have no restrictions so it can be available to the TFTP file transfer protocol. This requires that a password contained in the configuration file must be encrypted.

Refer to Section 1.11.1.6 for more details.

```
# First, we define a system entry (such as subnet2) which
# specifies the things every host on that subnet uses.

subnet2.default:\

:hd=/home/HDSware/hds-fx/image:\

:bf=V30-D.hds:\

:T128="Galahad":\

:T129="test.3.cfg":\

:T130="/home/hdssun1/boot":

# Now come the individual ViewStations

marketing#1:\

:tc=subnet2.default:\

:ht=ethernet:\

:ha=008096000167:\

:ip=128.91.5.68:

marketing#2:\

:tc=subnet2.default:\

:hd=:bf=/home/HDSware/hds-fx/image/V30.hds:\

:ht=ethernet:\

:ha=00809600015d:\

:ip=128.91.5.13:
```

This sample file has sample entries for Bootp. Note that the vendor tag fields are present and that the configuration file called for the marketing#2 unit is test.3.cfg.

When you use BOOTP to specify the download of the boot file (server code), several different combinations of bootp fields can be used to attempt the download process. These combinations are listed below in the order they are tried, with the bootp field names shown in bold:

1. **T130** and **T129**
2. **T130** and hds.setup
3. **hd** and **T129** (if there is no T130 field)
4. path of **hd** and **hds.setup** (if there are no T129 or T130 fields)

3.2.3.1 Password Security with BOOTP

If you are using a password (T128) as part of your bootp process, note that the */etc/bootptab* file can have restricted access, but the configuration file (T129) must

- Read the README file to prepare to build the bootp daemon.
- Check the Makefile to set the configuration file location. The line in the Makefile should read:

```
CONFFILE = -DCONFIG_FILE=\"/etc/bootptab\"
```

- Now build the bootp daemon:

```
make
```

Step 2. When the bootp daemon executable is finished, you must verify that your */etc/services* file contains the following two lines:

```
bootps 67/udp # bootp server
bootpc 68/udp # bootp client
```

Step 3. Edit the */etc/inetd.conf* (or */usr/etc/inetd.conf* for SysV) file to start the bootpd process on reboot. The entry should read:

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

At this point, the bootp process should be running on the host.

3.2.2.2 BOOTP Debugging

The HDSware directory for BOOTP (in */HDSware/hds-fx/contrib/bootp/src/bootp.xxx*) contains some sample files that can help you work with BOOTP. A file called "dobootp" shows you how to start BOOTP in stand-alone mode, and sets the debug level to 5. This gives you a powerful diagnostic tool to see how BOOTP is working.

3.2.3 Sample *bootptab* File

This file would normally be located in */etc/bootptab*. If the file doesn't exist, copy the sample file provided with the source code:

```
cp /etc/bootptab.orig /etc/bootptab
```

In the bootptab file, note that lines marked # are comment lines. Note also that the passwords shown here are in clear text, since the bootp table can be read-protected on the system. (A configuration file specified by BOOTP cannot be read-protected, since it must be readable by TFTP, and so passwords occuring there must be encrypted. Configuration files loaded directly by the ViewStation can be read-protected, and so can use clear-text passwords.) The configuration file is loaded from the host which sent the bootp response.

This sample file contains a section for global settings, which would apply to everything on the system and/or also a subnet entry, which would apply to smaller divisions of the entire system. The entries for individual machines are entered last. Refer to the RFC documents and your system documents for the specific details of *booptab* file syntax. This is sometimes a difficult problem; for instance, some systems require the boot file name to have a leading slash, others refuse to read it with the leading slash, and some allow :hd and :bf on the same line, others do not.

This example starts the BOOTP server (bootps), gives the path to the BOOTP daemon itself, then the path to the bootptab, in this case "/etc/bootptab". This command line contains a timeout value of 0 (-t0 - never timeout) so you can always see whether the process is running. Enter the locations appropriate to your system configuration.

You should also check your */etc/services* file for entries for BOOTP's operation. This file sets the appropriate sockets for communication and should have an entry something like this for BOOTP:

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

These entries indicate that the BOOTP server (bootps) uses socket 67 and the BOOTP client (bootpc) uses socket 68. Check your operating system materials for syntax, though these UDP socket numbers should be uniform across all host types.

You should verify that the BOOTP daemon is running on your host. This can be a little tricky if the BOOTP daemon was initiated in the inetd process, since this doesn't actually start the BOOTP daemon until a BOOTP request is received (the default timeout for the daemon is 15 minutes). If you try to start a ViewStation and it sends a BOOTP request, then the daemon (if it is there) will start and linger long enough for you to check for its process with the **ps** command.

You can start BOOTP manually, or restart the inetd daemon as follows. First locate the inetd process using the **ps** command, then send that process ID a Hangup signal to restart the inetd daemon.

```
kill -1 <inetd process ID>
```

If you modify your bootptab, the BOOTP daemon will notice the new file date and reread it when the daemon receives a subsequent BOOTP request.

3.2.2.1 Installing BOOTP on your Host

If Bootp is not installed on your system, you can use the source code for bootp provided in the */HDSware/hds-fx/contrib* directory on the HDSware tape. The */contrib* directory has several executable bootp versions as well as the source code for bootp. Try the source code (BSD or SysV). If it doesn't work, you will have to compile the source code. Use the appropriate steps in the process outlined below.

