure TCP.* If necessary, install TCP on either or both hosts.

G. *Network congested?* If TCP is installed on both hosts, do a file transfer to another remote host on the network. Use `netstat` to check for lost packets.

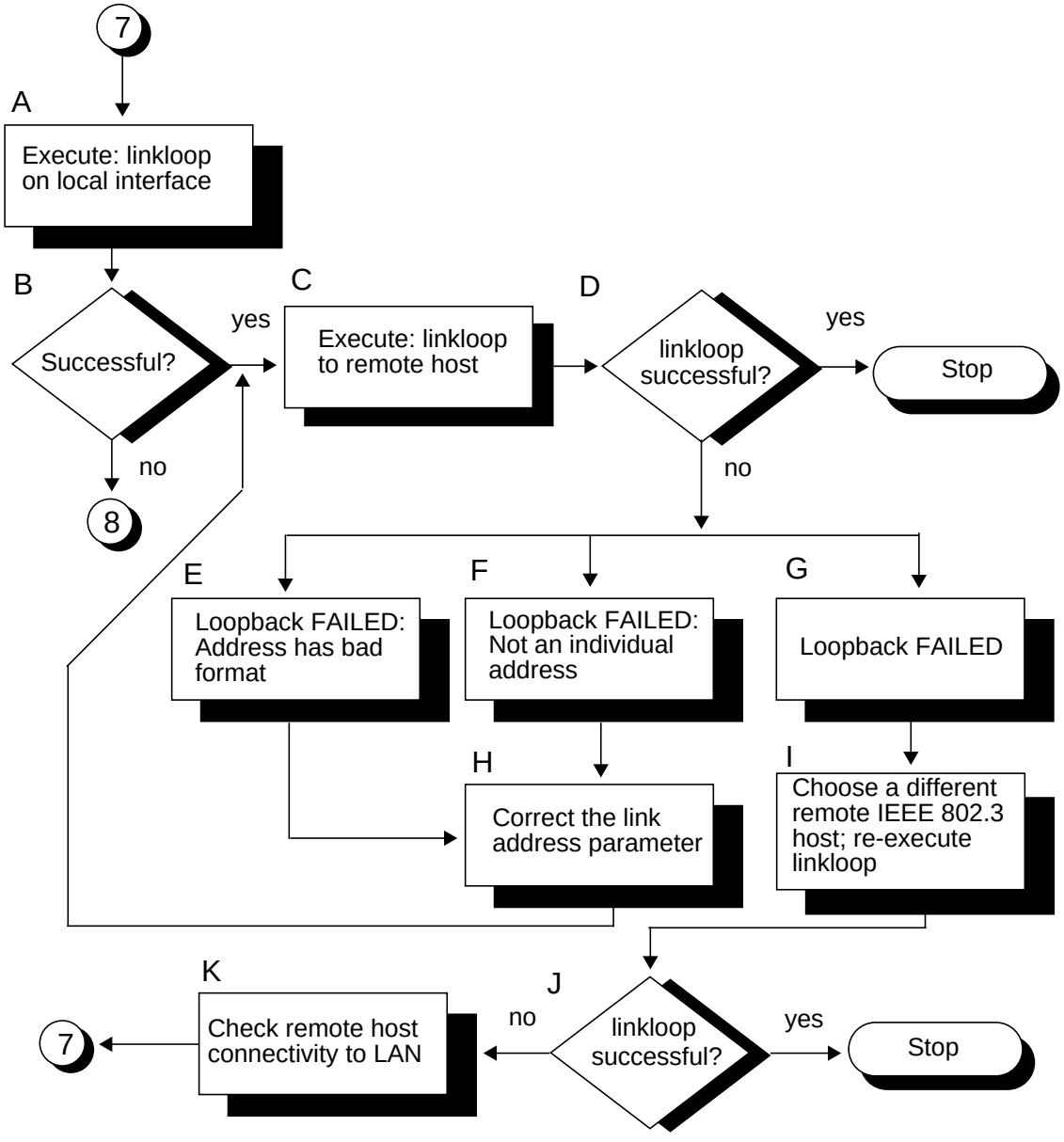
If 10 percent or more packets are lost, the network is extremely busy. If you cannot determine the cause, contact your HP representative for help.

If both `ftp` and `telnet` fail, the `/etc/inetd.conf` file may be configured incorrectly or the `inetd` daemon may not be running on the remote system.

If the problem is not resolved, more detailed diagnostics are required. Again, contact your HP representative.

Flowchart 7: Link Level Loopback Test

Figure 4-9 Flowchart 7



Flowchart 7 Procedures

A. *Execute: linkloop on local interface.* Execute the `linkloop` command with the station address of the local interface. Execute `lanscan` to find the link level address (station address) on the remote host or obtain it from your network map. For more information, refer to the `linkloop(1M)` man page.

B. *linkloop successful?* If not, your LAN card may not be operational. Go to Flowchart 8.

C. *Execute: linkloop to remote host.* Enter the link level address (station address) of the remote host.

D. *linkloop successful?* If the test was successful, stop. Network connectivity is okay through the Link Layer (OSI Layer 2). If not successful, note which error was returned and continue with this flowchart.

E. *Loopback failed; Address has bad format.* The link level address is not correct. Go to H.

F. *Loopback failed; Not an individual address.* The link level address is not correct. The second hexadecimal digit is odd. This means it is a multicast or broadcast address, which is not allowed. The address must be unique to one remote host. Go to H.

G. *Loopback failed.* The remote host did not respond. Go to I.

H. *Correct the link address parameter.* Change the link level address to an allowed value and go to C.

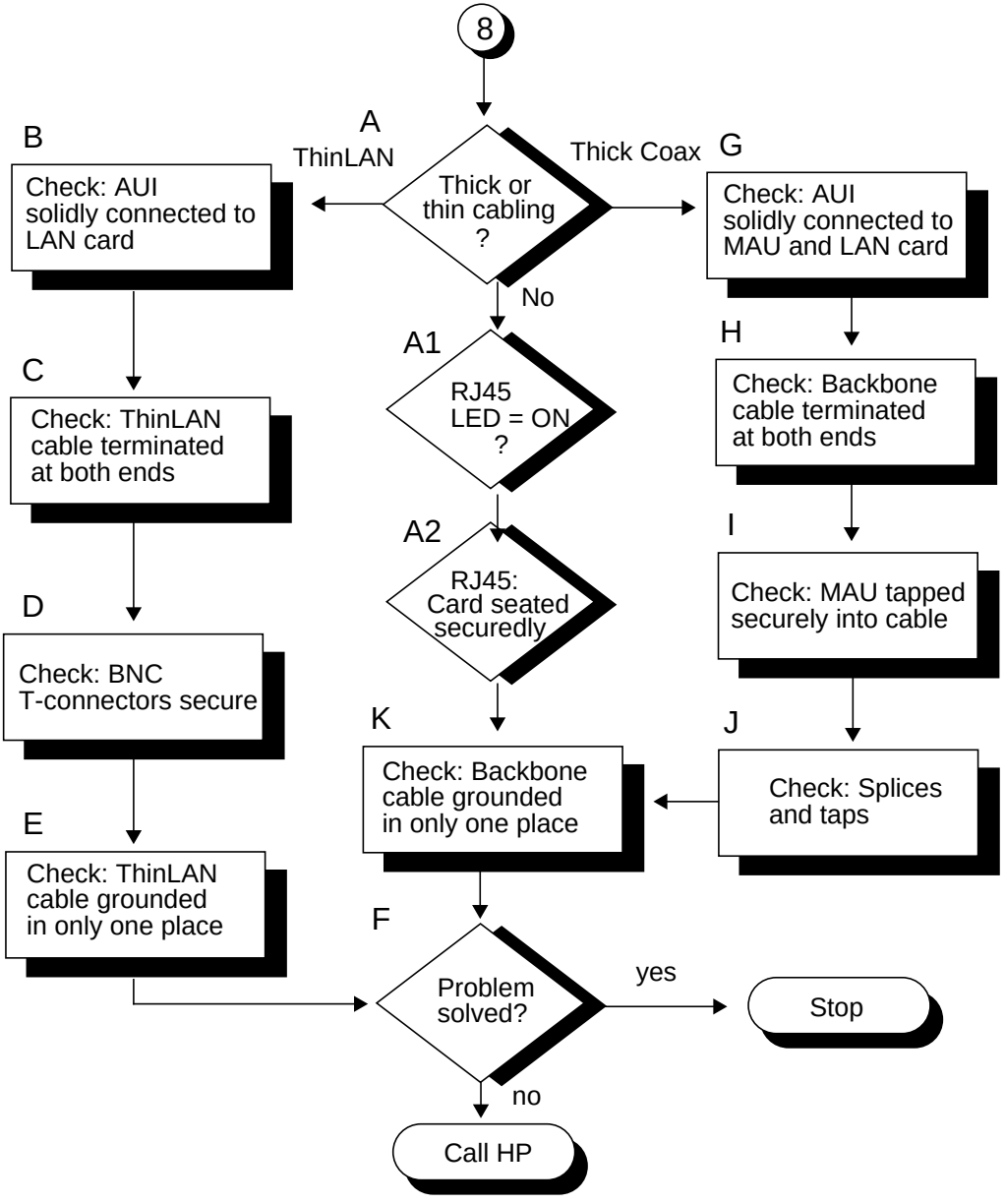
I. *Choose a different IEEE 802.3 host; re-execute linkloop.* Restart this flowchart using a different remote host. NOTE: Make sure the remote host is an HP Server and try again.

J. *linkloop successful?* If the test was successful, stop. Network connectivity is okay through the Link Layer (OSI Layer 2). If not successful, go to K.

K. *Check remote host's connectivity to LAN.* Contact the node manager of the remote host. Check that the host is configured correctly and that its network interface is up. If necessary, use Flowcharts 1 and 12 to verify configuration and connectivity of the remote host.

Flowchart 8: LAN Connections Test

Figure 4-10 Flowchart 8



Flowchart 8 Procedures

A. *Thick or thin cabling?* If your network cabling is the thicker coaxial cabling, continue in the direction marked “Thick Coax.” If your network cabling is the ThinLAN cabling, continue in the direction marked “ThinLAN.”

A1. *RJ45 Adapter?* Verify LEDs. Network Activity and Link Status LED displays: Link Status LED is lit GREEN for a valid link. Network Activity LED is lit AMBER color for both 10 Mbps and 100Mbps. Sometime Link Status LED is not lit up at 10 Mbps mode.

A2. *Card seated securely?* Check adapter installation, reset and reseat adapter. Check for incorrect or faulty network cable or connector. Ensure settings for switch and adapter are the same.

B. *Check: AUI solidly connected to LAN card.* Make sure the AUI cable is solidly connected to the LAN card. If the AUI cable is not connected, turn off the power to the computer before you connect it.

C. *Check: ThinLAN cable terminated at both ends.* Make sure the backbone cable is terminated at both ends.

D. *Check: BNC T-connectors secure.* Make sure each BNC T-connector is securely attached to a BNC connector on the ThinLAN cable and that no intervening cable is between the MAU and the T-connector.

E. *Check: ThinLAN cable grounded in only one place.* Make sure the ThinLAN cable is grounded in only one place.

F. *Problem solved?* If so, stop. If you still have a problem after working through this flowchart, you may have a failed LAN card, an incorrect jumper setting on the LAN card, or a problem with the transmit or receive function of the MAU. Contact your HP representative for help. Be prepared to discuss the problem as described in “Contacting Your HP Representative” at the end of this chapter.

G. *Check: AUI solidly connected to MAU and LAN card.* Make sure the AUI cable is solidly connected to the MAU and the LAN card. If the AUI cable is not connected, turn off the power to the computer before you connect it.

H. *Check: Backbone cable terminated at both ends.* Make sure the backbone cable is terminated at both ends.

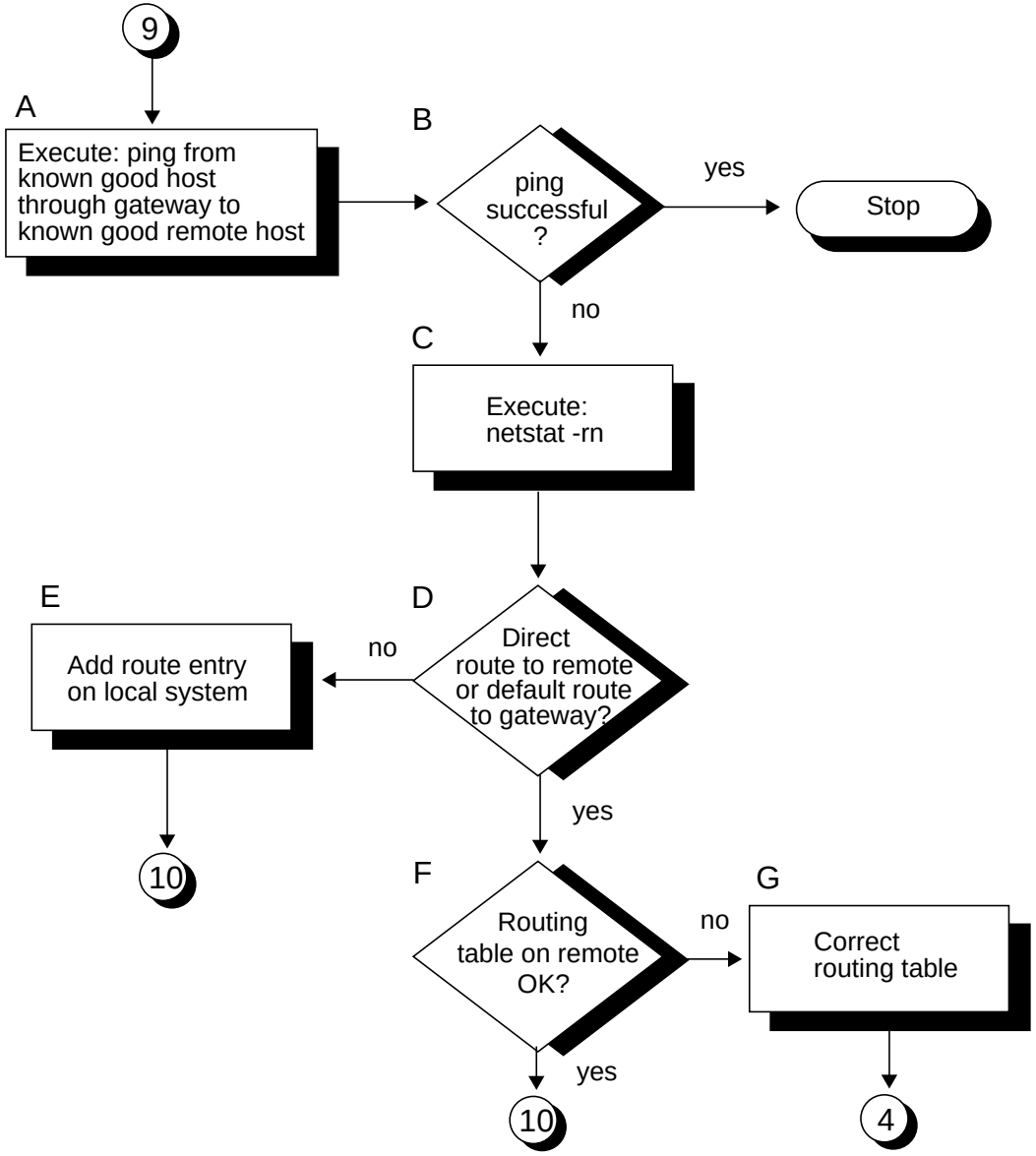
I. *Check: MAU tapped securely into cable.* Make sure the MAU is tapped securely into the backbone cable.

