3 Booting—Address Discovery

This chapter describes alternative methods of obtaining the network information necessary for the terminal to boot and to participate on the network. The following topics are covered in this chapter:

Summary of Address Discovery Protocols

When an NCD terminal powers up, it knows only its Ethernet or Token-Ring address, which is set in NVRAM at the factory. To participate on a network, a terminal must be able to discover its network address. Depending on your network setup, the terminal may need other information. The three ways for the terminal to discover addresses and other information before loading the X server are:

- ❏ DHCP (Dynamic Host Configuration Protocol) and BOOTP (Bootstrap Protocol)—DHCP and BOOTP are widely available protocols, and are the recommended methods of address discovery. By default, an NCD terminal broadcasts alternating DHCP, BOOTP, and RARP requests when it boots. These requests contain the terminal's Ethernet or Token-Ring address. A host running the DHCP or BOOTP daemon and configured with information about the terminal responds with the IP address of the terminal.

Depending upon the protocol implementation and the information in the host's database, it can return other addresses and permit the terminal to boot from a host on a different subnet. It can also specify the X server that each terminal boots.

For more information about DHCP and BOOTP, see "Using BOOTP / DHCP for Address Discovery" on page 3-3.

- RARP (Reverse Address Resolution Protocol)—RARP is another widely available address discovery protocol. By default, an NCD terminal automatically broadcasts alternating DHCP, BOOTP, and RARP requests when it boots. These requests contain the terminal's Ethernet address or Token-Ring address. A host running the RARP daemon and configured with information about the terminal responds with the IP address of the terminal.

  Unlike BOOTP / DHCP, RARP supplies only the address of the terminal and the address of the host that responded to the terminal's request for an X server. This method is recommended if you do not have BOOTP / DHCP on your network.

  For more information about RARP, see "Using RARP for Address Discovery" on page 3-10.

- NVRAM—You can save all of the necessary network addresses in NVRAM, where they remain even when the terminal is powered off. This method is recommended if:

  — You have no address discovery protocol (BOOTP / DHCP or RARP).

  — You are booting from a host on a different network through a router that cannot pass on the boot request.

  — You are managing a small group of terminals.

  For more information about saving addresses in NVRAM, see "Storing Addresses in NVRAM" on page 3-11 and Chapter 11, Boot Monitor and NVRAM.

Changing the Order of Network Information Requests

By default, a terminal sends requests in the following order: first DHCP, then BOOTP, then RARP. You can change the order in which the terminal sends requests using the NVRAM Setup menus. For information about changing the order of requests, see Chapter 11, Boot Monitor and NVRAM.

Using BOOTP/DHCP for Address Discovery

This section describes preparations for using BOOTP/DHCP. For BOOTP, you can use the native software on the boot host or the software included in the NCDware distribution.

If you are using DHCP to provide network information, the terminal configuration is the same as for BOOTP, and the terminal must have Boot Monitor version 2.8 or later.

If you are not using ***ncdinstall*** to prepare hosts and terminals for BOOTP/DHCP service, you need to perform the tasks described in the following subsections. The commands and steps may vary from those outlined in this section depending on your host, your BOOTP/DHCP implementation, and your network. These instructions are guidelines; they are not precise procedures.

NCD terminals send BOOTP/DHCP requests by default; consequently, terminal configuration is unnecessary unless you want to change the order in which the terminal sends requests for network information or specify a second or third source. The default order is first DHCP, then BOOTP, then RARP.

Note Information provided by BOOTP (for example, the subnet mask) is passed to the X server and may be stored in NVRAM, overwriting previous values.

The BOOTP protocol is implemented through daemon programs, such as ***bootpd***(8), and a database file, such as **/etc/bootptab**. Hosts configured to use NCD's dynamic IP address allocation for terminals use an additional database file, **/etc/bootptab.cfg**.

For specific instructions on using BOOTP, refer to the following sources:

For Information About:	Refer to:
NCD's BOOTP implementation	The ***bootpd*** man page in the NCDware distribution
The native BOOTP already installed on your host	Your host's documentation

BOOTP / DHCP options recognized by NCD terminals	Table 3-1 page 3-6
Dynamic IP address allocation using BOOTP	The ***bootpd*** (8) and **bootptab.cfg** (5) man pages in the NCDware distribution and the *System Administrator's Guide.*

For instructions on using DHCP, refer to your host system's documentation.

