



Applied
Microsystems
Corporation

NetROM™

Installation Notes

December 1995

Copyright © 1995 Applied Microsystems Corporation.
All rights reserved.

Information in this document is subject to change without notice. Applied Microsystems Corporation reserves the right to make changes to improve the performance and usability of the products described herein.

Applied Microsystems Corporation's CodeTAP products are protected under U.S. Patent 5,228,039. Additional patents pending.

Trademarks

CodeTAP is a registered trademark of Applied Microsystems Corporation.

CodeICE, RTOS-Link, CPU Browser, and NetROM are trademarks of Applied Microsystems Corporation.

Other product names, trademarks, or brand names mentioned in this document belong to their respective companies.

Contents

NetROM Installation Notes

Introduction	v
Support Services	vi

Chapter 1 Setting Up NetROM

Installation Steps	1-1
--------------------------	-----

Chapter 2 UNIX Environment Tips

Address Resolution Servers	2-1
BOOTP.....	2-2
RARP	2-4
File Transfer Servers	2-6
Daemons on UNIX Systems.....	2-7
inetd.....	2-8
Daemon Conventions	2-10
System Configuration Files	2-10
/etc/hosts.....	2-11
/etc/ethers	2-11
/etc/services.....	2-11
Using NIS (Yellow Pages).....	2-12
Finding Servers on Your Network.....	2-13

Chapter 3

NetROM Installation Troubleshooting

Serial Communications	3-1
Address Resolution—BOOTP	3-2
Address Resolution—RARP	3-5
Startup Batch File	3-7
Downloading Images and Emulation	3-9
Miscellaneous Problems	3-10

NetROM Installation Notes

Introduction

Installing NetROM on your network is similar to installing a new workstation. Like a workstation, NetROM is a multitasking computer. Also like a workstation, NetROM requires some configuration before it can operate effectively with the other machines on your network.

Consequently, we suggest that a network system administrator set up NetROM for your site. After NetROM is connected to the network, the only subsequent configuration necessary is to edit the NetROM startup file. Editing the startup file needs to be done in the early stages of every project in which you plan to use NetROM to debug a new target system. See Chapter 3 in the *NetROM User's Manual* for how to do this.

NetROM uses the network services BOOTP (or RARP) and TFTP for address resolution and file transfer. Chapter 2 of the *NetROM Installation Notes* contains tips on setting up address resolution and file transfer servers on your network.

Support Services

Applied Microsystems provides a full range of support services. New software is covered by a 90-day warranty that includes full applications phone support. Additional support agreements are available to extend the initial warranty and to provide additional services.

If you have trouble installing or using your software, consult your manuals to verify that you are following the correct procedures.

If the problem persists, call Customer Support. Customers outside the United States should contact their sales representative or local Applied Microsystems office. When you contact Customer Support, have your software serial number available. The serial number is printed on the label of your master media.

Telephone

800-ASK-4AMC (800-275-4262)

(206) 882-2000 (in Washington State or from Canada)

Internet address

If you have access to the Internet, you can contact Applied Microsystems Customer Support using the following address:

support@amc.com

You can also browse the Applied Microsystems World Wide Web page using the following URL:

<http://www.amc.com>

FAX

(206) 883-3049

Chapter 1

Setting Up NetROM

During normal operation, NetROM uses two network services: 1) address resolution and 2) file transfer. Address resolution is not necessary, but it is convenient because it allows NetROM to configure its network address automatically. File transfer allows NetROM to download files into its emulation memory.

NetROM can use either BOOTP or RARP as address resolution protocols. It uses TFTP as a file transfer protocol. If you are not familiar with BOOTP or RARP, consult “Address Resolution Servers” on page 2-1. If you are not familiar with TFTP, consult “File Transfer Servers” on page 2-6.

Installation Steps

➤ **To set up your NetROM, follow these six steps.**

1. **Determine an IP address for NetROM.**

Part of integrating NetROM into your network is assigning it an IP address. This address should be unique on your network. If the address is dually assigned, both NetROM and the machine that uses its IP address will behave strangely. After you have determined a unique IP address for NetROM, write it down and keep it handy.

Note



NetROM's Ethernet address is guaranteed to be unique.

2. Locate the Address Resolution and TFTP Servers.

Next, determine which machine(s) on your network will be address resolution and TFTP servers for your new NetROM unit. If you are not sure how to do this, consult “Finding Servers on Your Network” on page 2-13. Make sure that the server software is running on the machine, and is not in a disabled state. If you are not sure how to do this, consult “Daemons on UNIX Systems” on page 2-7.

After you have determined which server will be your address resolution server and which will be your TFTP server, write down their IP addresses and keep them handy.

3. Create a Startup Batch File.

After successfully resolving its address, NetROM will attempt to download a startup batch file. To verify that the batch file is being executed, it should do something which will be an obvious modification of NetROM's defaults. A good temporary batch file, used only to verify that it is being executed, is the following:

```
begin
    set prompt -d Bob>
end
```

This sets NetROM's default prompt to be the string “Bob>”; such a prompt will be immediately obvious when you are using NetROM.