J. *Check: Splices and Taps.* Make sure all splices and taps are secure.

K. *Check: Backbone cable grounded in only one place.* Make sure the backbone cable is grounded in only one place.

Flowchart 9: Gateway Remote Loopback Test

Figure 4-11 Flowchart 9



Flowchart 9 Procedures

A. *Execute: ping from known good host through gateway to known good host on remote network*. This will test gateway connectivity to the remote network. For more information on `ping`, refer to chapter 6.

B. *ping successful?* If the executing `ping` returned does not return successfully, the problem may exist in the routing table for the problem host. Go to C.

C. *Execute netstat -rn*. To display gateway routing information in numerical form, execute:

```
netstat -rn
```

D. *Direct route to remote or default route to gateway?* If the route exists, go to F. If not, go to E to add a new route.

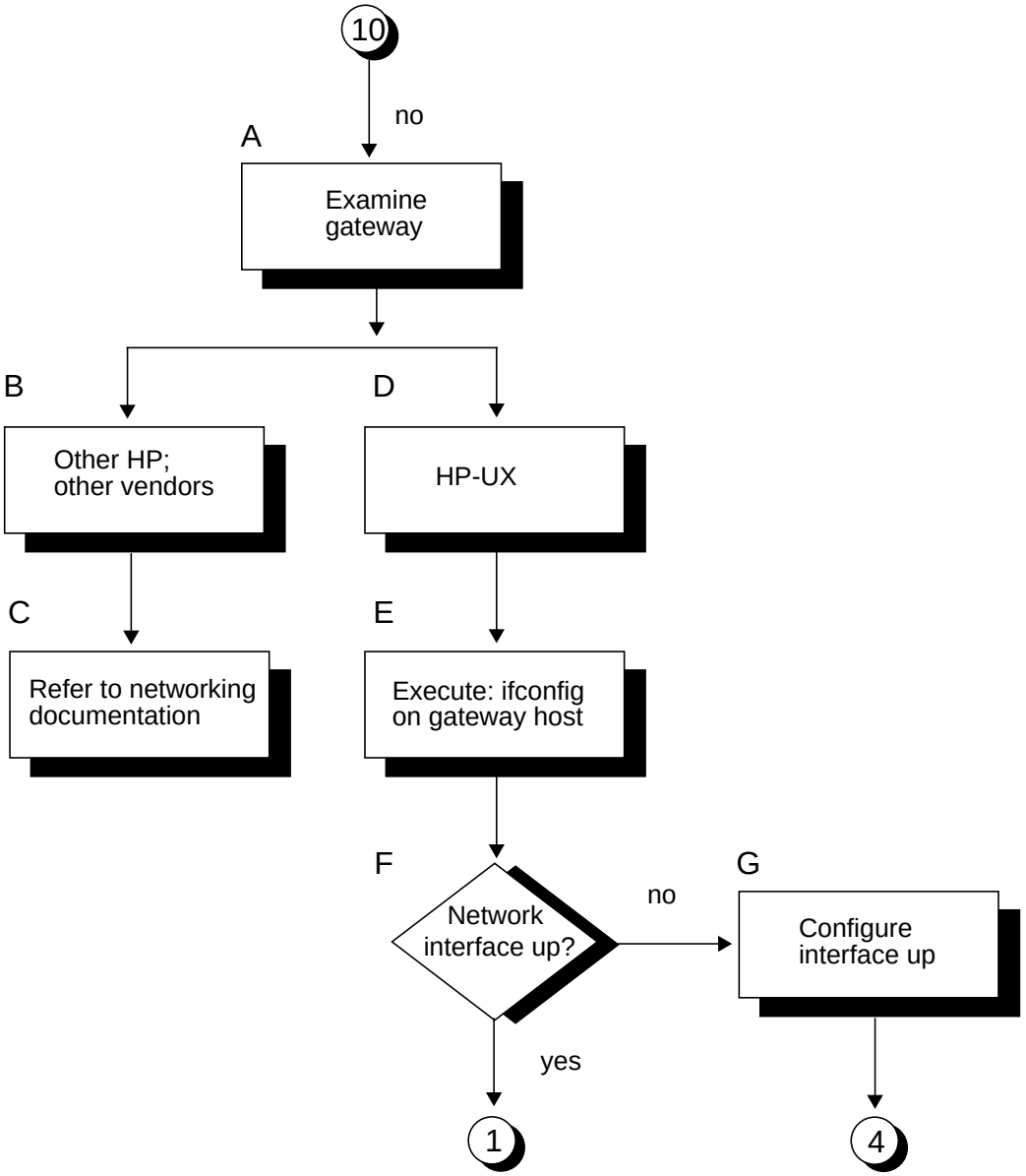
E. *Add route entry on local system*. Use the `route` command to add a route entry to the route table on the local system. Refer to the `route(1M)` man page for a complete description of the command.

F. *Routing table on remote OK?* Check that the routing information on the remote system is OK.

G. *Correct routing table.* If the routing information is incorrect, correct it using the `route` command.

Flowchart 10: Gateway Remote Loopback Test continued

Figure 4-12 Flowchart 10



Flowchart 10 Procedures

A. *Examine gateway.* If the gateway is an HP-UX Server, go to D. If it is not, go to B.

B. *Other HP; other vendors.* Go to C.

C. *Refer to networking documentation.* Refer to the documentation that came with the gateway for additional diagnostics.

D. *HP-UX.* Go to E.

E. *Execute: ifconfig on gateway host.* Execute `ifconfig` for all network interfaces on the gateway.

NOTE `Running` is always displayed by ifconfig output. It indicates only that there is OS support for the interface.

F. *Network interface up?* If the output from `ifconfig` does not include the UP parameter, the network interface is down. Execute `netstat -in` to check the status of the network interfaces. An asterisk (*) next to the interface indicates that the interface is down.

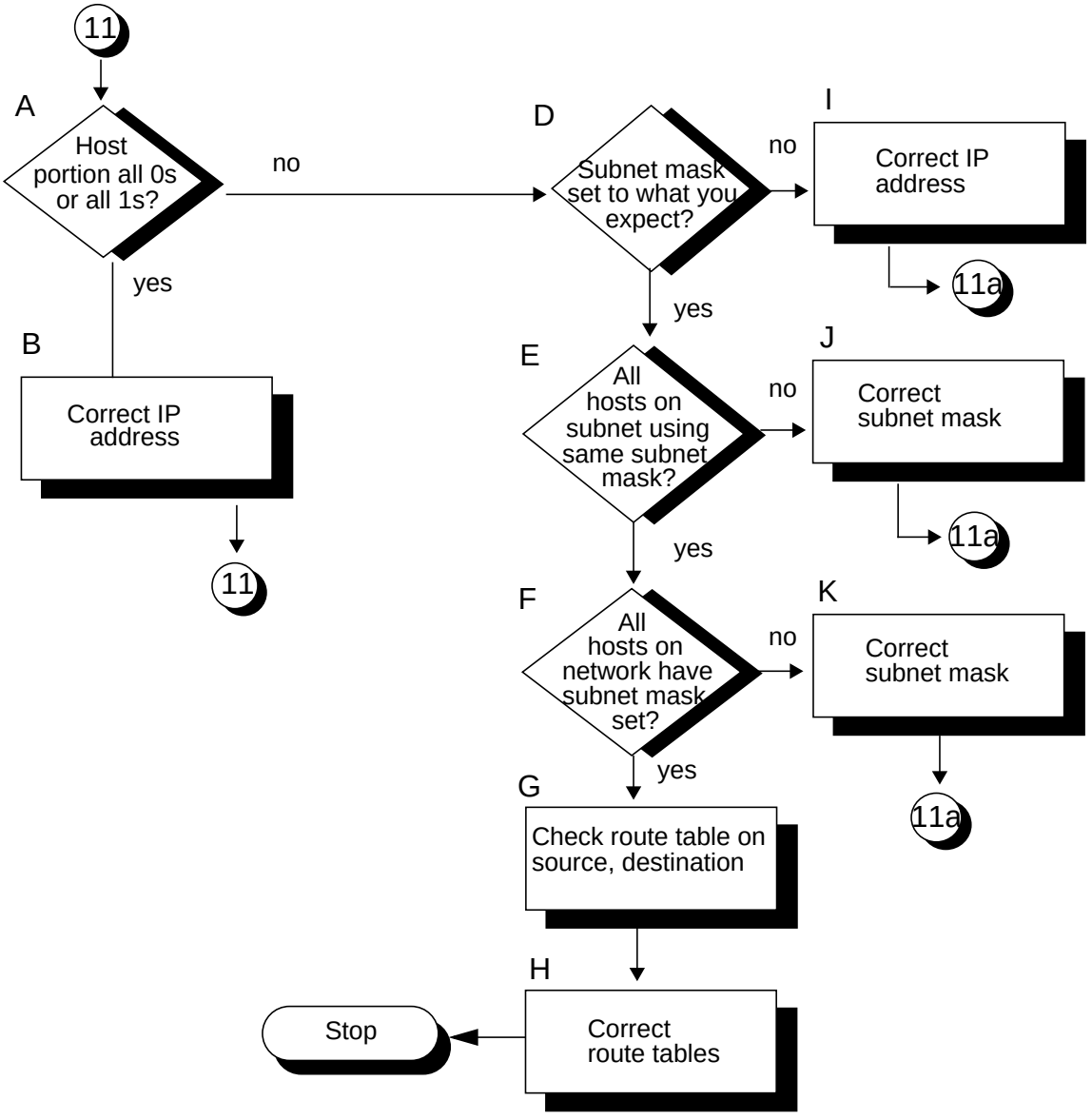
If the network interface is down, go to Flowchart 2. If the network interfaces are UP, start again with Flowchart 1. Using Flowchart 1, test all network interfaces on the gateway.

Use `netstat -in` to make sure that the configured encapsulation is correct.

G. *Configure interface up.* Execute `ifconfig` on each interface to bring it up. Start again with Flowchart 1. Using Flowchart 1, test all network interfaces on the gateway.

Flowchart 11: Subnet Test

Figure 4-13 Flowchart 11



Flowchart 11 Procedures

A. *Host Portion all 0's or all 1's?* Execute `ifconfig`. Is the host portion of the IP address all 0's or all 1's? These values are reserved. Refer to chapter 6 for details on subnets. If the host portion of the IP address is all 0's or all 1's, go to B to correct the IP address. Otherwise, go to C to examine the subnetwork number.

B. *Correct IP address.* Correct the IP address and start again with Flowchart 11.

D. *Subnet mask set to what you expect?* Check your network map and execute `ifconfig` to determine the subnet mask for your node. Refer to Chapter 6 for details on subnets. If the subnet mask is not what you expect, go to I. Otherwise, go to E.

E. *All hosts on subnet using same subnet mask?* Execute `ifconfig` for every network interface on each node on the entire network. If all nodes are using the same subnet mask, go to F. Otherwise, go to J to correct the subnet masks.

F. *All hosts on network have subnet mask set?* Execute `ifconfig` for every network interface on each node on the entire network. If all nodes have the same subnet mask set, go to G. Otherwise, go to K to set the correct subnet masks.

G. *Check route table on source, destination.* Execute `netstat -rv` on the two hosts. Go to H.

H. *Correct the route tables (if necessary).* In general, specify a network, not a host when adding to the route table. Specifying a network as the destination enables you to add nodes to the remote destination subnetwork without updating the route tables on the local subnetwork every time you add a node to the remote subnetwork.

I. *Correct IP address.* Set the subnet mask to the proper value. Go to D.

J. *Correct subnet mask.* To do so, execute `ifconfig` with the proper subnet mask. Go to D.

K. *Correct subnet mask.* To do so, execute `ifconfig` with the proper subnet mask. Go to D.

5 LAN Resources

In addition to this manual, use the following resources to maintain and administer HP HP-UX LAN software.

HP-UX Man Pages

While installing, configuring, or troubleshooting HP-UX LAN, you may need to refer to any of the following online man pages (manual reference pages) for useful HP-UX operating system or HP-UX LAN commands. To display a man page, type the following at the system prompt:

```
man <command name>
```

- `arp(1M)` displays and modifies the Internet-to-Ethernet and Internet-to-Fibre Channel address translation tables used by the Address Resolution Protocol.
- `hosts(4)` contains database that contains a single line entry for each host name entry.
- `ifalias(1M)` configures network interface to support multiple IP addresses at Network Layer.
- `ifconfig(1M)` assigns an address to a network interface and configures parameters, such as the netmask, broadcast address, and trailer support.
- `ioscan(1M)` scans system hardware, usable I/O system devices, or kernel I/O system data structures as appropriate, and lists the results.
- `lanadmin(1M)` resets or reports status of the LAN card.
- `lanconfig(1M)` configures network interface parameters at Data-Link Layer.
- `lanscan(1M)` displays information about LAN adapters that are successfully bound to the system.
- `linkloop(1M)` verifies network connectivity through the Data Link Layer.
- `ndd(1M)` displays and modifies network driver parameters.
- `netfmt(1M)` formats the `nettl` tracing and logging binary files.
- `netstat(1M)` provides network statistics and information about network connections.
- `nettladm(1M)` captures and controls network tracing and logging information.
- `nettl(1M)` controls network tracing and logging.
- `ping (1M)` verifies network connectivity through the Network Layer and reports round-trip time of communications between the local and remote hosts.
- `route(1M)` adds and deletes entries to the network routing table, allowing your system to communicate through a gateway.
- `swinstall(1M)` loads software filesets onto 10.0 systems.
- `traceroute(1M)` traces the network path between two points at Network Layer.

