NAME

boot − from power on to the login prompt

DESCRIPTION

At power on the machine reads the first sector of the boot device into memory and executes it. This bootstrap code loads **/boot**, the Minix Boot Monitor. The monitor loads the kernel binaries from **/minix**, or the newest file in **/minix** if it is a directory.

The Minix system is now running, the different tasks initialize themselves and control is transferred to the last one, **init**.

Init is the grandparent of all Minix processes, it is responsible for starting login processes on each terminal, but first it runs **/etc/rc**.

/etc/rc checks the state of the system and starts daemons. First it sets the keyboard translation to the mapping in **/etc/keymap** if present, followed by a call to **readclock**(8) to set Minix time from the hardware clock. Next the file systems are checked if necessary and the **/usr** file system is mounted.

The system is now ready for multiuser startup, **/etc/rc** calls **/usr/etc/rc** that cleans out /tmp, /usr/tmp, and resets or cycles log files by running **/usr/etc/daily**, starts the **update**(8) and **cron**(8) daemons, and initializes the network services. **/etc/rc** finally runs **/usr/local/etc** to initialize the system in a site or host dependent way.

Init reads **/etc/ttytab** and starts a **getty**(8) for each enabled terminal line to allow a user to log in.

OPTIONS

bootopts=−s

The value of the boot variable **bootopts** is passed to **/etc/rc**. If it contains **−s** then the system will run a single user shell before continuing with multiuser startup. (Note that one normally uses **boot −s** instead of setting **bootopts**.)

bootopts=−a

This flag tells that **/etc/fstab** must be ignored. The system asks for a device to use as /usr instead. This will also be done if the root device is not as mentioned in **/etc/fstab**.

bootopts=−f

Force a file system check, even if the system was shut down properly. (Do this once in a while to be sure about the state of the file systems.)

servers=*program*[,*program*...]

Names the special servers that must be started in /usr/etc/rc. The setting **servers=inet** will start the TCP/IP server.

BOOT ENVIRONMENT

Many features of the drivers inside the kernel are controlled by settings in the boot environmenti, like **bootopts** above does for **/etc/rc**. The values of these variables are usually colon or comma separated numbers configuring the driver. **DPETH0 = 300:10** tells the DP ethernet driver to use I/O address 0x300, interrupt request 10, and the default memory address (0xD0000, values may be omitted) for the first ethernet board. (Note that IRQ 2 is redirected to IRQ 9 on AT's and PS/2's, so use 9 if a device is jumpered for 2.)

Variables that are special to both the monitor and the kernel are described in **monitor**(8). This section lists extra variables or variable settings:

c*n* = **at** | **bios** | **esdi** | **xt** | **aha1540** | **dosfile** | **fatfile**

Choose the driver that is to be used as controller *n*, in order: IBM/AT (classic AT or newer IDE), BIOS (any disk), ESDI (some PS/2s), IBM/XT, Adaptec 154x, Minix under DOS "file as disk", FAT file system "file as disk". By default **at** is used on AT bus systems, **bios** on PS/2s and XTs, and **dosfile** when running under DOS. Most drivers are present in the kernel as distributed, but may be taken out by modifying **/usr/include/minix/config.h**. See **controller**(4). (An XT should always use the BIOS driver, not the XT driver, because BIOS calls are cheap on an XT. The XT driver can be used on AT machines with an old XT controller.)

DPETH*n* = **on** | **off**

Turn an ethernet board on or off. The driver is by default in "sink" mode for all boards. The sink mode allows one to use the driver without an ethernet board installed. The driver will play /dev/null for that device, i.e. nothing comes in, and anything send out is dropped on the floor. If the board is turned on then the driver will use it to send out packets, if it is turned off then the driver will fail for that board.

DPETH*n* = *I/O-addr*:*irq*:*mem_addr*:*mem_size* (WD80x3)
DPETH*n* = *I/O-addr*:*irq*:**0** (NE2000)
DPETH*n* = *I/O-addr*:*irq*:*flags* (3c503)