4. Configure the Servers.

Edit the address resolution server configuration files so the server is aware of NetROM. This involves editing the bootptab file if you are using BOOTP, and the ethers file if you are using RARP. Consult “Address Resolution Servers” on page 2-1 for information on BOOTP or RARP. You will probably also want to edit the hosts file to assign a symbolic name for NetROM. Consult “System Configuration Files” on page 2-10 for information on system files. Note that if you are using NIS (also called the Yellow Pages) to distribute your hosts file or your ethers file you will need to update the NIS server and do a “push” operation. Consult “Using NIS (Yellow Pages)” on page 2-12 for information on NIS.

Further, if your server is running as a daemon, you may need to send it a SIGHUP to force it to re-read its configuration file. Consult “Daemons on UNIX Systems” on page 2-7 for information on UNIX daemons. NetROM’s hardware address is printed on a label attached to the bottom of the unit.

Part of server configuration entails placing the startup batch file where it can be accessed by NetROM after its address resolution. The location of the file and its name will vary, depending on the address resolution protocol you are using and on your TFTP server configuration. Consult “RARF” on page 2-4 for more information on naming configuration files based on address resolution protocol, and “File Transfer Servers” on page 2-6 for information on TFTP.

5. Connect NetROM to the Network.

Caution



NetROM contains components that are subject to damage from electrostatic discharge. Whenever you are using, handling, or transporting the hardware, or connecting to or disconnecting from a target system, always use proper anti-static protection measures, including using static-free bench pads and grounded wrist straps.

To connect NetROM to the network, simply attach an Ethernet connection. If you are using 10Base-T, plug the RJ-45 connector into the socket labeled “10Base-T” on the back panel. If you are using a transceiver, plug it into the connector labeled “Ethernet-AUI” on the back panel. You may need to use a short length of ribbon cable if your transceiver has slider knobs that don’t fit into the holes on the panel.

6. Power NetROM On.

Turn your NetROM unit on. If NetROM is connected to a target system, make sure that pin 1 on each emulation pod is plugged into pin 1 of the ROM socket. NetROM’s LEDs

should jump around for a second or two, then settle down with the heartbeat LED blinking rapidly. Chapter 2 of the *NetROM User's Manual* describes the mapping of the LEDs.

You should now be able to log into your NetROM unit with telnet. When you TELNET to the NetROM unit, check the prompt. If the prompt is "Bob>" you are now ready to use your NetROM on a target system.

If you cannot TELNET to your NetROM unit, or if you can TELNET to the unit, but the prompt has not been set to the string specified in your startup batch file, consult Chapter 3 of the *NetROM Installation Notes*.

After NetROM has been installed on your network, modify the startup batch file to perform whatever initialization is required for your project. Typical command sequences to include in the startup batch file set ROM type, word size, image file type, image file name, and download paths.

Chapter 2

UNIX Environment Tips

This chapter provides tips on various aspects of configuring a UNIX environment to support NetROM. It is not a comprehensive “how to” manual because of the variety of possible development environments. Rather, it provides background information and pointers to additional information.

If you have difficulties in installing your NetROM unit, consult your local documentation on related servers and files.

Address Resolution Servers

At boot time, NetROM. is capable of using either of two address resolution protocols to determine its IP address: RARP and BOOTP. Both are described in the following sections. Servers listen for broadcast Ethernet frames containing protocol requests. When such a request is received, the server looks up the requester’s Ethernet address in a configuration file. If it finds a match, it sends a reply containing the requester’s IP address. BOOTP replies may contain other configuration information as well.

After you have decided which address resolution protocol to use, you need to update the configuration file for the server. If you are not sure which servers you have available on your network, consult “Finding Servers on Your Network” on page 2-13. Note that part of NetROM’s address resolution process involves downloading a configuration file. This download is done using TFTP. If you are not familiar with TFTP, or are not sure whether your server is running in “secure” mode or not, consult “File Transfer Servers” on page 2-6.

This section describes how to configure standard UNIX servers. If you have problems, you should consult your own documentation on the server. Local documentation may reveal discrepancies such as nonstandard configuration file locations, and provides much more detail than can be given in this section.

BOOTP

The BOOTP address resolution protocol is quite flexible. The protocol allows the server to specify the client's IP address, as well as the name of a configuration file and a TFTP server to contact to download the file. BOOTP servers for UNIX systems generally come in one of two varieties: the "Stanford" server and the "CMU" server. The CMU server is able to specify IP netmask and broadcast addresses for the client, in addition to filename, server, and IP address. NetROM is capable of understanding responses from both server types; however, because the servers have different configuration file formats, so it is important to determine which type you have on your system.

BOOTP servers, like TFTP servers, are generally invoked in one of two ways. They may run continuously, as a daemon process that is started when their host machine is booted, or they may be invoked by the *inetd*. The *inetd* is a daemon process that listens for requests for a variety of services and invokes the appropriate servers automatically.