Logging and Tracing Messages

HP-UX LAN comes with an online message catalog that reports LAN problems, probable causes, and actions for you to take to correct the problems. Messages are sent either to the system console or log files based on the logging and tracing configuration. To view the probable cause and action text of the message, use the `netfmt` command with the `-v` option. Below is an example.

```
# netfmt -v -f /var/adm/nettl.LOG00

**********************LAN/9000 NETWORKING**********************
Timestamp           : Tue Apr 04 PDT 1995 10:15:13.290279
Process ID          : [ICS]         Subsystem   : NS_LS_DRIVER
User ID ( UID )     : -1            Log Class   : ERROR
Device ID           : 0             Path ID     : 0
Connection ID       : 0             Log Instance: 0
Location            : 02004
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Network NS_LS_DRIVER Error 2004, pid [ICS]
82596 External loopback test failed.

The loopback test through the external MAU Failed

Check the LAN cable to make sure its properly connected to the
LAN card.Use the lanscan (1M) command to find the NMID of the
LAN card and reset this card. If the reset does not solve the
problem, reboot.

If rebooting is ineffective, notify your HP representative.
```

HP-UX LAN uses the `nettl` logging and tracing facility supplied with HP-UX to capture, control, and format messages produced by LAN. See the `nettl(1M)` man page for information on using the graphical user interface or command line interface. The graphical interface, which is accessed by running the `nettladm` command, includes the following features:

- screens which take you through any logging or tracing task desired
- screens which allow you to create and format reports
- the capability of logging and tracing subsystem-specific information
- main screens which are updated instantaneously with current logging and tracing information by subsystem
- context-sensitive on-line help

Contacting Your HP Representative

If you do not have a service contract with HP, you may follow the procedure described below, but you will be billed accordingly for time and materials.

If you have a service contract with HP, document the problem as a Service Request (SR) and forward it to your HP representative. Include the following information where applicable:

- A characterization of the problem. Describe the events and symptoms leading up to the problem. Attempt to describe the source of the problem.

  Your characterization should include: HP-UX commands; communication subsystem commands; functionality of user programs; result codes and messages; and data that can reproduce the problem.

- Obtain the version, update, and fix information for all software. To check your Internet Services or HP-UX LAN version, execute the command:

  ```
  what /stand/vmunix
  ```

  To check the version of your kernel, execute the command:

  ```
  uname -a
  ```

  This allows HP to determine if the problem is already known, and if the correct software is installed at your site.

- Illustrate as clearly as possible the context of any message(s). Record all error messages and numbers that appear at the user terminal and the system console.

- Prepare a listing of the HP-UX I/O configuration you are using for your HP representative to further analyze.

- Try to determine the general area within the software where you think the problem exists. Refer to the appropriate reference manual and follow the guidelines on gathering information for that product.

- Document your interim, or “workaround,” solution. The cause of the problem can sometimes be found by comparing the circumstances in which it occurs with the circumstances in which it does not occur.

- Create copies of any Internet Services or HP-UX LAN link trace files that were active when the problem occurred for your HP representative to further analyze.

- In the even of a system failure, obtain a full memory dump. If the directory `/var/adm/crash` exists, the HP-UX utility `/usr/sbin/savecore` automatically executes during reboot to save the memory dump. Hewlett-Packard recommends that you create the `/var/adm/crash` directory after successfully installing this product. Send the output of your system failure memory dump to your HP representative.
- Prepare copies of the `/etc/hosts` and `/etc/rc.config.d/netconf` files as well as any driver specific configuration files that apply.
- Run the verification command, `/usr/sbin/swverify`, and record the output.
- Execute the `display` command of the `lanadmin` diagnostic on the LAN interface and record the output.
- Record the troubleshooting flowchart number and step number where you are unable to resolve the problem.
- Save all network log files. Make sure that ERROR and DISASTER log classes are enabled when log files are collected.
- Execute the following commands and record the output:

```
netstat -s >> /tmp/filename
netstat -in >>/tmp/filename
netstat -rvn >>/tmp/filename
ndd -get /dev/tcp tcp_status >>/tmp/filename
ndd -get /dev/ip ip_ill_status >>/tmp/filename
ndd -get /dev/ip ip_ipif_status >>/tmp/filename
ndd -get /dev/ip ip_ire_status >>/tmp/filename
ndd -get /dev/ip ip_ill_config_status >> /tmp/filename
ndd -get /dev/arp arp_cache_report >> /tmp/filename
arp -an >>/tmp/filename
lanscan >>/tmp/filename
what /stand/vmunix >>/tmp/filename
uname -a >>/tmp/filename
```

Prepare the formatted output and a copy of the log file for your HP representative to further analyze.

6 Network Addressing

This chapter introduces network addressing concepts. It contains the following sections:

- Overview of Network Addressing Schemes

- Networking Terminology
- Network Addresses and Node Names
- Internet Addresses
- Subnet Addresses
- Configuring Gateways on Fixed-Length Subnets
- Variable-Length Subnet Addressing
- Configuring Gateways on Variable-Length Subnets
- Configuring Gateways on Supernets
- IP Multicast Addresses
- Virtual IP (VIP) Addresses
- CIDR - Classless Inter-Domain Routing

Overview of Network Addressing Schemes

On the HP-UX 11i v2 Release, Hewlett-Packard offers several types of addressing schemes. Table 6-1 below shows the advantages and disadvantages of each type of scheme.

Table 6-1 Comparison of Subnet and Supernet Addressing Schemes

Address Type	Advantages	Disadvantages
Fixed-Length Subnet Addressing	Simplicity - same netmask across network - same size subnets across the network	Inefficiency & Inflexibility - waste of address space - same size subnets across the network - cannot grow subnet beyond the fixed size
Variable-Length Subnet Addressing	Efficiency & Flexibility - address space allocated according to projected size of subnets - variability in subnet size - expandability in subnet size - grow subnet by changing subnet mask only	Complexity - keeping track of subnet ranges - keeping track of netmasks
Supernet Addressing	Simplicity - same netmask across subnets - no gateway configuration for networks	Network Impact - performance. Network bandwidth is shared by all nodes in the supernet. - requires bridges if the supernet is spread across multiple physical networks

NOTE If you are already using the fixed-length addressing scheme and do not need extra addressing space, then it is recommended that you not convert your network to one of the new addressing schemes.

Refer to the “Subnet Addresses” subsection for information on fixed-length and variable-length addressing.

Networking Terminology

Following are descriptions of important networking terms.

Nodes

A **node** is a computer on the network. **Local node** (or host) refers to the computer or host to which your terminal is physically attached. A **remote node** is a computer on the network with which your local node can communicate. A remote node does not have to be directly attached to your terminal.

Routes and Protocols

A **route** is the sequence of network nodes through which messages travel when sent from a source node to a destination node.

A **protocol** is a set of rules for a particular communication task. A protocol handler or protocol module is a piece of software that implements a particular protocol.

Network Interface Name

A **network interface** is a communication path through which messages can be sent and received. A hardware network interface has a hardware device associated with it, such as an Ethernet, Fibre Channel, ATM, Token Ring, or FDDI card. A software network interface does not include a hardware device, (for example, the **loopback** interface). An IP address is associated with an interface name. The interface name(s) for a hardware network interface can be found by running the `lanscan` command and looking at the “Net-Interface Name PPA” field.

For Ethernet cards, you can choose either Ethernet encapsulation by specifying `lan` when configuring the interface, or IEEE 802.3 encapsulation by specifying `snap` when configuring the interface.

The interface name may include a colon (:), followed by a number that denotes the logical interface number. The number 0 is the first logical interface number for a card/encapsulation type and is known as the initial interface. The interface name lan0 is equivalent to lan0:0, lan1 is equivalent to lan1:0, and so on.

You must configure the initial interface for a card/encapsulation type before you can configure subsequent interfaces for the same card/encapsulation type. For example, you must configure lan0:0 (or lan0) before you configure lan0:1 and lan0:2. Once you have configured the initial interface, you can configure the subsequent interfaces in any order. Note that the IP

addresses assigned to a card may be on the same subnet or on different subnets. See the section “Interfaces,” in Chapter 7 for more information about logical interfaces and interface names.

An initial interface cannot be removed from the system until all subsequent logical interfaces are removed. You can remove subsequent interfaces from the system with the `ifconfig` command, as in the following example:

```
ifconfig lan1:1 inet 0.0.0.0
```

The initial interface (for example, lan1) can then be removed from the system with the `ifconfig` command, as in the following example:

```
ifconfig lan1 unplumb
```

A loopback interface does not have a hardware device associated with it. The name of this type of interface is `lo0`, denoting the loopback interface. A loopback interface is automatically created by the TCP/IP stack even if the system is not connected to a network.

Gateway

A **gateway** is a device used to connect two or more networks. The gateway serves to route information among the networks. An HP-UX Server with two or more LAN cards installed may act as a LAN-to-LAN gateway. Such a node may also be referred to as a LAN-to-LAN router or IP router. If a node is a gateway, it affects how you configure and maintain LAN software. Refer to node D in the network maps in Figure 6-7 and Figure 6-13 for examples of gateways. A gateway system has to have at least two network interfaces configured, one for each network to which it belongs. A gateway can be either a router or a system.

Routing Table

Each node on the LAN has a routing table. A **routing table** contains information about the route to nodes on other LANs. The connections to other LANs are made through gateways.

ARP Cache

Each interface card on a system is identified by an IP address and a station address. The **ARP cache** contains the IP address of a remote interface and the station address which is used to send packets to that IP address. If the remote system is not on the same physical network, the station address in the ARP cache is for an interface on a gateway.

Usually, an ARP cache entry is automatically created when the system needs to send the first packet to a remote IP address. ARP cache entries are usually deleted automatically when they have not been used for a period of time.

Network Addresses and Node Names

Several types of names and addresses are used in networking software. This can be confusing to first-time users. Table 6-2 illustrates which address type is used by each layer of the OSI model. A description of each address type and how it is used by LAN and the services which run on it follows inTable 6-3. Refer to “Network and System Names” on page 28 for additional information on how these names are assigned.

Table 6-2 Network Addresses and the OSI Model

	OSI Layer Name	OSI Layer Function	Address Type Used
7	Application	network programs	hostname
6	Presentation	data interpretation	hostname
5	Session	connection control	socket address
4	Transport	end-to-end transfer	port address
3	Network	routing and switching	Internet (IP) address
2	Data Link	data packaging and error detection	link level address
1	Physical	physical connection	link level address

Table 6-3 Network Address Types, Descriptions, and Examples

Address Type	Description	Recorded In	Used By
link level address	Also referred to as **MAC address** or **station address**. A link level address is the unique address of the LAN interface card. This value can be configured in SAM and `lanadmin`. An example of a link level address in hexadecimal: 0800090012AB.	Interface card; can be changed in `/etc/rc.config.d/hpbtlanconf` file. Refer toTable 3-2 for sample filenames.	`linkloop` diagnostic; internals of networking software to uniquely identify nodes on the LAN; displayed by `lanadmin` and `lanscan` diagnostics.
internet (IP) address	Also referred to as **IP address**. An internet address is the network address of a computer node. This address identifies which network the host is on (see network address description below) and which host it is (see host address description below). An example of an internet address: 192.6.23.3	`/etc/hosts`; `/etc/rc.config.d/netconf`.	Typically used in combination with the network address to form an internet address. HP-UX `ifconfig` command.

Table 6-3 (Continued) (Continued)Network Address Types, Descriptions,

Address Type	Description	Recorded In	Used By
network address	Also, **network number**. The network address is the network portion of an internet address that represents the local network on which a host exists. The network address is the same for all nodes on that network. Refer to “Internet Addresses” in this chapter for a definition of Internet address classes. If the IP address is 192.6.23.3 (Class C), and the subnet address is 255.255.255.0, then the network address portion is 192.6.23.	`/etc/networks`. Combined with host address in `/etc/hosts`.	Routing facility. Displayed by: `netstat -in`, `netstat -rn`, and `netstat -rnv`. HP-UX `ifconfig` command.
host address	Also, **host number**. The host address is that portion of the internet address that is unique to the network. The host address identifies a particular node on the network. Refer to “Internet Addresses” in this chapter for a definition of Internet address classes. If the IP address is 192.6.23.3 (Class C), and subnet address is 255.255.255.0, then the host address portion is 3.	Combined with network address in `/etc/hosts`.	Typically used in combination with the network address to form an internet address. HP-UX `ifconfig` command.

Table 6-3 (Continued) (Continued)Network Address Types, Descriptions,

Address Type	Description	Recorded In	Used By
port address	Also referred to as **TCP port number**, **UDP port number**, or simply **port**. A port address is an address within a host that is used to differentiate between multiple communication endpoints with the same internet address and protocol. A port address is associated with a particular service. Well known port numbers are defined by RFC 923, "Assigned Numbers." For example, if your local address is listed as 192.6.23.3.1023, then .1023 is the port address.	`/etc/services`.	Service requests. Displayed by `netstat -an`.
socket address	A **socket address** is declared in processes defined by the interprocess communication software. Refer to *Using Internet Services* for more information on interprocess communication. Refer to the `sockaddr` struct in the *BSD Sockets Interface Programmer's Guide* for examples.	Socket address variables.	Interprocess communication.

Table 6-3 (Continued) (Continued)Network Address Types, Descriptions,

Address Type	Description	Recorded In	Used By
system name	Also referred to as the **system host name** and **system node name**. This is the name your HP-UX system is known by and is assigned using the HP-UX `uname` command. An example of a system name is: `host3`. Assigned automatically by the system.	`/etc/rc.config.d/netconf` (as HOSTNAME variable).	`uucp` facilities.
host name node name	Also known as the **Internet host name** and **NFS host name**. A symbolic name associated with an internet address by which a node can be uniquely identified. An example of a host name is: `host3`. Assigned by using the `hostname` command.	`/etc/hosts` ; `/etc/hosts.equiv` (optional); `$HOME/.rhosts` (optional) ; `$HOME/.netrc` (optional) ; `/usr/adm/inetd.sec` (optional) .	All Internet Services.

Internet Addresses

Internet addresses are used extensively by LAN and WAN products as well as Internet Services.