Making Sure that BOOTP Is Enabled

To verify that BOOTP is enabled on your boot host, check the following:

- To find out if ***bootpd*** is running, enter a ***ps***(1) command. For example:

  # **ps -axc | grep bootp**

  On some systems, the command is:

  # **ps -ef | grep bootp**

- Host operating systems based on the 4.3 BSD UNIX operating system (such as SunOS) require an entry in the **/etc/inetd.conf** file for BOOTP. Often, these entries exist, but are rendered ineffective by a comment symbol (#) at the beginning of the entry. If this is the case, remove the #. For example, for SunOS:

  ```
  bootps dgram udp wait root /usr/etc/bootpd bootpd
  ```

- Make sure that TCP / IP ports are reserved for the ***bootpd*** server and client processes in the **/etc/services** file.

  The usual entries in **/etc/services** are:

  ```
  bootps          67/udp
  bootpc          68/udp
  ```

  Remove any comment symbols (#) at the beginning of these entries.

- If you make any changes in the **/etc/inetd.conf** or **/etc/services** file, restart the ***inetd*** daemon.

  To restart the daemon, find the ***inetd*** process ID and send a hangup signal as shown in the following example. This causes the daemon to read the configuration file and use the new information. For example:

  ```
  # ps -axc | grep inetd
  17601 ?  I 0:12 inetd
  # kill -HUP 17601
  ```

On some systems, the command for finding the process ID is:

```
# ps -ef | grep inetd
```

Adding Options for NCD Terminals to the bootptab File

If you do not use ***ncdinstall*** to add terminals to the network, you must add BOOTP options for each terminal to the **/etc/bootptab** database file on the boot host.

bootptab Format and Options

For BOOTP configuration on your boot host, see your vendor documentation. The normal rules for **/etc/bootptab** file entries are:

- A colon (:) indicates the end of a field, and a backslash (\) indicates that the entry is continued on the next line.
- Spaces are not permitted between the characters on a line.
- Fields consist of a tag followed by an equals sign (=) and a value. Each tag identifies a unique parameter.
- The Ethernet address has no internal punctuation, such as periods or dashes.

The typical file structure is one or more template entries, containing information common to all terminals or a group of terminals, followed by individual entries, each containing information about a specific terminal:

```
# Template entry
template.name:\
     tag=value:\
     .
     .
     .
     tag=value

# Entry for an NCD terminal
hostname:\
     tc=template.name:\
     tag=value:\
     .
     .
     .
     tag=value
```

The tags recognized by NCD terminals and supported by the version of ***bootpd*** provided in the NCDware distribution are listed in Table 3-1. The table lists the normal two-letter tag name or a tag number and a description of the information provided by the tag. Although some versions of ***bootpd*** may provide additional information, only information listed in the table is used by NCD terminals.

Table 3-1 BOOTP/DHCP Tags Used by NCD Terminals

Tag Name	Information Returned by BOOTP or DHCP
ip	Terminal IP address
ha	Terminal Ethernet address
hd	X server directory
bf	X server filename
sm	Subnet mask
gw	Gateway address(es)
ns	IEN-116 name server host address(es)
ds	Domain name server host address(es)
cs	Vendor magic-cookie selector
hn	Terminal host name (you do not need to supply a value for this field; it is taken automatically from the first field [up to the first colon]).
ts	Names of hosts supplying the current time on UDP port 37
T15	Domain name suffix
T144	Configuration file name
to	Time offset from Coordinated Universal Time
ts	Time server host address(es)
T31	ICMP router discovery enabled
T28	IP broadcast address
T49	XDM (X Display Manager) host address(es), listed in order of preference

A fragment of an example **bootptab** file follows, with comments identifying the fields.

```
# Template entry
global:\                                    Template entry name
        gw=192.43.153.1:\                   Gateway address
        sm=255.255.255.000:\                Subnet mask
        hd=/tftpboot:\                      X server file directory
        ht=ether                            Interface name
# Entry for an individual NCD terminal
ncd1:\                                      NCD terminal's hostname
        tc=global:\                         Include the template
        ha=0000a70015d5:\                   Ethernet address
        ip=192.43.153.224:\                 IP address
        bf=Xncdxpl                          X server filename
```

Adding Terminal Entries to the bootptab File

The required and optional entries in the **bootptab** file are:

- One or more template, or global, entries for the information common to all terminals or groups of terminals (optional)
- An entry for each terminal containing at least the terminal's Ethernet address and IP address or its Token-Ring address and IP address

  If you do not supply the address and the terminal cannot determine its address through other means (using RARP or reading the address from NVRAM) the terminal will be unable to boot.

Note If the terminal has a TRP board, the current Token-Ring address (also called the active address) is different from the built-in Token-Ring address (also called the static address). When configuring BOOTP, use the current address.