Consult "inetd" on page 2-8 for more information. Both types of BOOTP servers read the *bootptab* file for configuration information. Generally this file is located in the */usr/etc* or the */etc* directory. The location of the *bootptab* file can be specified when the server is invoked, so you may have to do some detective work to find it (look for *bootp* in */etc/rclocal*).

CMU BOOTP Server

The format of the bootptab file for CMU BOOTP is a set of “stanzas” separated by newlines. Each stanza is broken up into fields specifying what a BOOTP response to a particular client should contain. Fields within a stanza are separated by colons, and stanzas can be continued on subsequent lines using escaped newlines. For example, the stanza

```
netrom1:hd=/work/project/build:bf=startup.bat:\  
:sm=255.255.255.0:ht=1:ha=0002f4000024:ip=192.0.0.210
```

specifies the following information:

- Symbolic name of the client is netrom1.
- Directory /work/project/build contains the client's boot file.
- Name of the client's boot file is startup.bat.
- Client's subnet mask is 255.255.255.0.
- Client's “hardware type” is 1 (which stands for Ethernet).
- Client's hardware address is 00:02:f4:00:00:24 (note that colons are not used as separators on the stanza).
- The client's .i.IP address; is 192.0.0.210.

Note that the “hd” and “bf” fields are concatenated to give the full path to the configuration file. In this case, /work/project/build/startup.bat is the full path. If the TFTP server is running in “secure” mode, the file will actually reside in /tftpboot/work/project/build/startup.bat. Consult documentation on your server for other stanza fields.

To use a CMU BOOTP server to provide address resolution for NetROM, you must add a line similar to the above example to your bootptab file. If your BOOTP server is running continuously as a daemon, you need to notify it that you have changed the bootptab file by sending it a SIGHUP signal. See “Daemon Conventions” on page 2-10 for more information.

Stanford BOOTP server

The format of the bootptab file for the Stanford server is less complex than that of the CMU server. To configure a Stanford server for a hypothetical NetROM unit, we would add a line like the following:

```
netrom1 1 00:02:f4:00:00:24 192.0.0.210 /work/project/  
build/startup.bat
```

The first field is the symbolic name of the NetROM unit; the second field is the hardware type number (a 1 represents Ethernet); the third field is the NetROM unit's hardware address; the fourth field is its IP address in dotted-decimal notation; and the last field is the startup batch file for the NetROM unit. NetROM will use the IP address of the BOOTP server as the IP address of the TFTP server to contact for the startup batch file.

The Stanford server is capable of using defaults for the startup file, and indeed for the whole record. Such defaults are beyond the scope of this document. To use a Stanford BOOTP server to provide address resolution for NetROM, you must add a line similar to the above example to your bootptab file. If your BOOTP server is running continuously as a daemon, you need to notify it that you have changed the bootptab file by sending it a SIGHUP signal. See "Daemon Conventions" on page 2-10 for more information.

RARP

RARP is another way to establish a mapping between an Ethernet address and an IP address. Unlike BOOTP, the RARP protocol does not provide a mechanism to specify a download file. However, there is a convention that allows the RARP client to determine what configuration file it should download, as well as what its IP address is.

Also unlike BOOTP, RARP cannot be invoked by the *inetd*; if it is running at all it is running as a daemon.

The `ethers` file configures RARP servers. The `ethers` file is similar to the `hosts` file, except that the first field is an Ethernet hardware address, not an IP address. For example, the line

```
00:02:f4:00:00:24      netrom1
```

tells the RARP server that requests coming from hardware address “00:02:f4:00:00:24” should be responded to with the IP address of “netrom1”. (The mapping between the symbolic name “netrom1” and the IP address “192.0.0.210” is accomplished through the `/etc/hosts` file.)

NetROM maps RARP responses to startup file names as follows. First it converts the individual numbers in its dotted-decimal Internet address to hex. For `netrom1`, 192 becomes `0xC0`, 0 remains `0x00`, and 210 becomes `0xD2`. NetROM then concatenates these numbers to produce the filename (in uppercase letters) `C0000D2` and will attempt to download this file from the same network host that responded to the RARP request. Since NetROM cannot know whether or not the TFTP server is running in secure mode, it actually requests `/tftpboot/C0000D2`. If your server is running in secure mode, the batch file should actually reside in `/tftpboot/tftpboot/C0000D2`.

To use RARP to provide address resolution to NetROM, you must add a line similar to the one above to your `ethers` file. The `ethers` file is usually located in the `/etc` directory; however, if you are using the NIS (also called the Yellow Pages) to maintain the `ethers` file, you need to update the database on the NIS server and do a “yppush” operation. If NIS is running, the local `ethers` file may be silently ignored. Consult “Using NIS (Yellow Pages)” on page 2-12 for more information on NIS.