An internet address (often referred to as the IP address) consists of two parts:

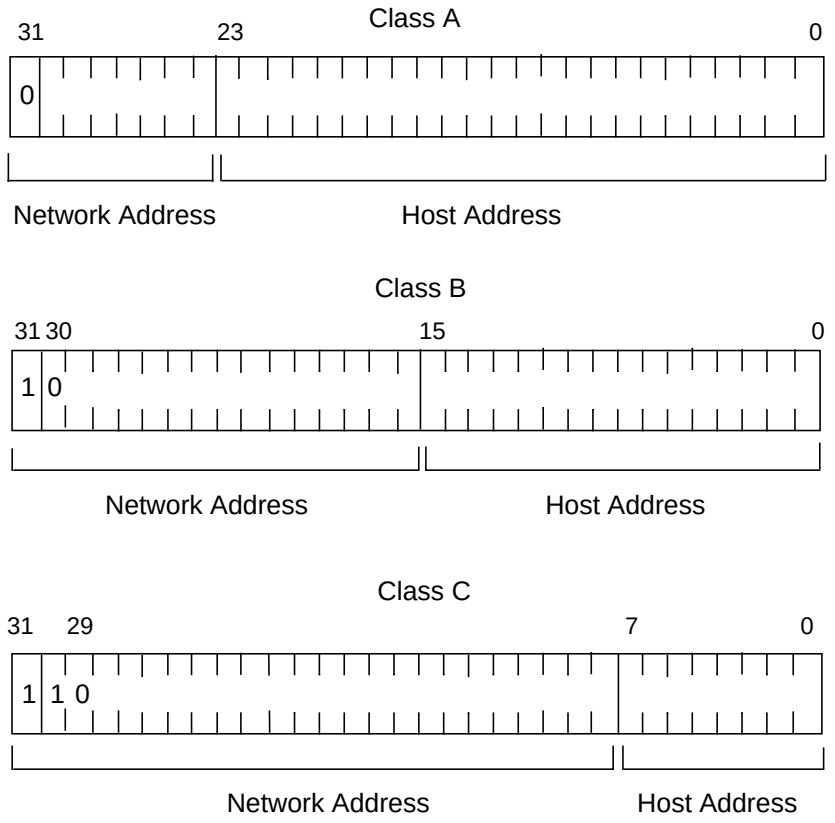
- Network address.
- Host address.

The network address identifies the network. The host address identifies a node within the network. A network address is concatenated with a host address to form the internet address and to uniquely identify a node within a network.

Internet Address Formats

There are four internet address classes, each accommodating a different number of network and host addresses. The address classes are defined by the most significant bits of the binary form of the address as shown in Figure 6-1. This section discusses three of the classes (Class A, Class B, Class C). The fourth class (Class D) is discussed in the section “IP Multicast Addresses.”

Figure 6-1 Internet Address Classes



The address classes can also be broken down by address ranges. Internet addresses are typically represented by converting the bits to decimal values an octet (8 bits) at a time, and separating each octet's decimal value by a period (.). Therefore, internet addresses are typically of the following form:

```
n.n.n.n
```

where n is a number from 0 to 255, inclusive. This form is referred to as decimal dot notation or dot notation.

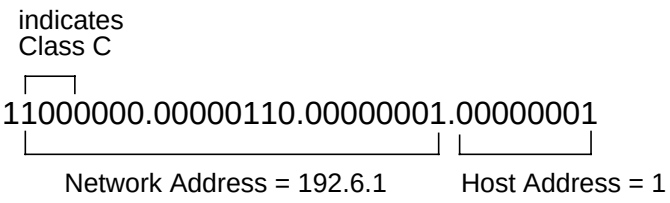
Table 6-4 lists the number of networks and nodes and the address ranges for Class A, Class B, and Class C networks. Class D networks are described later in this chapter in "IP Multicast Addresses."

Table 6-4 Internet Address Classes

Class	Networks	Nodes per Network	Address Range
A	127	16777215	1.0.0.1 – 126.255.255.254
B	16383	65535	128.1.0.1 – 191.255.255.254
C	2097151	255	192.0.1.1 – 223.255.255.254
Reserved	-	-	224.0.0.0 – 255.255.255.255

To determine a network address and host address from an internet address, you must separate the network and host address fields. For example, the bit representation of internet address 192.6.1.1 is separated as follows:

Figure 6-2 Bit Representation of Internet Address



Assigning an Internet Address

Each node on the network has at least one internet address. When assigning internet addresses, you must determine network addresses and host addresses as described in this section.

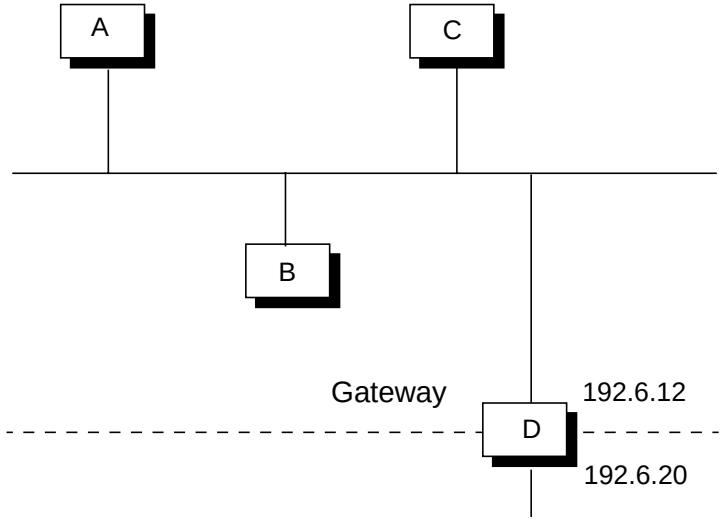
NOTE When specifying internet addresses, do not use leading zeroes within address fields. For example: 192.006.012.023 is incorrect; 192.6.12.23 is correct.

Assigning Network Addresses

To assign network addresses, follow these rules:

- You must have a network address for each logical network.
- If your system is attached to more than one physical network via a gateway, the network addresses of these interfaces may not be the same. Refer to Table 6-3 below for a gateway example.
- All nodes in the same network, however, must have the same network address.
- Do not assign the network addresses 0 or 255 (Class A), 0.0 or 255.255 (Class B), or 0.0.0 or 255.255.255 (Class C) to any network. *Those addresses are reserved.*
- Do not assign Class A network address 127. This address is reserved for the loopback interface.

Figure 6-3 Assigning Network Addresses



Assigning Host Addresses

Host addresses must be unique within each network. You can assign host addresses according to your own needs, but they must be within the range for the internet address class that you are using.

NOTE Do *not* assign the host addresses 0.0.0 or 255.255.255 (Class A), 0.0 or 255.255 (Class B), or 0 or 255 (Class C) to any nodes; *these addresses are reserved*.

IP Address for Loopback Interface (lo0)

The loopback interface (`lo0`) is automatically configured when the system boots with the TCP/IP software. The loopback interface is really a “pseudo-device,” since there is no hardware card associated with it. This interface is created solely to facilitate sending IP datagrams to the IP loopback address 127.0.0.1.

The default IP address and netmask of the loopback interface are 127.0.0.1 and 255.0.0.0, respectively. By default, the loopback interface sets up routing entries so that packets for any 127.*.*.* address will loop back to the local host. Any attempt to change the address of the initial loopback interface (lo0:0) will fail.

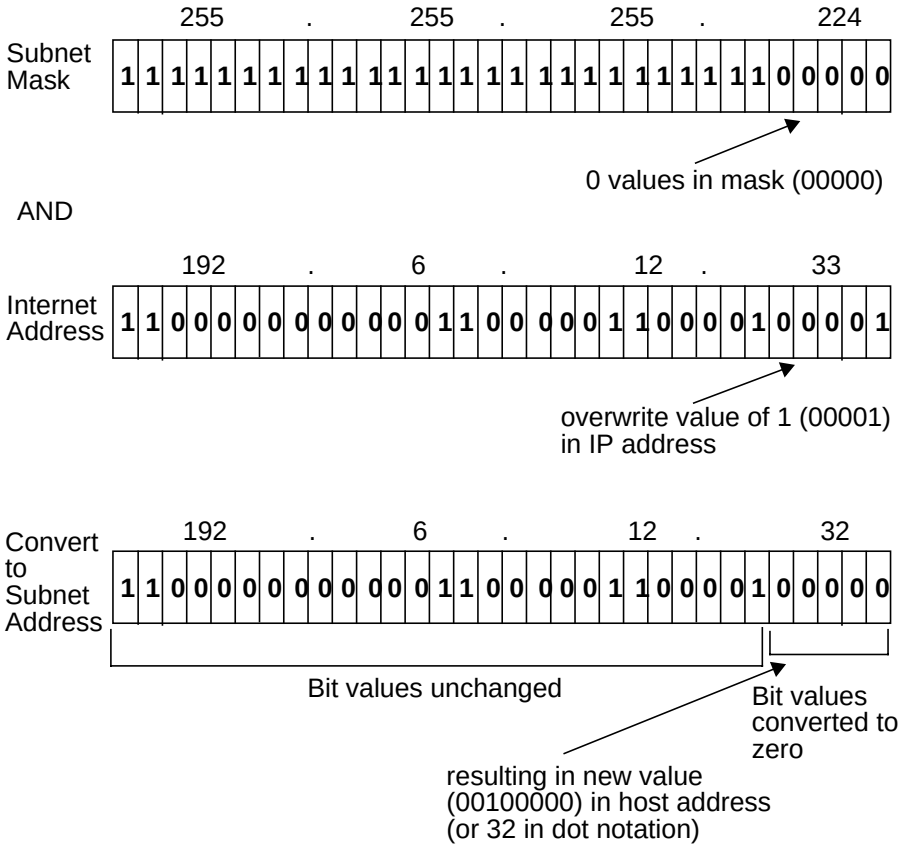
Subnet Addresses

Subnetting is an optional addressing scheme that allows you to partition the host address portion of an internet address into discrete subnetworks. This allows you to have multiple physical networks without requiring you to obtain multiple network addresses. The physical networks are connected via gateways.

For example, if you have a large installation with many interconnected nodes, you could run into hardware configuration restrictions or performance degradation if you tried to place all nodes on the same physical network. With subnetting you can install several smaller physical networks but have them all share the same network address. When messages with subnet addresses are routed across the network, the internet address is ANDed with the subnet mask to determine the subnetwork address. (0 values in the subnet mask convert corresponding bits in the IP address to 0.)

Figure 6-4 shows how a Class C Internet address and a subnet mask combine to form a subnet address.

Figure 6-4 Internet Address 192.6.12.33 ANDed with Subnet Mask 255.255.255.224



When the internet address is ANDed with the subnet mask, the zero values in the host portion of the subnet mask will “overwrite” the corresponding bits of the host portion of the internet address and the resultant subnet address will be 192.6.12.32 as shown in Figure 6-4 above. Non-zero values in the subnet mask indicate that the corresponding bits in the internet address do not change. The subnet mask may be fixed-length or variable-length.

IP Subnet Mask

The subnet field (the portion of an IP address that identifies the subnet beyond the network portion of the address) can be all 0's or all 1's, as described in RFC 1878.To disallow subnet fields with all ones or all zeroes (conform to RFC 1122 behavior), set the ndd parameter `ip_check_subnet_addr` to `1` in the `nddconf` file (`/etc/rc.config.d/nddconf`).

Selecting a Subnet Addressing Scheme

In addition to fixed-length subnet addressing, HP-UX11i v2 systems support variable-length subnet addressing. The advantages of using variable-length subnet addressing over fixed-length subnet addressing include the following:

- Allows the local administrator to easily increase/decrease the size of individual subnets when variable-length subnet addressing is used in conjunction with a non-contiguous numbering system (and/or ranges of numbers for each subnet that are non-contiguous).

  The mirror-image counting feature, which will be discussed in the following section, also allows for more possibilities in the numbers that can be assigned to individual subnets.

- Reduces the amount of rework that network planners will have to do on network design after the initial plan has been completed.

  Fixed-length subnets were easy to implement, but growth restrictions often meant that it was necessary to invest more time whenever a change was made to a subnet after it had been originally designed.

This new feature also makes it possible to have more than one subnet mask on a network.

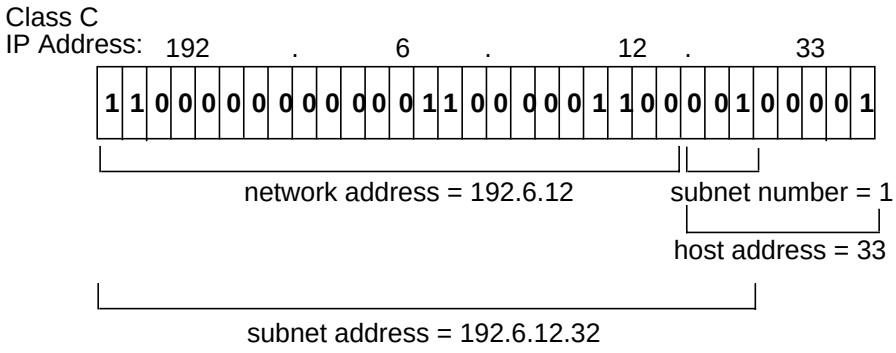
As described previously, an internet address can be represented as four fields separated by a period, each of which represents 8 bits of the overall address.

The subnet address is based on the host address portion of the internet address. The host address portion subdivides into subnet number and host number fields to accommodate a given number of subnetworks and a given number of nodes per subnetwork. The size of the subnet number field is determined by the subnet mask, which is explained later in this section. The subnet number field must contain a minimum of one bit.

In the example below, the IP address, 192.6.12.33, has a subnet number of 1.

Refer to Figure 6-5 earlier in this chapter for an illustration of how the subnet mask is ANDed with the IP address to form the subnet number.

Figure 6-5 Subnet Address and Subnet Number of Class C Internet Address 192.6.12.33



The following rules apply when choosing a subnet addressing scheme and an internet address:

- All subnets on the same network must have the same network address.
- If your system is attached to more than one physical network, the subnet addresses of the interfaces on your system cannot be the same.
- Do not assign a host address where all the bits of the host number are 0 or all the bits are 1.

You may choose a fixed-length subnet addressing scheme in which one subnet mask will be used in all subnets in your network. You may also choose a variable-length subnet addressing scheme in which the subnet masks may be varied depending on the size of the subnets you want.