Set the I/O address (hex), IRQ (decimal), memory address (hex), memory size (hex), or flags (hex) of the *n*-th ethernet board and turn it on. By default they are configured as 280:3:D0000 and 300:5:C8000 with the memory size set to 2000, 4000, or 8000 depending on the type of board found. For the Western Digital cards the IRQ must be what the board expects, but the memory address is programmed into the board by the driver. The SMC EtherEZ board, a WD8013 successor, has only 8K memory. This confuses the driver, so you need to explicitly specify the board size as being 2000. The memory address and size have no meaning for the Novell ethernet boards, but the address may be explicitly set to zero to indicate that the board **is** a Novell ethernet board. For the 3Com 3c503 the third parameter are flags, with the low bit indicates that the on-board tranceiver must be used if 0 (thin ethernet), or that an external tranceiver is used on the AUI port if set to 1. The IRQ is software settable, and must be specified as 2 (XT), 3, 4, 5, or 9 (AT). The memory address is set on the board by jumpers. The driver does not support I/O mode for the 3c503. (Note the little differences between board types. For the 8003/8013 and NE1000/NE2000 the IRQ is fixed and the memory address variable, for the 3c503 the IRQ is variable and the memory address is fixed, but need not be specified. Messy.)

DPETH*n***_EA** = *e0*:*e1*:*e2*:*e3*:*e4*:*e5*

Set the ethernet address of the *n*-th ethernet board. The address is normally obtained from the ethernet board, so only in exceptional circumstances is this setting ever needed. (Use the address of the main server if you want a career change.)

AHA0 = *I/O-addr*:*bus-on*:*bus-off*:*tr-speed*

Configure the Adaptec 154xA SCSI host adapter to use the given I/O address (hex), Bus-on time (decimal), Bus-off time (decimal) and transfer speed (hex). The default is 330:15:1:00. The default transfer speed is always 5.0 Mb/s (code 00) ignoring the jumper settings.

aha1540-d*n* = *sleep-time*:*target*,*lun*

Program SCSI disk *n* to have the given target and logical unit number. The target and lun of a tape or other SCSI device may be changed by setting the **aha1540-d***n* variable that would be used had it been a disk. So tape device c0t7 can be set to target 4, lun 1 with aha1540-d7=:4,1. (The *sleep-time* parameter is present but ignored to be compatible with Minix-vmd.)

dosfile-d*n* = *file*

Tells the DOS virtual disk driver for disk *n* to use a given file as a disk. The file is a DOS file name that the boot monitor must be able to open.

fatfile-d*n* = *driver*:*minor*:*file*

Tells the FAT virtual disk driver for disk *n* to use a given file as a disk. The *driver* parameter is the name of driver that handles the disk, and *minor* is the device number of the partition where the file is found. See **controller**(4) for names and numbers. The *file* argument is the path to the file from the root directory down. The driver named must also be tied to a controller with a **c***n* variable, so that the FAT file driver can find it.

TZ = **GMT0**

This sets the time zone the hardware clock is running in. **Readclock** uses this to correctly obtain the time of the clock. The timezone of the system is set in **/etc/profile**. This boot variable is normally not set, only a few UNIX die-hards who don't care about the time Windows sees and don't want to change the clock twice a year for daylight savings use this option. (Set Windows time to the time zone of Casablanca to match.)

TCP/IP CONFIGURATION

To use TCP/IP you need to run the **inet** server, and unless you are running standalone you have to enable the ethernet driver. See the **servers** and **DPETH***n* boot variables above. The driver supports these ethernet cards: Western Digital 8003, Western Digital 8013, SMC Elite Ultra 16, Novell NE1000 and NE2000, 3Com Etherlink II (3c503). Many newer variants of the WD8013, now under the SMC brand, may also work.

You are likely to use TCP/IP in one of three situations:

Standalone with no connection to a network.

In a small network with no support from a "big" host.

Connected to a large network with address and name servers.

In each situation you need a different set of configuration files.

Standalone

All you need is a name and an IP address. Suppose the name is "flotsam" and the IP address is 192.168.0.1 from the private IP space, then this is put in **/etc/hosts**:

```
192.168.0.1     flotsam
```

And this in **/etc/dhcpd.conf**:

```
host 192.168.0.0/24 {};
interface ip0 flotsam;
```

Small Network

In a network where the Minix machine can't obtain its IP address and name from a different host you need specify the ethernet address of your machine and host names of all machines in the hosts and DHCP configuration files. Suppose your machine is to be named "flotsam", and another machine in the network is named "jetsam", and let's use network 192.168.0.0/24 again. The file **/etc/hosts** now looks like this:

```
192.168.0.1     flotsam
192.168.0.2     jetsam
```

And **/etc/dhcp.conf** like this:

```
host 192.168.0.0/24 {};
client 0:1:1b:a:68:ce flotsam;
```

Use **hostaddr −e** to find out what the ethernet address of your network card is. (The address above is an example.)