File Transfer Servers

TFTP is a standard file transfer protocol often used to provide boot images to standalone devices. The TFTP server can be run on any number of nodes on a network. TFTP servers generally have a “home” directory that they use as the default place to look for files when they receive requests. For example, if a server receives a request for `startup.bat`, it looks in its home directory for the file. The default home directory is `/tftpboot`, so by default the server looks for the file `/tftpboot/startup.bat` when it receives such a request. This section describes standard UNIX servers. If you have problems, you should consult your own documentation on the server. Local documentation may reveal discrepancies such as nonstandard home directories, and provides much more detail than can be given in this section.

The server can be run in either of two modes: “secure” mode or “normal” mode. These two modes differ primarily in the way they handle root-specific filenames in requests. Root-specific requests are those that specify directory location, starting at the root of the directory tree. For example, the client may request the file `/startup.bat`. Servicing this kind of request could lead to a security risk; the client could request any file on the server system, and since TFTP performs no authentication, the client would get the file. Secure TFTP was developed to get around this problem. Secure TFTP servers respond to root-specific requests as if their home directory (usually `/tftpboot`) were the root directory of their machine. Thus, a request for `/startup.bat` results in downloading `/tftpboot/startup.bat` from a secure server. Secure servers cannot be circumvented using symbolic links or similar tricks.

The problem with secure servers is they prevent legitimate clients from downloading files outside of the home directory subtree. For example, NetROM may want to download a file from an engineer's development directory, but secure-mode TFTP can make this impossible. There are at least two ways to

handle this problem: 1) either do not use secure TFTP, or 2) copy images into the server's home directory subtree after each modification.

To determine what the home directory is for your server, and to find out whether it is running in secure mode, it is necessary to find out how the server is invoked. TFTP servers can be run as daemons that are invoked at the server system's boot time, or they can be started by the *inetd*. Consult "Daemons on UNIX Systems" on page 2-7 for information on daemons on UNIX systems. After you have determined how the server is invoked, it is relatively simple to find out what its home directory and security mode are. Generally a *-s* flag indicates a secure server, and any kind of pathname indicates a non-default home directory. Consult local documentation for further information on your server.

Daemons on UNIX Systems

On UNIX systems, "daemon" is a term used to refer to processes that run constantly in the background waiting for requests. Network servers are usually run as daemons because they are invoked more or less at random by clients. Daemons are usually started up at system boot time. There are two shell script files which are executed at system boot; these are */etc/rc.boot* and */etc/rc*. These shell scripts can call other shell scripts, such as */etc/rc.local* or */etc/rc.ip*; daemon processes are usually started by *rc* or *rc.local*.

If you know that a daemon, such as the TFTP server, is running, and you want to know what its invocation parameters were, you should examine the various "rc" scripts in */etc*. It is often useful to know what the invocation parameters for a daemon are; in the case of the TFTP server, they will help you find out what the server's "home" directory is, and whether or not it is running in "secure" mode. The UNIX command:

```
unixprompt> grep -i rarp /etc/rc*
```

will look through all of the rc scripts and prints all lines which have “rarp” in them, regardless of case. Generally UNIX daemons have descriptive names; the RARP server is usually called “rarpd” for example. However, the TFTP server is often called “in.tftpd,” so it is not safe to assume that daemon names are a protocol name followed by a “d.”

If you find something interesting with the above “grep” command, such as a line that looks like

```
rarpd -a
```

remember that a leading pound sign (“#”) comments out the line, and that the invocation may be inside of a conditional statement in the shell script. You will need to look at the file more closely to see what is going on in more detail. If you are pretty sure a daemon is being invoked, consult “Finding Servers on Your Network” on page 2-13 for tips on how to verify that it is actually running.

inetd

Since UNIX systems are generally network oriented, they provide many network services. Many of these services are implemented using daemons that listen for connections to well-known (“advertised”) TCP or UDP ports. A detailed discussion of TCP and UDP is beyond the scope of this document, but, in brief, “ports” are simply numbers that allow the TCP or UDP protocols to determine to which (of possibly many) protocol connections data should be delivered. Common UNIX daemons provide TELNET, rlogin, rsh, ftp, uucp, and other services, not to mention TFTP, BOOTP, and RARP.

To help reduce the number of daemon processes to a manageable level, most UNIX systems use a special daemon, the *inetd*. The *inetd* is a server of servers; that is, it listens for requests destined for some particular server, such as TFTP, and invokes the TFTP server to respond to the request. The *inetd* is configured by the *inetd.conf* file, usually found in /etc. A typical line in the *inetd.conf* file might be

```
tufty dram up wait root /usr/etc/in.tftpd in.tftpd /
```

```
tftpboot
```

This line configures *inetd* to monitor requests for the TFTP server. The first field is the symbolic name of the port to listen on; the second and third fields, “dgram” and “udp,” tell *inetd* that the server uses UDP; the fourth field, “wait,” has to do with *inetd*’s behavior when there are multiple simultaneous requests; the fifth field indicates that the TFTP server will execute with root (super-user) permissions; the sixth field is the location of the server program’s executable code; and the remainder of the line is the command line used in the invocation of the TFTP server by *inetd*. Note that the invocation includes the name of the daemon. As far as “in.tftpd” is concerned, “tftpboot” is its first parameter.