Step 1. If you want to load the bootp code from the HDSware tape, move to the appropriate directory to unload the tape:

```
cd /tftpboot
tar xvfB <tape-device> ./hds-fx/contrib/bootp/src
```

(If your system is System V based, use this directory:)

```
./hds-fx/contrib/bootp/src/bootp.sys5
```

(If your system is BSD based, use this directory:)

```
./hds-fx/contrib/bootp/src/bootp.SUN4
```

:T130= (vendor tag) configuration directory
example: :T130="/home/hds-fx/config":\

BOOTP may be used to load the configuration file (containing Setup Mode values) or load only the ViewStation boot file (containing the server code) or both. Do not confuse these functions.

These entries and values, along with any other BOOTP fields your system may use, are retrieved from the bootp table and sent to the ViewStation in the bootp response. The configuration path and filename are also sent if the vendor tag fields are set. This configuration file contains individual resources and values for Setup Mode fields, which are given in detail in Section 1.7.3.

When you turn on the ViewStation, it sends the BOOTP request automatically. BOOTP then sets the IP address for the ViewStation and loads the software file. You have a root window with the Setup icon in the lower right corner.

3.2.1.3 Problems with BOOTP Starting

If there is a problem with starting the ViewStation via Bootp, try Monitor Mode, or look for these things:

- Verify that BOOTP is running on your system. If you do a “ps” command, you may not see a BOOTP process because it may be sleeping. The best procedure is just to restart it.
- Verify that your entries in the bootptab file are correct; verify the file syntax and the paths and filenames. Check the length of the response; some file names may be truncated for being too long.
- Verify that the file transfer protocol, TFTP by default, is running so the file can be transferred correctly.
- If you are running TFTP in secure mode, remember that this changes the root directory to your secure directory (such as */tftpboot*) and TFTP pre-pends this directory to your path. This must be reflected in your hd and T130 entries, which should then start with the directory below */tftpboot*.
- Note that there is a 64 byte limit (actually 60 bytes) on the “vendor” fields, which includes T128, T129, T130, gw, sm, and ds.

3.2.2 Setting Up BOOTP on your Host

The BOOTP daemon is usually provided with your host's operating system, though it may not be started as part of your system installation script. If you don't have a copy of the BOOTP daemon, you can get the source code or executable files for some systems from the */HDSware/hds-fx/contrib* directory of the HDSware tape or you can probably find a copy on the Internet.

The Bootp daemon is usually started from the */etc/inetd.conf* file, which manages the network services. If this file doesn't have a "bootp" entry, or the entry is commented out, you should add a line to start BOOTP. The syntax for this entry may vary, but it should look something like this:

```
bootps dgram udp wait root /etc/bootpd bootpd -t0 /etc/bootptab
```

3.2.1.1 Ethernet Hardware Address

The Ethernet hardware address is a 12 digit number (hexadecimal) unique to each network device. It consists of a six digit number for the vendor (for HDS this is 008096) and six digits more to make this a unique number.

This hardware address is found on a label on the ViewStation’s back panel or as part of the ViewStation serial number.

The ViewStation’s serial number consists of three characters to give the date of manufacture (first digit is the year - 4=1994; second two digits are the month - 05=May). The remaining six digits are the lowest six digits of the hardware address. Thus c8161b from the serial number, combined with the vendor code, would give the complete hardware address - 008096c8161b.

3.2.1.2 Bootptab Command Fields

The fields in the */etc/bootptab* file have an abbreviation for the entry name (such as **ha** for hardware address, and **ip** for IP address, etc.) and an associated value. These values are sent with the bootp response and received by the ViewStation to start its boot process. Check the syntax for these files on your particular system.

The BOOTP response is limited to 1024 bytes. The ViewStation uses the following Bootp fields:

:ha=	hardware address example: :ha=008096006a5c:\
:ip=	IP address example: :ip=128.91.6.15:\
:hd=	home directory example: :hd=/home/hdssun6/boot:\
:bf=	bootfile < *filename (limited to 128 chars)* > example: :bf=V30-D.hds:\
:gw=	gateway address example: :gw=128.91.2.5:\
:sm=	subnet mask example: :sm=255.255.0.0:\
:ds=	domain nameserver (entry is one or two addresses separated by a space) example: :ds=128.91.3.12 128.91.3.1:\ more than two entries are ignored; if only one is used, secondary is zero
:hn	host (ViewStation) name host(no entry; host does a name lookup based on :ip)
:T128=	(vendor tag) configuration password example: :T128="ApollO":\
:T129=	(vendor tag) configuration filename example: :T129="config.fx":\

BOOTP Operations

This section describes BOOTP operations and its use with booting and configuring ViewStations.

Section 3.0 - Network and Host System Operations

3.2 BOOTP Operations

BOOTP is a network service program (refer to RFC 951, RFC 1048, and RFC 1084 for specific details) running on your host computer that listens for startup or boot requests. When the BOOTP daemon receives a boot request, it looks in the bootp table (a file called */etc/bootptab*) for information about the machine making the request.

3.2.1 Starting with BOOTP

When you turn the ViewStation on, it sends a Bootp request (four times) as its default behavior. This boot request contains only the ViewStation's hardware address (ethernet address). BOOTP requests are supported across the Token Ring interface (effective with HDSware 3.2). The first host responding, whether from the Ethernet or the Token Ring network, is used as the BOOTP host.

You can disable the BOOTP requests on the Startup Options menu of Setup Mode by selecting "Setup" in the Determine IP address field.

When the BOOTP daemon receives a boot request, it takes the hardware address and matches it to an entry in its bootptab file. From this entry in the file, it takes the information appropriate for that ViewStation and sends a response that gives the ViewStation its assigned IP address, the location of its boot file, and so on.