If you wish to implement variable-length subnetting, first read the conceptual information, “Assigning Variable-Length Subnet Addresses” later in this chapter. Then refer to the specific task-oriented instructions in the “Manually Installing and Configuring HP-UX LAN” chapter earlier in this manual. For additional information, refer to the following online man pages: `ifconfig(1M)`, `netstat(1)`, `ppl.remotes(4)`, `routing(7)`, and `route(1M)`. Detailed descriptions of fixed-length and variable-length subnet addressing follow.

Fixed-Length Subnet Addressing

This section shows you how to apply a fixed-length subnet addressing scheme on a class C network. The same steps may be applied on a Class A or Class B network.

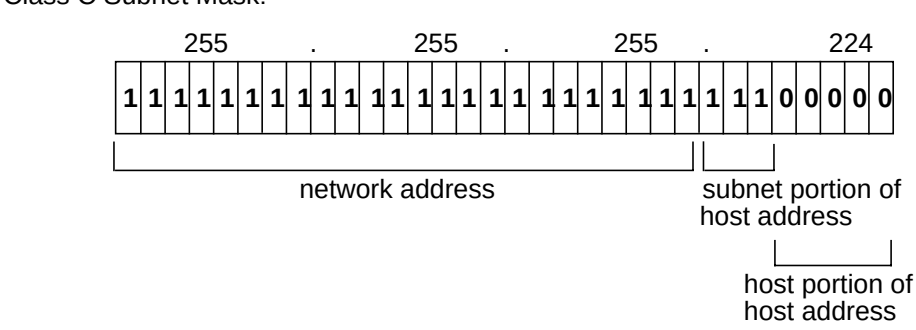
Assigning a Fixed-Length Subnet Mask

Subnet addressing is implemented by specifying a 32-bit subnet mask in the `ifconfig` command or the `netconf` file when a LAN interface card is assigned an internet address. All nodes on a subnet must have the same subnet mask. Under the fixed-length subnet address scheme, all subnets on a network, must have the same subnet mask.

The subnet mask is ANDed with the address attached to a message coming across the network to determine if that message should be routed to a node on the local network or ignored.

Before a subnet mask is applied, the first three octets of a Class C IP address represent the network address. If the subnet mask 255.255.255.224 is applied, the first three octets of the address and the first three bits of the last octet indicate the subnet address while the last five bits of the last octet indicate the host address. This is shown in the following figure.

Figure 6-6 Fixed-Length Subnet Mask Example



With this subnet mask, up to 8 subnets with 30 nodes per subnet are possible. Table 6-5 below lists the valid internet address ranges for nodes in a network with the subnet mask 255.255.255.224.

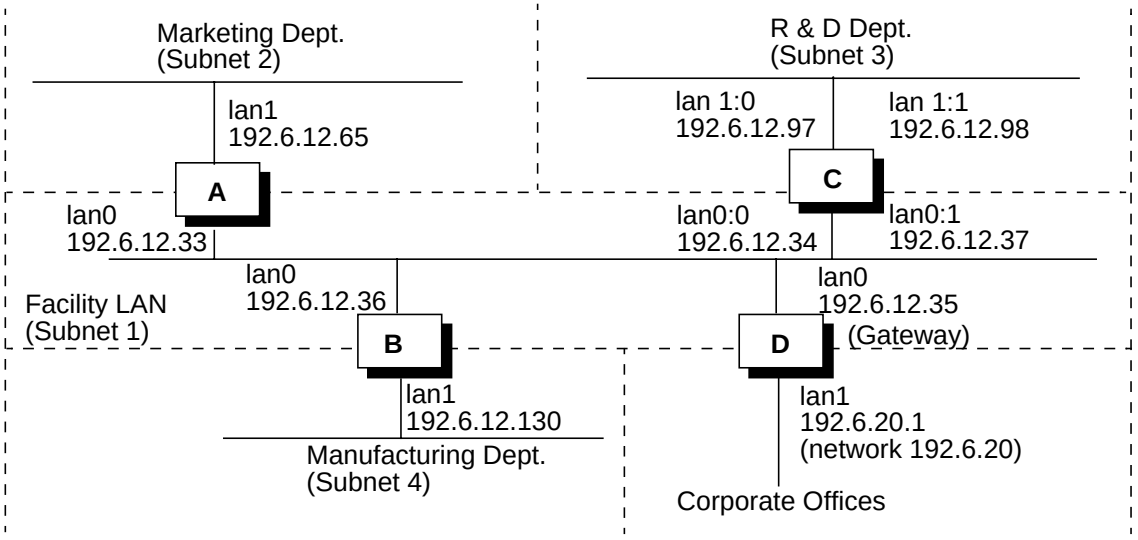
Table 6-5 Fixed-Length Subnet Addressing (Subnet Mask 255.255.255.224)

Subnet Address (dot notation)	Internet Address Range (dot notation)	Subnet Broadcast Address
n.n.n.0	n.n.n.1 - n.n.n.30	n.n.n.31
n.n.n.32	n.n.n.33 - n.n.n.62	n.n.n.63
n.n.n.64	n.n.n.65 - n.n.n.94	n.n.n.95
n.n.n.96	n.n.n.97 - n.n.n.126	n.n.n.127
n.n.n.128	n.n.n.129 - n.n.n.158	n.n.n.159
n.n.n.160	n.n.n.161 - n.n.n.190	n.n.n.191
n.n.n.192	n.n.n.193 - n.n.n.222	n.n.n.223
n.n.n.224	n.n.n.225 - n.n.n.254	n.n.n.255

Example of Subnets with a Fixed-Length Subnet Mask

The following example shows four subnetworks within the 192.6.12 network along with the `netconf` entries necessary to configure these subnetworks. The complete network map is shown in Figure 6-8.

Figure 6-7 Network Map for Subnetting



Summary network information:

```
Company division network = 192.6.12
    Subnet mask: 255.255.255.224

Facility LAN subnet number = 1
    Host address range: 22 to 62
    Host A internet address: 192.6.12.33 for network interface lan0
    Host B internet address: 192.6.12.36 for network interface lan0
    Host C internet address: 192.6.12.34 for network interface lan0
    Host D internet address: 192.6.12.35 for network interface lan0

Marketing Department subnet number = 2
    Host address range: 65 to 94
    Host A internet address: 192.6.12.65 for network interface lan1

R & D Department subnet number = 3
    Host address range: 97 to 126
    Host A internet address: 192.6.12.97 for network interface lan1:0
    Host B internet address: 192.6.12.98 for network interface lan1:1
```

```
Manufacturing Department subnet number = 4
    Host address range: 129 to 158
    Host B internet address: 192.6.12.130 for network interface lan1
```

Configuring Hosts on Fixed-Length Subnets Using the netconf file

There are two ways to configure a subnet mask:

- Use SAM to configure the subnet mask.
- Use the `ifconfig` command. These changes will disappear, however, when the system reboots.

To set a subnet mask, you may either include it in the SUBNET_MASK variable in the `netconf` file or the `ifconfig` command. The `netconf` file contains information you entered in SAM and this information is used to start networking when the system reboots. If you configure your network interfaces using SAM, SAM will enter the proper information in the `netconf` file for you. The examples below are `netconf` examples for the hosts in the example above after you have configured them in SAM:

Host A:

```
INTERFACE_NAME[0]="lan0"
IP_ADDRESS[0]="192.6.12.33"
SUBNET_MASK[0]="255.255.255.224"

INTERFACE_NAME[1]="lan1"
IP_ADDRESS[1]="192.6.12.65"
SUBNET_MASK[1]="255.255.255.224"
```

Host B:

```
INTERFACE_NAME[0]="lan0"
IP_ADDRESS[0]="192.6.12.36"
SUBNET_MASK[0]="255.255.255.224"

INTERFACE_NAME[1]="lan1"
IP_ADDRESS[1]="192.6.12.130"
SUBNET_MASK[1]="255.255.255.224"
```

Host C:

```
INTERFACE_NAME[0]="lan0"
IP_ADDRESS[0]="192.6.12.34"
SUBNET_MASK[0]="255.255.255.224"

INTERFACE_NAME[1]="lan0:1"
IP_ADDRESS[1]="192.6.12.37"
SUBNET_MASK[1]="255.255.255.224"

INTERFACE_NAME[2]="lan1"
IP_ADDRESS[2]="192.6.12.97"
SUBNET_MASK[2]="255.255.255.224"
```

```
INTERFACE_NAME[3]=”lan1:1”
IP_ADDRESS[3]=”192.6.12.98”
SUBNET_MASK[3]=”255.255.255.224”
```

Host D:

```
INTERFACE_NAME[0]=”lan0”
IP_ADDRESS[0]=”192.6.12.35”
SUBNET_MASK[0]=”255.255.255.224”

INTERFACE_NAME[1]=”lan1”
IP_ADDRESS[1]=”192.6.20.1”
SUBNET_MASK[1]= 255.255.255.0
```

In addition, every other host on each subnetwork would require the subnet mask 255.255.255.224 in their `netconf` file entries.

Example of Network Map with Fixed-Length Subnets

This sample network combines networks, subnets with a fixed-length mask, and clusters previously described and illustrated in this chapter along with a sample worksheet that provides configuration information necessary to attach these systems to the networks.

Figure 6-8 Network Map I

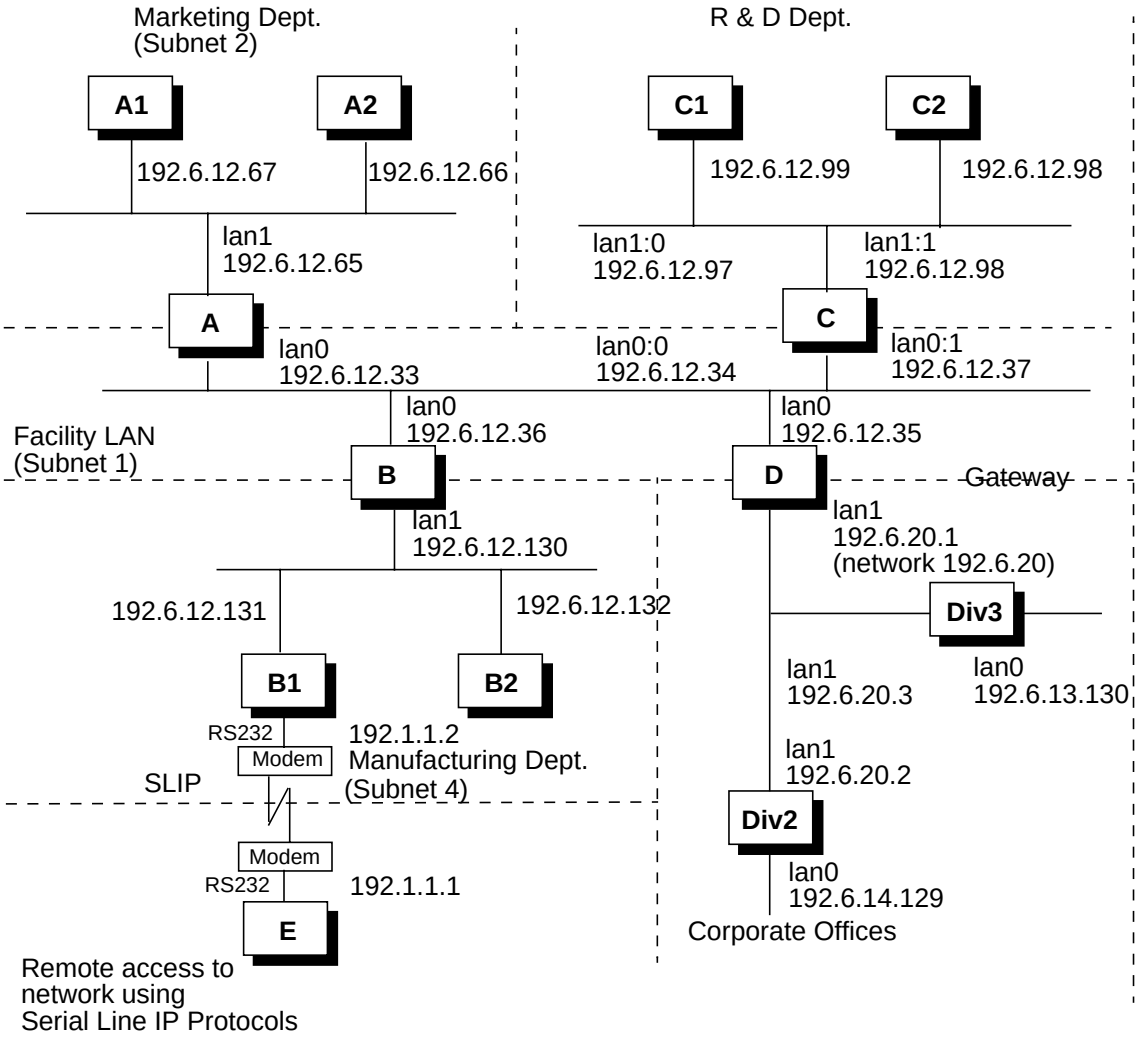


Table 6-6 Network Map I Worksheet

Host	Inter-face Alias	Internet Address	Station Address	Cnode Type
A	mkt_32 mkt_64	192.6.12.33 192.6.12.65	08000909030D 080009080102	Server
A1	mkt_a1	192.6.12.67	080009005201	Client 1
A2	mkt_a2	192.6.12.66	080009003001	Client 2
B	mfg_32 mfg_128	192.6.12.36 192.6.12.130	080009005201 080009000C24	N/A
B1	mfg__b1 b1_slip	192.6.12.131 192.1.1.2	080009001001 N/A	N/A
B2	mfg_b2	192.6.12.132	080009002125	N/A
C	rd_32 rd_36 rd_96 rd_99	192.6.12.34 192.6.12.37 192.6.12.97 192.6.12.98	080009267C14 080009267C14 080009260C85 080009260C85	Server
C1	rd_c1	192.6.12.99	08000900079C	Client 1
C2	rd_c2	192.6.12.98	08000900601A	Client 2
D	div1_32 div_gw	192.6.12.35 192.6.20.1	080009000740 080009000B30	N/A
Div2	div2_128 div2_gw	192.6.14.129 192.6.20.2	080009006041 080009007104	N/A
Div3	div3_128 div3_gw	192.6.13.130 192.6.20.3	080009004020 080009010312	N/A
E (SLIP)	e_slip	192.1.1.1	N/A	N/A

Subnet mask = 255.255.255.224

Configuring Gateways on Fixed-Length Subnets

Besides using the appropriate subnet masks, each gateway needs to be configured so that it can properly route messages among the several subnetworks. Following are descriptions of two types of routing: explicit routing and dynamic routing. When using explicit routing, you must specify the IP address of each gateway to which you are directly connected. When using dynamic routing, you need to specify only the IP address of one gateway, and the system learns the IP address of other gateways from the specified gateway.