The mapping between the symbolic name “tftp” and the actual port number of the TFTP protocol is done using the services file, usually located in /etc. Note that services may be mapped using the NIS as well. Consult “System Configuration Files” on page 2-10 and “Finding Servers on Your Network” on page 2-13 for information about the services file and about NIS.

As with most UNIX configuration files, a leading pound sign (“#”) comments out the line. If your BOOTP or TFTP server is supposed to be started by *inetd*, but doesn’t seem to be working, check for commented out lines in *inetd.conf*. For example,

```
# bootpd dgram udp wait root /etc/bootpd bootpd
```

does not start the BOOTP server, though it might look like it does.

If you edit the *inetd.conf* file, for example to remove a pound sign, make sure that you tell *inetd* to re-read its configuration file. This is done by sending it a SIGHUP signal. See “Daemon Conventions” on page 2-10 for details.

As a final note on the *inetd*, note that it cannot be used to start the RARP server. This is because RARP does not use TCP or UDP as a transport protocol. If RARP is running, it is running as its own daemon on your system.

Daemon Conventions

Since daemons are invoked at system boot time, they are not generally killed and restarted. However, if a change is made to a daemon configuration file, such as *inetd.conf* or *bootptab*, the daemon must be notified. This is usually done by sending the daemon a SIGHUP signal. This can be done using the UNIX kill command. For example, to reconfigure the *inetd*, you must first edit its configuration file. Then, determine the process number of the daemon, by executing

```
unixprompt> ps -aux | grep inetd
```

Then send it the signal, by executing

```
unixprompt> kill -HUP process-number
```

where “process-number” is the process number obtained with “ps.” Note that you must be the super-user to send signals to daemons running with root permissions.

If a server is started by *inetd*, there is no need to send it a signal after editing its configuration file.

System Configuration Files

The UNIX system has a number of system configuration files. A few of these may need to be modified to make using NetROM more convenient. You must be super-user to modify the system files, and you should be aware that many of these files are silently ignored when the Network Information Service (also called Yellow Pages) is running on your system. See “Downloading Images and Emulation” on page 3-9 for more information on NIS.

/etc/hosts

The `/etc/hosts` file is used to maintain a mapping between the names of hosts on a network and their network address. The network address is the standard Internet dotted-decimal notation IP address. For example, the line

```
192.0.0.210      netrom1
```

allows networking programs to use the symbolic name “netrom1” to refer to the network node “192.0.0.210.”

/etc/ethers

The `/etc/ethers` file is used to maintain a mapping between the Ethernet addresses of hosts on the network and their IP addresses. This file is used by the RARP server. For example, the line

```
00:02:f4:00:00:24  netrom1
```

indicates that the host “netrom1” has the Ethernet address “00:02:f4:00:00:24.” Note that “netrom1” is a symbolic name; the mapping between symbolic names and IP addresses is made using the `/etc/hosts` file.

/etc/services

The `/etc/services` file establishes a relationship between the symbolic name for a service, its “port” number, and the protocol used to implement it. For example, the lines

```
telnet 23/tcp
tftp 69/udp
bootp 67/udp
```

indicate that “tftp” uses UDP port 69 and “bootps,” the BOOTP server, uses UDP port 67. You may need to add these lines to the services file on your server for the `inetd.conf` lines given in “Daemons on UNIX Systems” on page 2-7 to work.

Using NIS (Yellow Pages)

The Network Information Service, also called the Yellow Pages, is a distributed database used to automate updates to UNIX system configuration files. For a large network, updating `/etc/hosts` for every host, every time a new machine is added to the network, is a major chore. NIS allows a single machine to be the repository of the “real” files. This machine is called the master NIS server. There also may be several “slave” servers to reduce the load on the master.

A complete discussion of NIS is beyond the scope of this document. If you are using NIS in your development environment, consult your local documentation for information on how to add entries for your NetROM unit into the NIS database. This section provides only an overview of how NIS works and is not a “how-to” guide.

When a system is using the NIS as a client, it starts up a daemon process that determines the location of the nearest NIS server. This daemon, `ybind`, acts as an interface to the remote server. When a program on the client machine, such as `rarpd`, wishes to access a system file, such as the hosts file, it checks to see if `ybind` is running. If it is, the program uses `ybind` to get the hosts file data. If `ybind` is not running, it will read the local copy of the hosts file. Thus, if the NIS is being used in your system, you must be sure to update the master copies of the hosts and/or ethers files with NetROM's addresses. Changes to the local copies of these files will be quietly ignored.

You can determine whether NIS is running on your system with either of the two commands:

```
unixprompt> ps -aux | grep ybind
unixprompt> ypmatch -x
```

The first command checks to see if the `ybind` process is running, and the second makes contact with it and asks what its “map nickname table” is.