Both addresses are printed on the sticker on the terminal base and on the packing box. The built-in address always begins with 00:00:a7 (for example, 00:00:a7:11:2a:4b). The current address is a bit-reversed version of the built-in address (for example, 00:00:e5:88:54:d2).

The current address is displayed by the Boot Monitor during booting. Both the current address and the built-in address are displayed in Statistics ⇒ Show Version. The current address is also displayed in Statistics ⇒ Show Statistics ⇒ Network Interfaces ⇒ Interfaces Table and in Statistics ⇒ Show Statistics ⇒ Token-Ring ⇒ Interface Table.

- ❑ If you use BOOTP/DHCP to specify the X server to be downloaded by the terminal, add the X server filename (and the X server directory name, if not specified in the global, or template, entry).

  If you do not specify the X server in the **/etc/bootptab** file, the Boot Monitor uses the default X server download sequence. For information about this sequence, see Chapter 4, Booting—X Server Loading.

  If you specify an X server file in **/etc/bootptab** and ***bootpd*** cannot locate the file, the terminal cannot download an X server. In that situation, the Boot Monitor displays its prompt and waits for a manual boot command. Syntax errors in the file may also prevent the terminal from booting.
- ❑ If the terminal is booting through a gateway (that is, servers are installed on a host located on a remote network), make sure the gateway to the boot host and subnet mask (if used) are specified in the **bootptab** file. See the next section for more information about booting through a gateway.

Configuring the Gateway Device and Terminal for Booting through a Gateway

For booting through a gateway, the device serving as the gateway and the **bootptab** file on the boot host must be properly configured.

This section provides an example setup (illustrated in Figure 3-1) for a Cisco router, which is a commonly used gateway device.

The IP addresses of the gateway's Ethernet interfaces are:

192.43.157.2	Ethernet Interface 0—for the remote network where the boot host is located.
192.43.156.2	Ethernet Interface 1—for the local network where the terminals are located.

The IP address of the boot host is 192.43.157.40.

The IP address of the NCD terminal is 192.43.156.171. Its Ethernet address is 0000a70023a3.

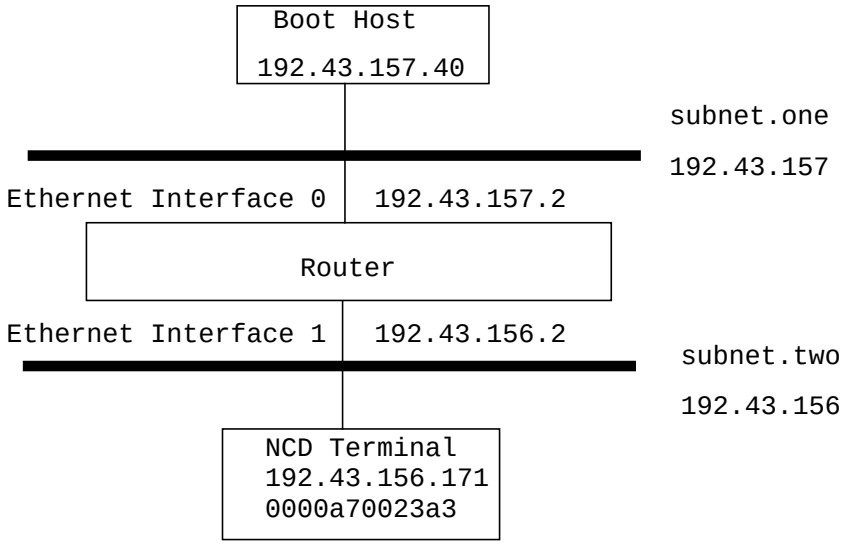

Figure 3-1 Booting Through a Gateway

On the router, set the helper address to the address of the remote boot host by entering the following command:

```
ip helper-address 192.43.157.40
```

On the boot host, the **bootptab** file contains the following entries for this example:

```
# Template entry - every host uses this information
global.dummy:\
    :sm=255.255.255.0:\                 Subnet mask
    :hd=/tftpboot/Xncd.4.2.0:           X server file directory

# Entries for each subnet
subnet.one:\                            Include the template and
    :tc=global.dummy:gw=192.43.157.2:     specify the gateway address
```

```
# Template entry - every host uses this information
global.dummy:\
subnet.two:\                                  Include the template and
    :tc=global.dummy:gw=192.43.156.2:           specify the gateway address

# Entry for an NCD terminal
ncdhmx1: tc=subnet.two:\                      Include the template
    :ht=ethernet:\                            Network interface type
    :ha=0000a70023a3:\                        Terminal Ethernet address
    :ip=192.43.156.171:\                      Terminal IP address
    :bf=Xncdhmx                               X server filename
```

Using RARP for Address Discovery

You can use RARP instead of BOOTP / DHCP to determine addresses; however, RARP returns only the IP address of the terminal and the boot host. You must set other addresses needed by the terminal, such as the gateway and subnet mask, in NVRAM. For information about configuring addresses in NVRAM, see “Storing Addresses in NVRAM” on page 3-11.