Explicit Routing

There are many ways to set up routing. For example, you might use the route command or you may add the following entries to the `netconf` file on Host A in Figure 6-8:

```
ROUTE_DESTINATION[0]="net 192.6.12.128"
ROUTE_GATEWAY[0]="192.6.12.36"
ROUTE_COUNT[0]="1"

ROUTE_DESTINATION[1]="net 192.6.12.96"
ROUTE_GATEWAY[1]="192.6.12.34"
ROUTE_COUNT[1]="1"

ROUTE_DESTINATION[2]="net default"
ROUTE_GATEWAY[2]="192.6.12.35"
ROUTE_COUNT[2]="1"
```

The 1 in each ROUTE_COUNT entry specifies an indirect route. For example, messages for the system on the 192.6.12.128 subnetwork will first be sent to Host B (192.6.12.36), and from there they will be forwarded to the destination system.

Dynamic Routing

Alternatively, and perhaps the easiest way to manage growth on the 192.6.12 network, you might add the following entries to each `netconf` file.

Hosts A, B and C:

```
ROUTE_DESTINATION[0]="default"
ROUTE_GATEWAY[0]="192.6.12.35"
ROUTE_COUNT[0]="1"
```

Host D (Site gateway):

```
ROUTE_DESTINATION[0]="net 192.6.12.64"
ROUTE_GATEWAY[0]="192.6.12.33"
ROUTE_COUNT[0]="1"
```

```
ROUTE_DESTINATION[1]="net 192.6.12.128"
ROUTE_GATEWAY[1]="192.6.12.36"
ROUTE_COUNT[1]="1"

ROUTE_DESTINATION[2]="net 192.6.12.96"
ROUTE_GATEWAY[2]="192.6.12.34"
ROUTE_COUNT[2]="1"

ROUTE_DESTINATION[3]="default"
ROUTE_GATEWAY[3]="192.6.20.1"
ROUTE_COUNT[3]="0"
```

If you add a new subnetwork to the Facility LAN at a later time, you will need to add only an appropriate routing entry on Host D. It will not be necessary to configure the other subnet gateways A, B, and C.

With this configuration, each subnet gateway (Hosts A, B, and C) will initially route messages for a system outside its subnet to Host D. The subnet gateway, however, will learn of the more direct routes automatically when Host D redirects the messages to one of the other subnet gateways. Subsequent messages for the destination system will be routed directly to the appropriate subnet gateway.

For example, referring to Figure 6-8, suppose messages are sent from system A1 (192.6.12.67) to system B1 (192.6.12.131). The first message will actually be routed to Host D (through Host A). Host D then will redirect the message through Host B. At the same time, Host D will notify Host A that Host B is a more direct route for messages to system B1. Subsequent messages to system B1 will be routed directly from Host A to Host B.

Redirected routes are called dynamic routes. You can see these dynamic routes by executing the command `netstat -r` on Host A. Dynamic routes are indicated in the display by a D flag.

Proxy ARP Server

The default direct route entry on Host D assumes that there is a proxy ARP server on the 192.6.20 network. If one does not exist, additional indirect route entries can be configured for each gateway that is directly connected to the 192.6.20 network.

For example, referring to Figure 6-8, you might add the following indirect routes on Host D to send messages to Division 2 and Division 3.

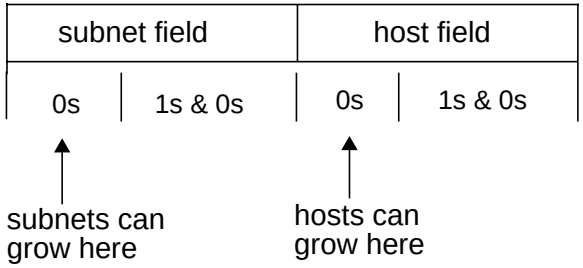
```
ROUTE_DESTINATION[4]="net 192.6.14"
ROUTE_GATEWAY[4]="192.6.20.2"
ROUTE_COUNT[4]="1"

ROUTE_DESTINATION[5]="net 192.6.13"
ROUTE_GATEWAY[5]="192.6.20.3"
ROUTE_COUNT[5]="1"
```

Variable-Length Subnet Addressing

For the most efficient use of address space and maximum flexibility in increasing/decreasing the size of your subnets, Hewlett-Packard recommends variable-length subnet addressing. To maximize the possibilities offered with this new approach, you should utilize mirror image counting, as described in this section, to select subnet numbers.

In the past, a network administrator typically assigned values to the subnet number and host address fields in numerical order. For example, within a given subnet, hosts were numbers 1, 2, 3, etc. and within a given network, the subnets were numbered 1, 2, 3, etc. The result was that some bits on the right side of the subnet field and host field were “ones and zeros” and some bits on the left side of the subnet and host fields were “all zeros” for all subnets and hosts. As shown below, the “all zeros” bits represented room for growth, and the “ones and zeros” bits represented bits already consumed by growth.

Figure 6-9 Traditional Subnet and Host Field Allocation



In the next example. the entire host field has been allocated. As a result, only the subnet field can grow.

Figure 6-10 Entire Host Field Allocated

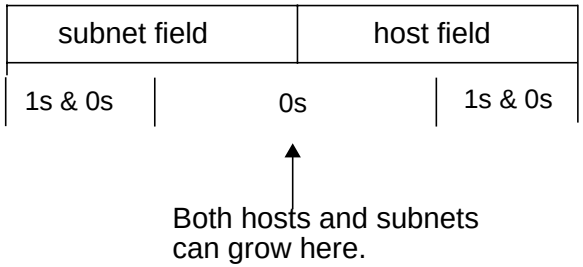
To alleviate this restrictive situation, when you use the variable-length subnetting approach, you can, alternatively, assign subnet numbers from the left of the subnet field and work right. This is implemented using mirror-image counting. In the mirror image approach, the bits for subnet numbers are assigned left to right instead of right to left (normal situation). This would result in the following format:

Table 6-7 Mirror Image Counting

Traditional Approach	Mirror Image Approach
01	10
10	01
011	110
100	001
101	101

This will allow for more growth bits between the subnet field and the host fields as shown below.

Figure 6-11 Mirror Image Subnet and Host Field Allocation



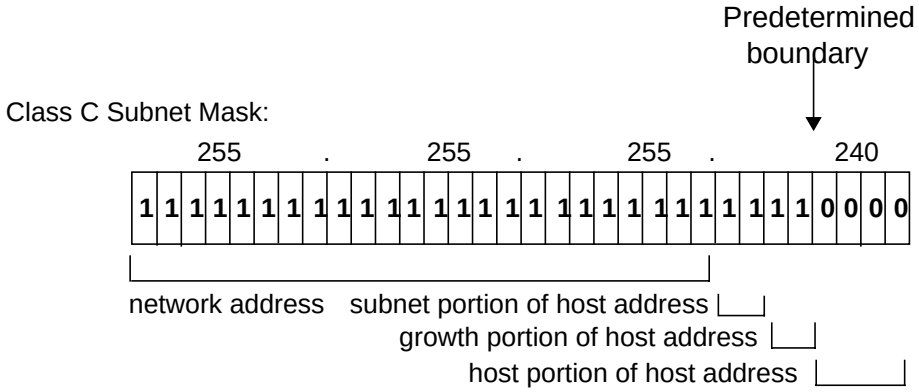
In this case, both the host field and subnet field have considerably more growing space than before, although the combined growing space is the same. As it is difficult to predict how many hosts might end up in a subnet, or how many subnets there might eventually be, this arrangement allows for maximum flexibility in growth.

Assigning Variable-Length Subnet Masks

In Figure 6-11, the boundary between the host and subnet fields is shown in the middle of the growth area. The boundary, however, could exist anywhere within the growth area. The subnet mask determines where the boundary is located. “Ones” in the subnet mask indicate subnet bits, and “zeros” indicate host bits. To minimize the amount of rework after the initial planning of your network, you should choose a subnet mask for a given subnet based on the projected growth of that subnet. As shown in Figure 6-12, the subnet is projected to have a maximum size of 14 hosts. Therefore, the subnet mask should be 255.255.255.240. There are two remaining growth bits for this subnet. If the subnet grows beyond 14 hosts, you may choose to change one of the two remaining growth bits to a host bit. The new subnet mask will be 255.255.255.224.

Variable-length subnet masks are assigned using the `netmask` parameter of the `ifconfig` command, SUBNET_MASK in `netconf` with a 32-bit subnet mask indicated in dot notation, or SAM. In the example below with a Class C IP address, the host portion has three types of assigned bits: subnet number bits, growth bits, and host bits, with a chosen subnet mask that allows for growth in both the host field and subnet field without a mask change.

Figure 6-12 Example of Variable-Length Subnet Mask



The example below shows how the bits might look in the host address portion of a Class C address in a network with four subnets using variable-length subnetting. In this example “g” stands for growth bits and “h” stands for host bits. The 0's and 1's in the host address are assigned subnet numbers.

Table 6-8 Subnets with Variable-Length Subnet Mask

Subnet	Host Address (Bits)	Subnet Mask (Bits)
A	10gg gghh	1111 1100
B	01gg ghhh	1111 1000
C	110g ghhh	1111 1000
D	001g ghhh	1111 1000

If hosts are added to subnet B above, so that all of its remaining “g” bits become “h” bits, the mask in subnet B also must change. Table 6-9 shows the results of these changes. Also, two additional subnets have been added, subnet E and subnet F.

Table 6-9 Subnets with Subnet Mask (B) Modified

Subnet	Host Address (Bits)	Subnet Mask (Bits)
A	100g gghh	1111 1000
B	01hh hhhh	1100 0000
C	110g ghhh	1111 1000
D	001g ghhh	1111 1000
E	101g ghhh	1111 1000
F	1110 ghhh	1111 1000

Notice that these additions caused A to change its leftmost g-bit into a subnet-bit (“s” bit). With the addition of subnet address 101 for subnet E, the old subnet address 10 of subnet A must be changed to 100 so that the proper number of bits are used to make the subnet numbers unique.

Table 6-10 shows the results of removing subnet E.

Table 6-10 Subnets with Subnet E Removed

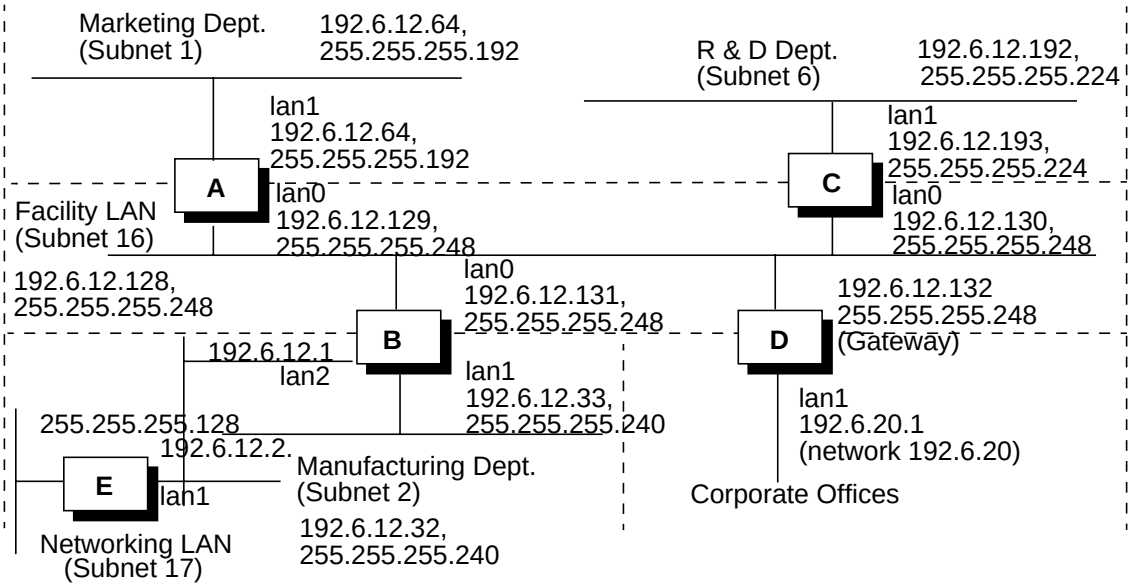
Subnet	Host Address (Bits)	Subnet Mask (Bits)
A	10gg gghh	1111 1100
B	01hh hhhh	1100 0000
C	110g ghhh	1111 1000
D	001g ghhh	1111 1000
F	1110 ghhh	1111 1000

Notice that with this change, subnet A gains back one growth bit (“g” bit) as its old subnet address of 10 is now unique again.

Example of Subnets with Variable-Length Subnet Masks

The following example shows four subnetworks within the 192.6.12 network along with the `netconf` entries necessary to configure these subnetworks with variable-length subnet masks. Note that there are four different subnet masks used in this network. Also note that the subnet numbers in the network map correspond to the mirror image subnet numbers listed in Table 6-7.

Figure 6-13 Network Map with Variable-Length Subnets



The subnet numbers shown below correspond to the subnets show in the network map in Figure 6-13.

In this example of mirror image counting, the first subnet has 6 hosts (with space allocated for a maximum of 6 hosts), the second subnet has 60 hosts (with space allocated for a maximum of 62 hosts), the third subnet has 30 hosts (with space allocated for a maximum of 30 hosts), and the fourth subnet has 14 hosts (with space allocated for a maximum of 16 hosts). Notice how the range of numbers (129-135, 65-127,193-223, and 33-47) is spread out to allow room for additional growth in each case.

Table 6-11 Variable-Length Subnet Addressing in a Class C Network

Subnet Mask 255.255.255	Subnet Number	Mirror Image Subnet Number	Subnet Address	Internet Address Range
248	16 (10000)	1 (00001)	n.n.n.128	n.n.n.129 - n.n.n.134
192	1 (01)	2 (10)	n.n.n.64	n.n.n.65-n.n.n.126
128	1 (01)	2 (10)	n.n.n.0	n.n.n.1-n.n.n.126 n.n.n.129-n.n.n.254
224	6 (110)	3 (011)	n.n.n.192	n.n.n.193-n.n.n.222
240	2 (0010)	4 (0100)	n.n.n.32	n.n.n.33-n.n.n.46

NOTE In a Class C network, no two subnets can have more than 125 hosts.