Finding Servers on Your Network

Use the following commands to find which machines on your network, if any, are running servers that will interact with NetROM. These commands look for daemon processes running, check the *inetd.conf* file for BOOTP and TFTP servers, and check the “rc” script files for invocations of servers.

```
unixprompt> ps -aux | grep rarp
unixprompt> ps -aux | grep tftp
unixprompt> ps -aux | grep bootp
unixprompt> ps -aux | grep ypbind
unixprompt> grep rarp /etc/rc*
unixprompt> grep bootp /etc/rc* /etc/inetd.conf
unixprompt> grep tftp /etc/rc* /etc/inetd.conf
```

For convenience, you may want to collect these commands into a shell script.

If you find servers running on the system, you need only configure them for your NetROM unit. If they are not actually running, and you find invocations in the “rc” files or in *inetd.conf*, you need to make sure they are actually being invoked. Remember that if ypbind is running you need to update the master NIS database and do a “yppush” operation, rather than update local copies of UNIX system files.

If you are sure that servers are available on a particular machine, but they don’t seem to be responding to NetROM, it is possible that they are not running. This may happen because the invocation of daemon servers in “rc” files or similar scripts are commented out or occur conditionally at system boot time. Servers started by *inetd* may have their line in *inetd.conf* commented out. You may have to do some detective work in the “rc” files or in *inetd.conf* to determine if either of these situations apply.

If your server is definitely running but still does not seem to be responding to NetROM, your server configuration files may not be set up properly. Consult Chapter 3 for additional help.

Chapter 3

NetROM Installation Troubleshooting

The following sections contain possible categories of problems you may encounter when installing your NetROM. Attempt to match your problem with one of the categories, then check the matrix for troubleshooting tips.

Serial Communications

The following matrix lists possible problems you may encounter with serial communications. Check the symptom column, try to identify the problem, and attempt the solution.

Symptom	Problem	Things to try
No response on NetROM console.	Serial line parameters are misconfigured.	Check baud rate, parity and stop bits on console terminal against NetROM defaults given in the <i>NetROM User's Manual</i> , Appendix E.
	Serial line is connected to the wrong socket.	Make sure that the RJ-45 connector is plugged into the socket labeled "Console" on NetROM's back panel.
	RS-232 connection is DCE-to-DCE or DTE-to-DTE.	Insert a null modem between NetROM and the console terminal.
No response on target console.	See NetROM console problems above.	See "things to try" for NetROM console problems above. Note that serial line parameters can be changed with the stty command.

Address Resolution—BOOTP

The following matrix lists possible problems you may encounter with BOOTP. Check the symptom column, try to identify the problem, and attempt the solution.

Symptom	Problem	Things to try
NetROM doesn't get a response from BOOTP server.	Server is not running.	If server is a daemon, make sure it is running. If started by <i>inetd</i> , make sure the "bootps" line in <i>inetd.conf</i> is not commented out.
	Server's <i>bootptab</i> file has not been configured for NetROM.	Make an entry in the <i>bootptab</i> file for your NetROM unit.
	Server does not know <i>bootptab</i> file has been modified.	Send the BOOTP daemon a SIGHUP signal using the <i>kill</i> command, <i>if</i> it is running as a daemon. Make sure it is not killed by the signal.
	Server's <i>bootptab</i> file is in the wrong format.	Make sure you have not created a Stanford-format entry for a CMU BOOTP server, or vice versa.
	Syntax error in the <i>bootptab</i> entry.	Make sure that your <i>bootptab</i> entry is exactly correct. Many BOOTP servers will silently ignore entries with syntax errors.
	The Ethernet address in the <i>bootptab</i> entry for NetROM is incorrect.	Check the address against the label on the bottom of the NetROM unit, or given by the <i>dilanceha</i> command on the NetROM console.

Symptom	Problem	Things to try
NetROM doesn't get a response from BOOTP server. (Continued)	The <i>bootptab</i> entry for NetROM uses a symbolic name for the unit's IP address, but the name hasn't been entered into the <i>hosts</i> file.	Make sure that there is an entry in the local <i>hosts</i> file for the NetROM unit. If using NIS, check the master database.
	Your BOOTP server is running on a machine which uses NIS, but the NIS <i>hosts</i> database does not have an entry for the NetROM unit.	Make sure that the master NIS database has an entry for the NetROM unit. Run the <i>ypmatch</i> command on the BOOTP server's machine to verify that the change has propagated to all NIS servers.
	Startup batch file doesn't exist, or the path to it is wrong.	Make sure that the startup file specified by your <i>bootptab</i> entry exists in the directory specified by the entry. Remember that the path will be different if your TFTP server is running in "secure" mode.
NetROM configures itself with the wrong IP address and/or downloads the wrong startup batch file.	More than one BOOTP server is responding to NetROM's request, or a RARP server is responding to the request.	Make sure that all BOOTP servers in your environment are using the same configuration information. NetROM prints a message on its console indicating which BOOTP server responded to it. Make sure that no RARP servers are responding to NetROM. NetROM prints a message indicating which RARP server responded to it.