A host needs to have all hostnames used on your little network in its host file. In the DHCP configuration you only need the client entry of the system itself, but it may be useful to add all client entries to make them all the same.

If one of the machines is always on when any of the others is, then you can let it be a DHCP server. The other machines don't need a hosts or DHCP file anymore. If flotsam is the server then its **/etc/dhcp.conf** looks like this:

```
host 192.168.0.0/24 {
        DNSserver flotsam;
};
client 0:1:1b:a:68:ce flotsam { option server; };
client 0:0:c0:3a:12:10 jetsam;
```

Large Network

In a network with a central network administration your machine's IP address and name are given by the DHCP server. You don't need any configuration files. If you want your machine to do more, like being a router or something, then see **inet**(8) on setting up more networks.

Simpler configuration tools

The **dhcpd** and **nonamed** daemons are complex little programs that try to obtain information about

their surroundings automatically to tell the machine what its place in the network is. It should come as no surprise that there are simpler utilities to configure a machine. On a memory starved machine it may even be wise to configure a machine statically to get rid of the daemons. The first daemon, **dhcpd**, can be replaced by:

ifconfig −h *host-IP-address* **−n** *netmask*
add_route −g *gateway-IP-address*

to set the IP address and netmask of the machine. Note that you can only do this if the machine has a static IP address, or chaos will follow. Remove **/usr/adm/dhcp.cache** if the DHCP daemon has run before.

The name daemon, **nonamed**, can be replaced by an entry in **/etc/resolv.conf** that specifies an external name daemon:

nameserver *nameserver-IP-address*

The **ifconfig** and **add_route** calls can be placed in the file **/etc/rc.net**. Check **/usr/etc/rc** to see how **/etc/rc.net** can be used to override running the normal series of network deamons. Note that **/etc/rc.net** is sourced, so you can use the same variables and functions that **/usr/etc/rc** uses. These changes undo all the efforts to make Minix TCP/IP autoconfigurable. Make very sure that all the IP addresses are correct, and that the IP address of your machine is unique. (Mistakenly using the address of a main server will make all other machines look at your machine, and will make all the users of all other machines look at you.)

FILES

/boot	Minix Boot Monitor.
/minix	Kernel image, or directory containing them.
/etc/rc	Basic system initialization.
/usr/etc/rc	Complete system initialization.
/etc/rc.net	Specialized network initialization.
/usr/local/etc/rc	Per site initialization.
/etc/hosts	Name to IP address mapping.
/etc/dhcp.conf	Network initialization.
/etc/resolv.conf	Name resolver configuration.

SEE ALSO

monitor(8), **init**(8), **inet**(8), **loadkeys**(8), **readclock**(8), **fsck**(1), **fstab**(5), **update**(8), **cron**(8), **ttytab**(5), **getty**(8), **hostaddr**(1), **ifconfig**(8), **dhcpd**(8), **nonamed**(8), **tcpd**(8), **hosts**(5), **ethers**(5), **resolv.conf**(5), **inet**(8).

DIAGNOSTICS

Checking File Systems.

If the system has crashed then **fsck** is called for the root and /usr file systems. It is wise to reboot if the root file system must be fixed.

Finish the name of device to mount as /usr: /dev/

The prompt for the **−a** option, or if the name of the /usr file system has not been set in /etc/fstab. You can type a device name, say **fd0**.

Unable to obtain an IP address after 10 seconds.

TCP/IP misconfiguration. The DHCP daemon may have failed because the ethernet address of the machine is not known to the DHCP server, the DHCP configuration is not filled in properly, or the DHCP server can not be reached. Either talk to your Network Administrator, or make a dhcp.conf and a hosts file.

1.2.3.4 login:

If you see an IP address instead of a host name then the system failed to translate the IP address. Either talk to your Network Administrator to have the reverse address translation tables fixed, or

make a hosts file.

NOTES

The 10.0.0.0/8, 172.16.0.0/12, and 192.168.0.0/16 networks can be used for private networks. (This so-called CIDR notation names an IP address and the number of bits in the network number. So 172.16.0.0/12 includes all addresses from 172.16.0.0 to 172.31.255.255.) RFC-1597 will tell you why private networks are good, and RFC-1627 why they are bad.

BUGS

Indefinite hangs are possible if I/O addresses or IRQ's are wrong. A driver may babble about addresses and IRQ's, but that does not mean that what it says is true, it may just be configured that way. It is very difficult to find peripherals on a PC automatically, and Minix doesn't even try.

AUTHOR

Kees J. Bot <kjb@cs.vu.nl>