Summary network information:

```
Company division network = 192.6.12

Facility LAN subnet
    Subnet mask: 255.255.255.248
    Subnet number: 16
    Mirror image subnet number: 1
    Subnet address: 192.6.12.128
    Host address range: 129 to 134
    Host A internet address: 192.6.12.129 for network interface lan0
    Host B internet address: 192.6.12.131 for network interface lan0
    Host C internet address: 192.6.12.130 for network interface lan0
    Host D internet address: 192.6.12.132 for network interface lan0

Marketing Department Subnet
    Subnet mask: 255.255.255.192
    Subnet number: 1
```

```
    Mirror image subnet number: 2
    Subnet address: 192.6.12.64
    Host address range: 65 to 126
    Host A internet address: 192.6.12.65 for network interface lan1

Networking LAN subnet
    Subnet mask: 255.255.255.128
    Subnet number: 17
    Mirror image subnet number: 2
    Subnet address: 192.6.12.0
    Host address range: 1 to 126 and 129 to 254
    Host A internet address: 192.6.12.1 for network interface lan0
    Host B internet address: 192.6.12.120 for network interface lan0
    Host C internet address: 192.6.12.130 for network interface lan1
    Host D internet address: 192.6.12.182 for network interface lan1

R & D Department Subnet
    Subnet mask: 255.255.255.224
    Subnet number: 6
    Mirror image subnet number: 3
    Subnet address: 192.6.12.192
    Host address range: 193 to 222
    Host C internet address: 192.6.12.193 for network interface lan1

Manufacturing Department Subnet
    Subnet mask: 255.255.255.240
    Subnet number: 2
    Mirror image subnet number: 4
    Subnet address: 192.6.12.32
    Host address range: 33 to 46
    Host B internet address: 192.6.12.33 for network interface lan1
```

Example of Network Map with Variable-Length Subnets

This sample network combines networks, subnets with variable-length subnet masks, and clusters previously described and illustrated in this chapter. Note that the subnet masks for each IP address are shown on the accompanying worksheet.

Figure 6-14 Network Map II

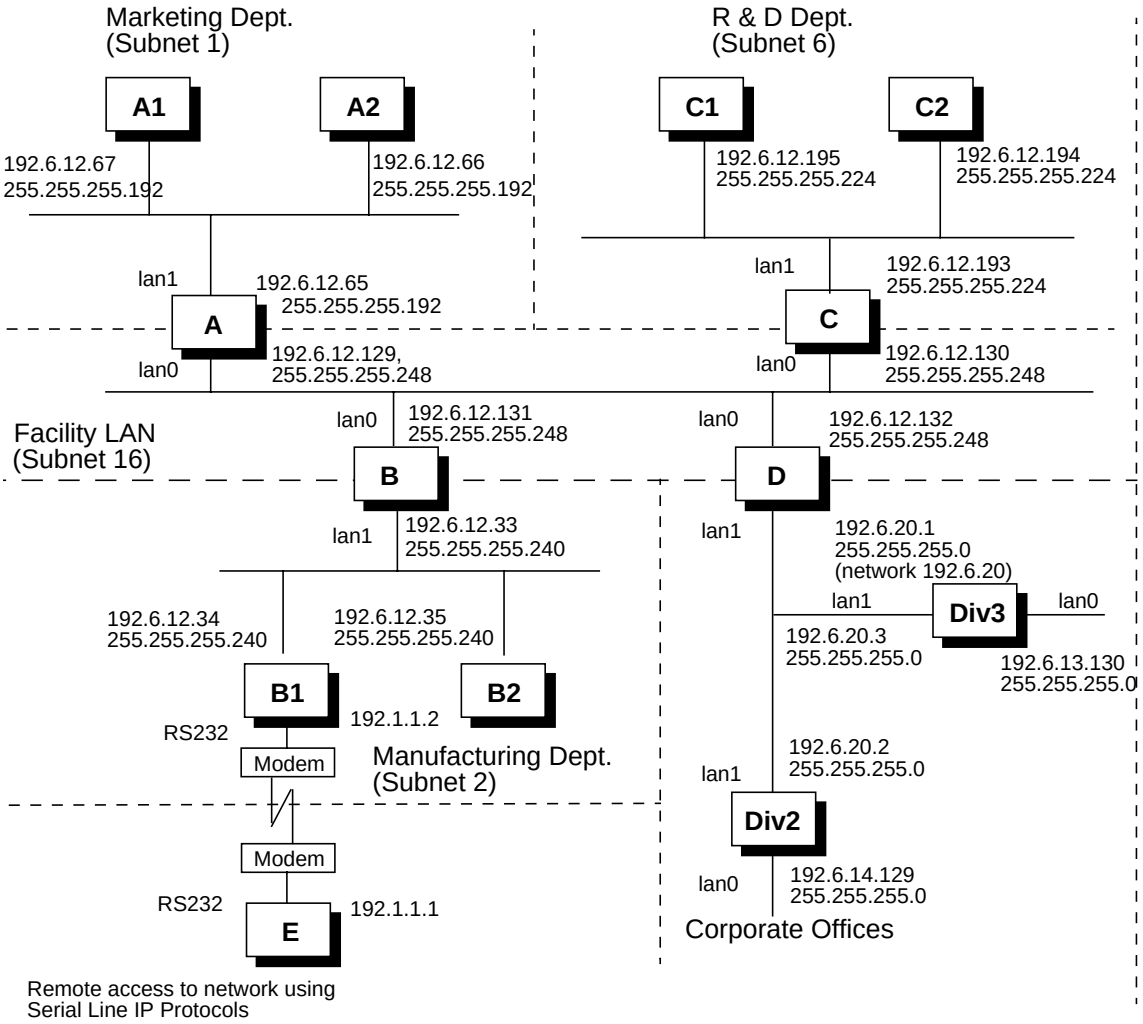


Table 6-12 Network Map II Worksheet

Host	Interface Alias	Internet Address	Subnet Mask	Cnode Type
A	mkt_32 mkt_64	192.6.12.129 192.6.12.65	255.255.255.248 255.255.255.192	Server
A1	mkt_a1	192.6.12.67	255.255.255.192	Client 1

Table 6-12 Network Map II Worksheet (Continued)

Host	Interface Alias	Internet Address	Subnet Mask	Cnode Type
A2	mkt_a2	192.6.12.66	255.255.255.192	Client 2
B	mfg_32 mfg_128	192.6.12.131 192.6.12.33	255.255.255.248 255.255.255.240	N/A
B1	mfg__b1 b1_ship	192.6.12.34 192.1.1.2	255.255.255.240 N/A	N/A
B2	mfg_b2	192.6.12.35	255.255.255.240	N/A
C	rd_32 rd_96	192.6.12.130 192.6.12.193	255.255.255.248 255.255.255.224	Server
C1	rd_c1	192.6.12.195	255.255.255.224	Client 1
C2	rd_c2	192.6.12.194	255.255.255.224	Client 2
D	div1_32 div_gw	192.6.12.132 192.6.20.1	255.255.255.248 255.255.255.0	N/A
Div2	div2_128 div2_gw	192.6.14.129 192.6.20.2	255.255.255.0 255.255.255.0	N/A
Div3	div3_128 div3_gw	192.6.13.130 192.6.20.3	255.255.255.0 255.255.255.0	N/A
E (SLIP)	e_slip	192.1.1.1	N/A	N/A

Configuring Gateways on Variable-Length Subnets

Besides using the appropriate subnet masks, each gateway needs to be configured so that it can properly route messages among the several subnet works. Following are descriptions of two types of routing: explicit routing and dynamic routing. When using explicit routing, you must specify the IP address of each gateway to which you are directly connected. When using dynamic routing, you need to specify only the IP address of one gateway, and the system learns the IP address of other gateways from the specified gateway.

Explicit Routing

There are many ways to set up routing. For example, you might add the following entries to the `netconf` file on Host A in Figure 6-14:

```
ROUTE_DESTINATION[0]="net 192.6.12.192"
ROUTE_MASK[0]="255.255.255.224"
ROUTE_GATEWAY[0]="192.6.12.130"
ROUTE_COUNT[0]="1"

ROUTE_DESTINATION[1]="net 192.6.12.33"
ROUTE_MASK[1]="255.255.255.240"
ROUTE_GATEWAY[1]="192.6.12.131"
ROUTE_COUNT[1]="1"

ROUTE_DESTINATION[2]="net default"
ROUTE_MASK[2]=""
ROUTE_GATEWAY[2]="192.6.12.132"
ROUTE_COUNT[2]="1"
```

The 1 in each ROUTE_COUNT entry specifies an indirect route. For example, messages for the system on the 192.6.12.32 subnetwork will first be sent to Host B (192.6.12.131), and from there they will be forwarded to the destination system.

Dynamic Routing

Alternatively, and perhaps the easiest way to manage growth on the 192.6.12 network, you might add the following entries to each `netconf` file.

Hosts A, B and C:

```
ROUTE_DESTINATION[0]="default"
ROUTE_GATEWAY[0]="192.6.12.132"
ROUTE_COUNT[0]="1"
```

Host D (Site gateway):

```
ROUTE_DESTINATION[0]="net 192.6.12.64"
ROUTE_MASK[0]="255.255.255.192"
ROUTE_GATEWAY[0]="192.6.12.129"
ROUTE_COUNT[0]="1"

ROUTE_DESTINATION[1]="net 192.6.12.192"
ROUTE_MASK[1]="255.255.255.224"
ROUTE_GATEWAY[1]="192.6.12.130"
ROUTE_COUNT[1]="1"

ROUTE_DESTINATION[2]="net 192.6.12.32"
ROUTE_MASK[2]="255.255.255.240"
ROUTE_GATEWAY[2]="192.6.12.34"
ROUTE_COUNT[2]="1"

ROUTE_DESTINATION[3]="default"
ROUTE_GATEWAY[3]="192.6.20.1"
ROUTE_COUNT[3]="0"
```

If you add a new subnetwork to the Facility LAN at a later time, you will need to add only an appropriate routing entry on Host D. It will not be necessary to configure the other subnet gateways A, B, and C.

With this configuration, each subnet gateway (Hosts A, B, and C) will initially route messages for a system outside its subnet to Host D. The subnet gateway, however, will learn of the more direct routes automatically when Host D redirects the messages to one of the other subnet gateways. Subsequent messages for the destination system will be routed directly to the appropriate subnet gateway.

For example, referring to Figure 6-14, suppose messages are sent from system A1 (192.6.12.67) to system B1 (192.6.12.34). The first message will actually be routed to Host D (through Host A). Host D then will redirect the message through Host B. At the same time, Host D will notify Host A that Host B is a more direct route for messages to system B1. Subsequent messages to system B1 will be routed directly from Host A to Host B.

Redirected routes are called dynamic routes. You can see these dynamic routes by executing the command `netstat -rv` on Host A. Dynamic routes are indicated in the display by a D flag.

Proxy ARP Server

The default direct route entry on Host D assumes that there is a proxy ARP server on the 192.6.20 network. If there is none, additional indirect route entries can be configured for each gateway that is directly connected to the 192.6.20 network.

For example, referring to Figure 6-14, you might add the following indirect routes to send messages to Division 2 and Division 3.

```
ROUTE_DESTINATION[4]=”net 192.6.14”
ROUTE_MASK[4]=”255.255.255.0”
ROUTE_GATEWAY[4]=”192.6.20.2”
ROUTE_COUNT[4]=”1”

ROUTE_DESTINATION[5]=”net 192.6.13”
ROUTE_MASK[5]=”255.255.255.0”
ROUTE_GATEWAY[5]=”192.6.20.3”
ROUTE_COUNT[5]=”1”
```

Configuring Gateways on Supernets

If all the hosts and gateways in your networks support variable-length subnet masks, then the gateway configuration of the supernet will be the same as in gateway configuration for variable-length subnets.

In Figure 6-18, if you use explicit routing, you may configure the following supernet route on Host A to enable Host A to communicate with any host on supernet 192.6.12

```
ROUTE_DESTINATION[0]="net 192.6.12"
ROUTE_MASK[0]="255.255.254.0"
ROUTE_GATEWAY[0]="192.6.14.2"
ROUTE_COUNT[0]="1"
```

If you use dynamic routing, then the default gateway, Gd, on network 192.6.14 must have the above supernet route configured.

If some of the hosts and gateways in your networks do not support variable-length subnet masks, then you must configure a separate network route for each of the networks in the supernet.

If you use explicit routing and Host A does not support supernetting, then you must configure the following two net routes on Host A.

```
ROUTE_DESTINATION[0]="net 192.6.12"
ROUTE_MASK[0]="255.255.255.0"
ROUTE_GATEWAY[0]="192.6.14.2"
ROUTE_COUNT[0]="1"
```

```
ROUTE_DESTINATION[1]="net 192.6.13"
ROUTE_MASK[1]="255.255.255.0"
ROUTE_GATEWAY[1]="192.6.14.2"
ROUTE_COUNT[1]="1"
```

If you use dynamic routing, then the default gateway, Gd, on network 192.6.14 must have the above net routes configured.

IP Multicast Addresses

IP multicasting provides a mechanism for sending a single datagram to a group of systems. Generally, only systems that have joined the multicast group process the datagrams.

Multicast datagrams are transmitted and delivered with the same “best effort” reliability as regular unicast IP datagrams. The datagrams are not guaranteed to arrive intact at all members of the destination group or in the same order as the datagrams were sent.

Membership in a multicast group is dynamic. Systems can join or leave groups at any time based upon the applications' behavior. A system remains a member of a multicast group until the last socket that joined the group is closed or drops membership in the group. A system can be a member of more than one group at a time. A system that has multiple interfaces might be a member of the same group on each interface.

IP Multicast Addresses

At the IP layer, a multicast address is a Class D Internet address with the following format:

Figure 6-15 Multicast Address Format

User IP multicast addresses can be in the range 224.0.1.0 through 239.255.255.255. The addresses 224.0.0.0 through 224.0.0.255 are reserved. The addresses of other well-known permanent multicast groups are published in the “*Assigned Numbers*” RFC (RFC-1060, March 1990).

Ethernet Multicast Addresses

The Ethernet data-link address, also called the link level or station address, is derived from the IP multicast address. The lower 23 bits of the IP multicast address are placed into the lower 23 bits of the Ethernet multicast address 01-00-5E-00-00-00 (hex). Ethernet multicast addresses can be in the range 01-00-5E-00-00-01 through 01-00-5E-7F-FF-FF.

NOTE Several IP multicast addresses may share the same Ethernet multicast address because the IP multicast address has 28 significant bits.