You cannot use RARP to specify the X server file to load; instead, the terminal uses the default download sequence to locate an X server. For information about the download sequence, see “Booting—X Server Loading” on page 4-1.

RARP is implemented through a daemon program, ***rarpd***(8) that runs on the boot host and a database file called **/etc/ethers**.

Complete the following tasks if you are using RARP for address discovery:

1. To verify that RARP is available, check the relevant host startup file. For example, on SunOS systems, the startup file to check is **/etc/rc.local**. The entry for RARP is similar to:

   ```
   rarpd
   rarpd -a
   ```

2. If the entry is disabled by a comment symbol (#) at the beginning of the line, remove the comment symbol. Then, start the daemon manually by typing the startup command. For example:

```
# rarpd
# rarpd -a
```

3. If you *are not* running NIS (Network Information Service), add each terminal's Ethernet address and hostname to the **/etc/ethers** file. For example:

   ```
   00:00:A7:00:00:AE ncd1
   ```

   The Ethernet address is entered into NVRAM at the factory. You can display the address through the Console (Statistics ⇒ Show Version). The Ethernet address also appears on a label on the bottom of the terminal base.

4. If you *are* running NIS, add the terminal to the **ethers** map and update the map. For example:

   ```
   # cd /var/yp
   # make ethers
   ```

Storing Addresses in NVRAM

You can manually set all the addresses the terminal needs in NVRAM. This is useful if you have just a few terminals to configure or if address resolution protocols are not running on the boot host. This method is less flexible than discovering addresses from the network because if you move the terminal to another network, you must reconfigure the addresses.

You can place other information in NVRAM, such as the name of an X server to download.

To configure a terminal to obtain addresses from NVRAM, set the following in Setup ⇒ Change Setup Parameters:

Note You can also set addresses in the Boot Monitor Setup menus.

1. In the IP hide box:
   a. Turn off the `Use Address Discovery` toggle so the terminal does not attempt to discover addresses from the network.
   b. Enter the addresses you need:

      You must enter at least the IP address of the terminal in `IP Address at Next Boot`.

If the terminal is booting through a gateway, set the `Subnet Mask`, `Initial Default Gateway 1`, and `Initial Gateway 2` (if you have more than one gateway). The terminal uses the boot host as a gateway if these fields are left at 0.0.0.0 (the default).

If desired, set the Broadcast Address. The default address of 255.255.255.255 works, although the Boot Monitor displays error messages as the terminal boots.

2. In the Booting hide box:
   a. Make sure `Primary Boot Source` is set to TFTP or NFS.
   b. Set `TCP/IP Desired Server` to the network address of the boot host. Set the `Secondary` and `Tertiary Server` fields, if desired, to designate backup boot servers.
3. Click on `Apply` to save addresses in NVRAM.

The next time the terminal boots, the Boot Monitor uses the addresses stored in NVRAM.

For more information about saving configuration settings set through the Setup menus or remote configuration files to NVRAM, see the *System Administrator's Guide for UNIX Systems*.

Setting the Broadcast Address

Whenever the terminal broadcasts to the network, for example, when discovering its IP address or broadcasting for an X server, it uses its broadcast address.

The default broadcast address is 255.255.255.255. If this address is not the correct address for your network and the terminal broadcasts for the X server, the server download succeeds but a warning message is displayed.

To prevent the warning message display, the broadcast address must be set to reflect the subnet mask, if any, and the host portion of the address.

If subnetting is used on your network, set the **ip-broadcast-address** parameter (Setup ⇒ Change Setup Parameters ⇒ IP ⇒ Broadcast Address). It should have all 1s in the host field. The parameter is saved in NVRAM.

Table 3-2 ip-broadcast-address Parameter

Possible Values	Result
default	255.255.255.255 or 0xFFFFFFFF
IP address	The address used by the terminal when broadcasting to the network.