Symptom	Problem	Things to try
NetROM configures its IP address but doesn't execute commands in the startup batch file.	Startup batch file doesn't exist, or the path to it is wrong.	See above.
	A RARP server is responding to NetROM's address resolution request, and the startup file calculated from NetROM's IP address does not exist.	Either create a startup file named according to NetROM's IP address, or configure the RARP server not to respond to NetROM.
	Another machine on the network has the same IP address as the NetROM unit, and this is confusing the TFTP download.	Make sure that NetROM's IP address is unique. Turn the unit off and attempt to <i>ping</i> it. If a response comes back, the IP address is in use.

Address Resolution—RARP

The following matrix lists possible problems you may encounter with RARP. Check the symptom column, try to identify the problem, and attempt the solution.

Symptom	Problem	Things to try
NetROM doesn't get a response from RARP server.	Server is not running.	Make sure that the server is running, using the <i>ps</i> command.
	Server's <i>ethers</i> file has not been configured for NetROM.	Make an entry in the <i>ethers</i> file for your NetROM unit.
	Server does not know <i>ethers</i> file has been modified.	Send the RARP daemon a SIGHUP signal using the <i>kill</i> command. Make sure it is not killed by the signal.
	The Ethernet address in the <i>ethers</i> entry for NetROM is incorrect.	Check the address against the label on the bottom of the NetROM unit, or given by the <i>di lanceha</i> command on the NetROM console.
	The <i>ethers</i> file entry for NetROM uses a symbolic name for the unit's IP address, but the name hasn't been entered into the <i>hosts</i> file.	Make sure that there is an entry in the local <i>hosts</i> file for the NetROM unit. If using NIS, check the master database.

Symptom	Problem	Things to try
NetROM doesn't get a response from RARP server. (Continued)	NIS is running on the server's machine, but the master NIS <i>ethers</i> database does not have an entry for NetROM. Your RARP server is running on a machine which uses NIS, but the NIS <i>hosts</i> database does not have an entry for the NetROM unit.	Make sure that the master NIS <i>ethers</i> database has an entry for the NetROM unit. Run the <i>ypmatch</i> command on the RARP server's machine to verify that the change has propagated to all NIS servers. Make sure that the master NIS <i>hosts</i> database has an entry for the NetROM unit. Run the <i>ypmatch</i> command on the RARP server's machine to verify that the change has propagated to all NIS servers.
NetROM configures its IP address but doesn't execute commands in the startup batch file.	Startup batch file doesn't exist, or the path to it is wrong.	Make sure that the startup file calculated from NetROM's IP address exists in the <i>/tftpboot</i> directory. If your TFTP server is in "secure" mode, the file must be in a <i>tftpboot</i> subdirectory of the server's home directory.
NetROM configures itself with the wrong IP address and/or downloads the wrong startup batch file.	More than one RARP server is responding to NetROM's request, or a BOOTP server is responding to the request.	Make sure that all RARP servers in your environment are using the same configuration information. NetROM prints a message on its console indicating which RARP server responded to it. Make sure that no BOOTP servers are responding to NetROM. NetROM prints a message indicating which BOOTP server responded to it.

Symptom	Problem	Things to try
NetROM configures its IP address but doesn't execute commands in the startup batch file.	A BOOTP server is responding to NetROM's address resolution request, and the startup file named in the response does not exist.	Either create a startup file named according to the <i>bootptab</i> entry, or configure the BOOTP server not to respond to NetROM.
	Another machine on the network has the same IP address as the NetROM unit, and this is confusing the TFTP download.	Make sure that NetROM's IP address is unique. Turn the unit off and attempt to <i>ping</i> it. If a response comes back, the IP address is in use.

Startup Batch File

The following matrix lists possible problems you may encounter with the startup batch file. Check the symptom column, try to identify the problem, and attempt the solution.

Symptom	Problem	Things to try
NetROM doesn't execute commands in the startup batch file	The batch file is not being executed at address resolution time.	See the above sections on troubleshooting BOOTP and RARP.

Symptom	Problem	Things to try
NetROM doesn't execute commands in the startup batch file. (Continued)	The batch file is not delimited by <i>begin</i> and <i>end</i> statements.	Put <i>begin</i> and <i>end</i> statements, on lines by themselves, at the beginning and end of the batch file.
	The batch file contains non-printable characters.	Make sure that the file has only printable characters. Files created with the <i>vi</i> editor will not have hidden formatting characters.
	There are syntax errors in the commands.	Errors during execution of the startup batch file are displayed on NetROM's serial console. If you do not have a terminal on the NetROM console, execute the batch file manually from a TELNET console and observe the status of the commands as they execute. Use the <i>batch</i> command to execute the commands.
	The file is too large.	NetROM batch files cannot be larger than 2048 bytes. Make sure that your file is smaller than this.
NetROM displays a message about TFTP errors such as "Access Violation" or "File Not Found"	The file does not reside at the named directory.	Make sure that the file exists at the named directory, and that the directory path leading to the file exists.
	The file exists at the named directory, but the TFTP server is attempting to control access to the file.	Make sure that the file has global read permissions. Consult documentation on your local TFTP server.