Multicast Routing

Multicast datagrams are sent through the interface associated with the default route. If that interface does not support multicast, attempts to send multicast datagrams will result in the ENETUNREACH error.

A default multicast route can be configured by specifying a network route for 224.0.0.0. The example below provides both the command line and `netconf` file entries.

```
route add 224.0.0.0 192.1.2.3 0 #192.1.2.3 is a local interface
```

```
ROUTE_DESTINATION[1]=”224.0.0.0”
ROUTE_GATEWAY[1]=”192.1.2.3”
ROUTE_COUNT[1]=”0”
```

Additionally, routes for specific multicast addresses can be configured just like any other host route. The example below provides both the command line and `netconf` file entries.

```
route add 224.1.2.3 192.5.6.7 0 #192.5.6.7 is a local interface
```

```
ROUTE_DESTINATION[2]=”224.1.2.3”
ROUTE_GATEWAY[2]=”192.5.6.7”
ROUTE_COUNT[2]=”0”
```

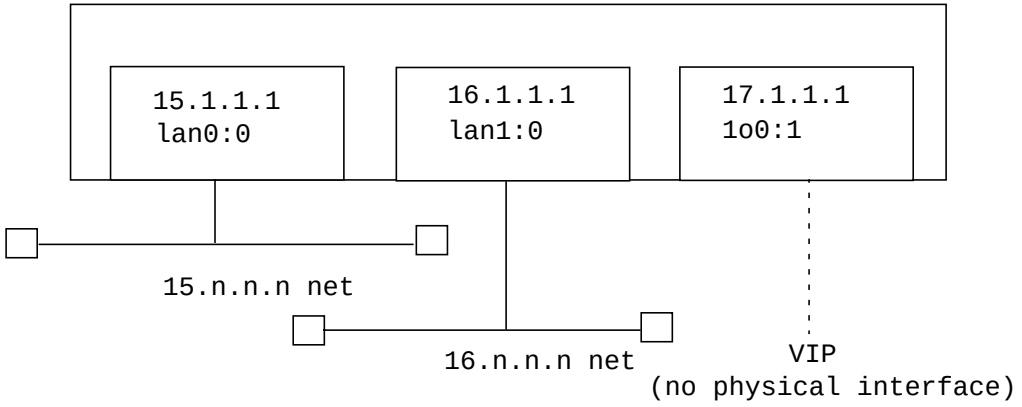
Virtual IP (VIP) Addresses

Systems can have Virtual IP (VIP) addresses that are not permanently assigned to a single, specific physical interface. The system will accept to a packet addressed to its VIP (or VIPs) regardless of the physical interface on which it was received. This allows a system to have a "system IP" address that is available as long as one interface stays usable.

To configure VIPs, associate the VIP address with a secondary loopback interface (lo0:n, where n is 1 or greater, such as lo0:1). The VIP address does not have to be in the same subnet (or network) of the addresses used for the physical interfaces.

In the example below, the system has two LAN interfaces. One is attached to the 15.n.n.n network and has the address 15.1.1.1. The second LAN is attached to the 16.n.n.n network and has the address 16.1.1.1. The VIP address is 17.1.1.1.

Figure 6-16



Note that the infrastructure of the network (routers, switches) must allow IP packets with the address 17.1.1.1 to be properly routed to this system's interfaces on the 15.n.n.n and 16.n.n.n networks for this configuration to be useful.

/etc/rc.config.d/netconf file statements for the above VIP:

```
INTERFACE_NAME[2]=lo0:1
IP_ADDRESS[2]=15.1.1.1
      :
      :
```

ifconfig command for the above VIP:

```
ifconfig lo0:1 inet 15.1.1.1
```

Note that you cannot assign VIPs to the primary loopback interface, lo0:0, or lo0.

CIDR - Classless Inter-Domain Routing

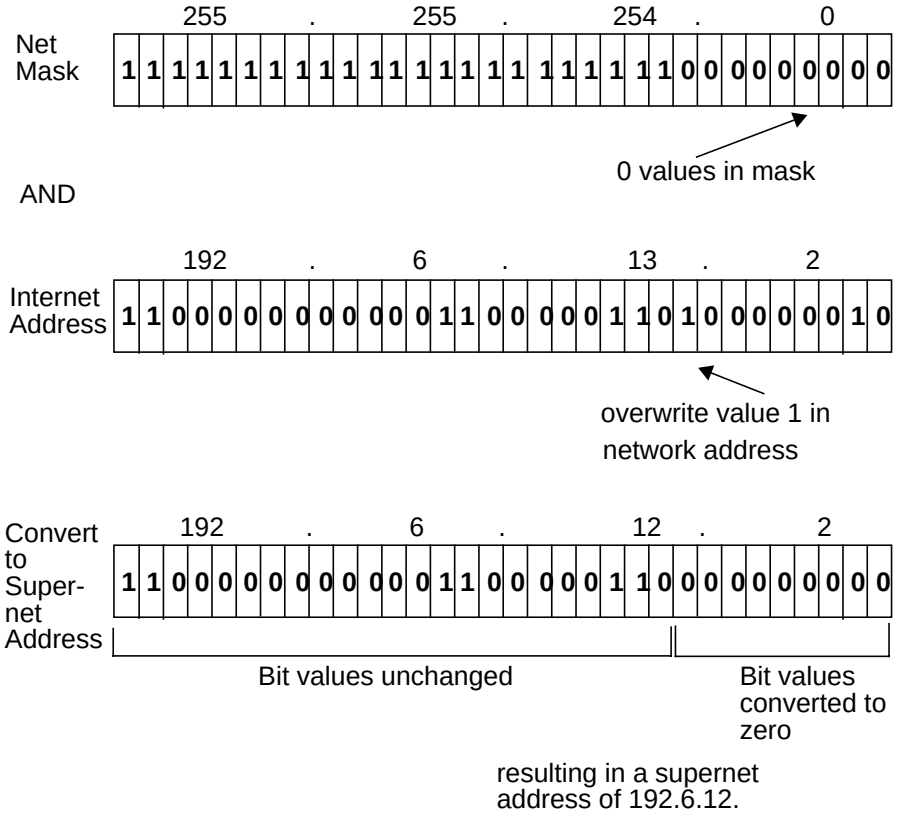
As the Internet has evolved and grown in recent years, it has become clear that it is facing several serious problems. These include:

- Exhaustion of the Class B network address space. One fundamental cause of the problem is the lack of a network class of a size that is appropriate for a mid-sized organization. Class-C, with a maximum of 254 host addresses, is too small, while Class-B, which allows up to 65534 addresses, is too large to be densely populated. The result is inefficient utilization of Class-B network numbers.
- Routing Information overload. The size and rate of growth of the routing tables in Internet routers is beyond the ability of current software (and people) to effectively manage.

Classless Inter-Domain Routing (CIDR) attempts to deal with these problems by defining a mechanism to slow the growth of routing tables and reduce the need to allocate new IP network numbers.

The basic idea of the CIDR plan is to allocate one or more blocks of Class-C network numbers to each network service provider. Organizations using the network service provider for Internet connectivity are allocated bitmask-oriented subnets of the provider's address space as required.

Figure 6-17 Internet Address 192.6.13.2 ANDed with Supernet Mask 255.255.254.0

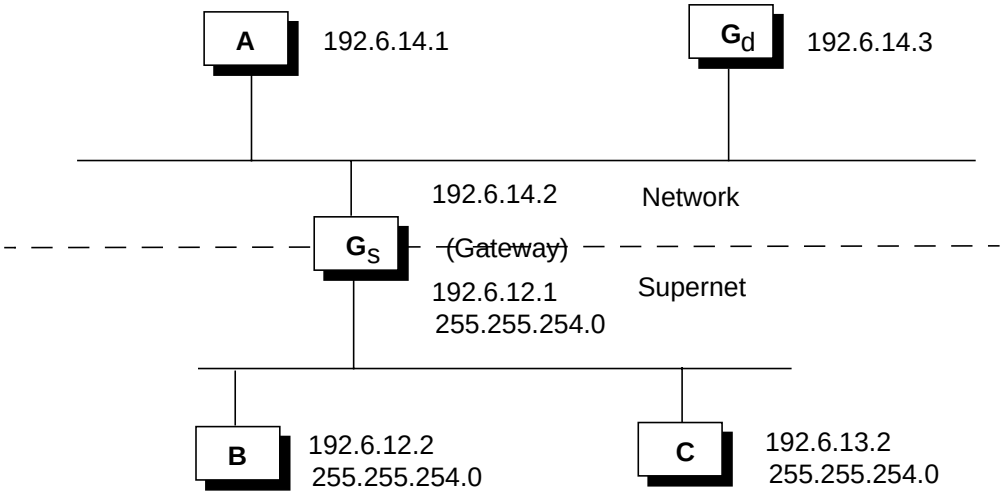


To implement this feature, you must apply the supernet netmask to all interfaces connected to the supernet using the `ifconfig` command. This feature will allow all hosts on the supernet to communicate with all other hosts on the supernet without standard routing.

In the example below, the hosts in two Class C networks, 192.6.12 and 192.6.13, are included in a larger supernet using the netmask option.

NOTE Since the gateway is itself a host on the supernet, it must support CIDR.

Figure 6-18 Network Map for Supernetting



7 LAN Device and Interface Terminology

Following is a description of terms used by the I/O subsystem to identify LAN cards and device files associated with LAN cards.

Interfaces

HP-UX 11i v2 allows you to configure multiple IP addresses for a single physical interface. This allows a single system to be seen as multiple systems, with multiple IP addresses and host names, even if the system has only one physical interface card. The IP addresses assigned to a card can generally be on the same subnet or on different subnets.

Logical interfaces are also used when an interface card is used for both IP/Ethernet and IP/IEEE802.3 packets. (In HP-UX 11i v2, all IP packets sent over IEEE802.3 must use Sub-Network Access Protocol (SNAP) encapsulation.) Sending IP packets using Ethernet and sending IP packets using IEEE 802.3 over the same card requires two separate logical interfaces. To send IP packets using Ethernet and IEEE 802.3, you must configure two logical interfaces, with two different IP addresses. In addition, the IP addresses must be in two different subnets. For logical interfaces with the same encapsulation, the IP addresses can be on the same or on different subnets.

In HP-UX 11i v2, the interface names used for `ifconfig` and `/etc/rc.config.d/netconf` statements can have a logical interface number appended to the card name. The syntax is:

nameX[:*logical_interface_number*]

name is the class of the interface. Valid names include lan (Ethernet LAN, Token Ring, FDDI, or Fibre Channel links), snap (IEEE802.3 with SNAP encapsulation), atm (ATM), du (Dial-up), ixe (X.25), or mfe (Frame Relay).

X is the Physical Point of Attachment (PPA). This is a numerical index for the physical card in its class. For LAN devices, the `lanscan` command displays the concatenated name and PPA number in the “Net-Interface NamePPA” field.

logical_interface_number is an index that corresponds to the logical interface for the specified card. The default is 0. The interface name lan0 is equivalent to lan0:0, lan1 is equivalent to lan1:0, and so on.

The first logical interface for a card type and interface is known as the **initial interface**. You must configure the initial interface for a card/encapsulation type before you can configure subsequent interfaces for the same card/encapsulation type. For example, you must configure lan2:0 (or lan2) before configuring lan2:1. Once you have configured the initial interface, you can configure subsequent interfaces in any order.

RARP Configuration

RARP (Reverse Address Resolution Protocol, RFC 903) configuration is an optional addressing scheme in which a freshly booted system queries a network server for the IP address of its own networking interface.

For example, if you have a large installation with many nodes, you can centralize the IP address database onto a system that serves as a RARP server. When each client system wants to configure its network interface, it queries the RARP server to provide the information listed in its database. In this way, each client system does not need to have its own customized version of the `netconf` file to contain its IP address.

When an interface needs to be configured with an IP address specified by a RARP server, the `/usr/sbin/rarpc` program is used to broadcast a RARP request over the network connected to the interface for which the information is being requested. If there is a RARP server on that network, and if the hardware address for the client interface is in the server's database, then the server will send a response to the client containing the IP address of that interface. The IP address is then passed to the configuration script, which uses it to configure the interface.

Setting Up a RARP Client

An interface is configured to use RARP configuration when the `IP_ADDRESS` variable in the `/etc/rc.config.d/netconf` file is assigned the value of RARP. The IP address variable is an array, each element of which corresponds to a particular interface number. Usually, this variable contains an explicit IP address, which is used to configure the interface specified by the `INTERFACE_NAME` variable.

To configure an interface to use RARP, the `/etc/rc.config.d/netconf` file must be modified so the `IP_ADDRESS` variable is set to `RARP`. This can be done with a text editor such as `vi`, and it can be done for interface entries other than 0. For example, for interface entry 1 to use RARP, the `IP_ADDRESS` line should be modified to `IP_ADDRESS[1]=RARP`. If the interface name for entry 1 is specified as `INTERFACE_NAME[1]=lan1`, then interface `lan1` will be configured with RARP information when the system is next booted.

When making this modification, the IP address should be recorded so the appropriate entry can be added to the server database.

Setting Up a RARP Server

A system is set up to be a RARP server when the `RARPD` variable in `/etc/rc.config.d/netconf` is set to `1`. Usually, this variable is set to `0` so that a RARP server is not started.

This can be done with a text editor such as `vi`. This will cause `/usr/sbin/rarpd` to be started at boot time.

Another file which must be modified is `/etc/rarpd.conf`. This is the RARP server database file. It contains a list of entries that map hardware address to IP address for each client interface.

The `/etc/rarpd.conf` file has the following format:

- A comment line is indicated by a pound sign (#) in the first column.

  Other than comment lines and blank lines, all lines are considered client entries. A client entry is of the form:

  — <hardware_address> <ip_address>

  where <hardware_address> consists of colon-separated (:) hexadecimal bytes, and <ip_address> consists of dot-separated (.) decimal bytes.

  Example:

  — 08:00:09:22:e4:a9 15.13.106.69

  There must be exactly 6 hardware address bytes.

  There must be exactly 4 protocol address bytes.

Once the `rarpd` server is started, it will receive RARP requests over all configured network interfaces, and it will respond to clients that have entries in the RARP database.

If the `/etc/rarpd.conf` file is modified while a `rarpd` is already running, the changes will not be reflected in the `rarpd` application until it is restarted or is sent the SIGHUP signal. To see what information is in the currently running `rarpd`, sending it a `SIGINT` signal causes it to dump its database into `/var/tmp/rarpd.db`.

See the `rarpd(1M)` and `rarpc(1M)` man pages for more information.

Index