For example, in a class B network with the network address 191.40.0.0 and a subnet mask of 0xFFFF0000 (255.255.000.000), the broadcast address setting is:

```
ip-broadcast-address = 191.40.255.255
```

Configuring Subnet Mask Discovery

Subnets are used to extend the network portion of IP addresses. This allows you to divide a physical network into separate subnets. The subnet mask distinguishes the subnet from the rest of the address. If subnetting is used on the local network, the terminal must be able to discover the subnet mask.

If a bit is on in the subnet mask, the equivalent bit in the IP address is interpreted as a network bit. If a bit is off in the mask, the equivalent bit in the IP address is interpreted as part of the host address. Therefore, the subnet mask has 1s in the network and subnet portions of the address and 0s in the host portion.

Subnet masks can be written as hexadecimal numbers or as decimal IP addresses.

For example, if the network portion of a class B address is extended by one byte, the subnet mask is 255.255.255.0. The first two bytes of the address define the class B network address, the third byte defines the subnet portion, and the fourth byte defines the host address.

There are three methods for making sure the terminal can discover the subnet mask:

- Set the subnet mask in the **bootptab** file (for information about using BOOTP to set the subnet mask, see “Using BOOTP / DHCP for Address Discovery” on page 3-3)

- ❏ Use ICMP to obtain the subnet mask from the network (see "Using ICMP to Discover the Subnet Mask" on page 3-14)
- ❏ Set the subnet mask explicitly in a remote configuration file, the Console Setup menus, or the Boot Monitor Setup menus and save it in NVRAM (see "Setting the Subnet Mask in NVRAM" on page 3-14)

Using ICMP to Discover the Subnet Mask

As an alternative to setting the subnet mask through BOOTP/DHCP or a terminal configuration parameter, the terminal can discover its subnet mask at boot time through ICMP (Internet Control Message Protocol). ICMP is included in the TCP/IP protocol family.

In this method of discovering the subnet mask, an ICMP message is sent to the broadcast address to determine the appropriate subnet mask when the terminal boots.

To discover the subnet mask through ICMP, set the **boot-send-broadcast-icmp-for-subnet-mask** parameter to "true." This option can produce a lot of network traffic, so you should use it only on networks with a small number of NCD terminals. This parameter is saved in NVRAM.

(Setup ⇒ Change Setup Parameters ⇒ Booting [TCP/IP Boot Options section] ⇒ Send Broadcast ICMP for Subnet Mask).

Table 3-3 boot-send-broadcast-icmp-for-subnet-mask Parameter

Possible Values	Results
default	false
false	The terminal does not use an ICMP message to determine the appropriate subnet mask.
true	The terminal uses an ICMP message sent to the broadcast address to determine the appropriate subnet mask.

Setting the Subnet Mask in NVRAM

Use the **ip-subnet-mask** parameter to set the subnet mask explicitly (Setup ⇒ Change Setup Parameters ⇒ IP ⇒ Subnet Mask). This parameter takes effect immediately and is saved in NVRAM.

Note You can also set the subnet mask in the Boot Monitor Setup menus.

Table 3-4 ip-subnet-mask Parameter

Possible Values	Result
default	0xFFFFFF00 (255.255.255.0)
hexadecimal constant or ***decimal IP address***	The network subnet mask

Communicating with Multi-Homed Hosts

If the terminal communicates with hosts that have more than one Ethernet interface on the same subnet, place all of the host addresses or hostnames in the **ip-equivalent-addresses** table. Each row in the table lists the network addresses or hostnames of the interfaces on a given host. This parameter is not saved in NVRAM.

Rows in the **ip-equivalent-addresses** table consist of the addresses or hostnames for all the interfaces on a given host. For example, on a network with two multi-homed hosts:

```
ip-equivalent-addresses = {
          { eagle1 eagle2 eagle3 }
          { peregrine1 peregrine2 }
          }
```

Using a Reverse Name Request

The **unit-query-for-name-at-boot** parameter controls whether, at boot time, the terminal sends a reverse name request to the name servers to discover the terminal's hostname.

Using a reverse name request is useful in situations in which the terminal's hostname must be known at boot time; for example, if the terminal should download a configuration file named for its hostname. This parameter is saved in NVRAM.

Note The DNS name service must be running for a reverse name request to succeed.

Table 3-5 unit-query-for-name-at-boot Parameter

Possible Values	Result
default	none
none	The terminal does not attempt a reverse name query at boot.
tcpip	The terminal tries a reverse query to the name server hosts. If no name servers are defined, the terminal tries the TCP/IP boot server, if the boot server is defined.
ncdnet	The terminal tries a reverse query to the MOP boot server, if the boot server is defined.