Downloading Images and Emulation

The matrix on the following page lists possible problems you may encounter with the downloading images and emulation. Check the symptom column, try to identify the problem, and attempt the solution.

Symptom	Problem	Things to try
NetROM doesn't execute commands in the startup batch file	The batch file is not being executed at address resolution time.	See the sections on troubleshooting BOOTP and RARP.
	The batch file is not delimited by <i>begin</i> and <i>end</i> statements.	Put <i>begin</i> and <i>end</i> statements, on lines by themselves, at the beginning and end of the batch file.
	The batch file contains non-printable characters.	Make sure that the file has only printable characters. Files created with the <i>vi</i> editor will not have hidden formatting characters.
	There are syntax errors in the commands.	Errors during execution of the startup batch file are displayed on NetROM's serial console. If you do not have a terminal on the NetROM console, execute the batch file manually from a TELNET console and observe the status of the commands as they execute. Use the <i>batch</i> command to execute the commands.
	The file is too large.	NetROM batch files cannot be larger than 2048 bytes. Make sure that your file is smaller than this.

Symptom	Problem	Things to try
NetROM displays a message about TFTP errors such as “Access Violation” or “File Not Found”	The file does not reside at the named directory.	Make sure that the file exists at the named directory, and that the directory path leading to the file exists.
	The file exists at the named directory, but the TFTP server is attempting to control access to the file.	Make sure that the file has global read permissions. Consult documentation on your local TFTP server.

Miscellaneous Problems

The following matrix lists other possible problems you may encounter. Check the symptom column, try to identify the problem, and attempt the solution.

Symptom	Problem	Things to try
NetROM responds to network connections from some hosts but not from others	The hosts it does not respond to are on different “subnets” from the ones to which it does respond.	Add a default route to NetROM’s routing table, using the <i>route</i> command.

Symptom	Problem	Things to try
You can't get out of your TELNET session with NetROM.	Your <i>telnet</i> program is waiting for you to tell it to exit.	Type the TELNET escape character (often (^]'), then type "quit" on the prompt that appears. Consult local documentation on TELNET for more information.
Even after you change it with the <i>stty</i> command, your target console exits when it sees the <i>old</i> "eof" character.	The "eof" character is interpreted by the <i>conspathtd</i> process, whose "eof" character was not changed by the <i>stty</i> command.	Set the <i>default</i> "eof" character in your <i>startup.bat</i> file using <i>stty -d</i> , and reset your NetROM unit.

Index

A

address resolution protocols 1-1
address resolution servers 2-1

B

background processes 2-7
BOOTP 1-1, 1-2, 2-1, 2-2
 CMU 2-3
 Stanford 2-4
BOOTP server
 no response 3-2
BOOTP stanza 2-3
bootptab file 2-2

C

CMU BOOTP server 2-3
CMU server 2-2
configuration
 file 2-1, 2-4
 RARP 2-4
configuring servers 1-2
connecting NetROM 1-3
console
 no response 3-1

D

daemon 2-7
defaults
 setting 1-2
downloading images
 troubleshooting 3-9

E

emulation
 troubleshooting 3-9
Ethernet connection 1-3

F

file transfer
 protocol 1-1, 2-6
 server 2-6

I

inetd 2-2
installation problems 3-1
IP address 1-1

L

LED indicators 1-3

N

network connections 1-3
NIS 1-2, 2-5
normal mode 2-6

P

push 1-2

R

RARP 1-1, 1-2, 2-1, 2-4
 no response 3-5, 3-6

S

- s flag 2-7
- secure mode 2-6
- securing a server 2-7
- server configuration 1-2, 2-1
- server security 2-7
- Stanford BOOTP server 2-4
- Stanford server 2-2
- startup batch file 3-7
- startup batch file
 - creating 1-2
- startup file 2-2
- startup file
 - RARP 2-4
- startup.bat 2-6
- subnets 3-10

T

- TELNET
 - troubleshooting 3-11
- TFTP 1-1, 2-1, 2-4, 2-6
- TFTP errors 3-8, 3-10
- TFTP server 2-7
- troubleshooting 3-1
 - address resolution-BOOTP 3-2
 - address resolution-RARP 3-5
 - BOOTP server 3-2
 - downloading images 3-9
 - emulation 3-9
 - eof 3-11
 - IP address 3-3, 3-6
 - miscellaneous 3-10
 - network connections, hosts 3-10
 - serial communications 3-1
 - startup batch file 3-3, 3-4, 3-6, 3-7, 3-9
 - stty command 3-11
 - target console 3-11
 - TELNET 3-11
 - TFTP 3-8, 3-10

U

- UNIX daemons 2-7
- UNIX servers 2-6
- UNIX tips 2-1

X-Y-Z

- Yellow Pages 1-2, 2-5
- yppush 2-5