Q: Help! My machine has slowed to molasses. Even executing **ls** in a terminal takes forever. A **ps** shows the process numbers in the PID table are in the thousandsÐeven soon after reboot. There are 3 defunct processes owned by root with priority of -1 which can't be killed. When trying to launch memory hungry applications on my low memory (RAM memory) machine I get ªinsufficient memoryº errors. What can I do?

A: The key symptom of this problem is the defunct or ªzombieº processes shown in the PID table. A zombie process occurs when a child process dies after its parent has died, and no **wait**() has been done for the child. Once the child process dies, it is inherited by **init** and owned by **root**. These processes are *not* consuming any system resources, except for one process table slot. They are not using memory or CPU time because they are done executing and therefore cannot be killed. When a UNIX system exhibits this type of problem (spawning lots of zombies) then some system process (often a getty) is trying to start, is failing, and is re-trying endlessly.

Good places to look for causes of such problems are the following system files (all in **/etc**):

rc
rc.boot
rc.local
rc.swap
crontab
ttys
gettytab

Perform a checksum (using **sum**) on the suspect file on your system and compare to the checksum of the same file on your Software Release master. A difference in the two numbers does not necessarily indicate an errorÐyou might have intentionally changed somethingÐbut it

does mean the file should be more closely scrutinized, especially if you don't think you've changed the file.

For example, the first several lines of the /etc/ttys file on a stock, unmodified Release 1 through Release 3 disk looks like this:

```
#
# name  getty                 type          status        comments
#
# If you do not want to start the window server by default, you can
# uncomment the first entry and comment out the second.
#
# console       "/usr/etc/getty std.9600"       NeXT          on secure
console /usr/lib/NextStep/loginwindow     NeXT          on secure window=/usr/lib/NextStep/WindowServer
onoption="/usr/etc/getty std.9600"
```

```
ttya    "/usr/etc/getty std.9600"       unknown         off secure
ttyb    "/usr/etc/getty std.9600"       unknown         off secure
ttyda   "/usr/etc/getty D9600"          unknown         off
ttydb   "/usr/etc/getty D9600"          unknown         off
ttyp0   none                    network
ttyp1   none                    network
```

The man page for **ttys** explains that the flag **on** and **off** in the fourth column specify whether **init** should execute the command given in the second field. It's an error to specify **on** if the second field contains **none**. The **secure** further qualifies an **on** to allow root to login to this line. (See more about the ªsecureº flag below.)

Specifying **on** so that someone may log in over the network tty line causes the performance degradation described above. The network entries should *never* specify **on**. In other words, an

/etc/ttys file which contains the following entries is an *error*:

```
ttyp0   none            network         on  secure
ttyp1   none            network         on  secure
ttyp2   none            network         on  secure
```

If this is the case, either change the **on** entry to **off** and reboot, or restore the file from a Release disk and reboot. As an alternative to rebooting, after modifying the **ttys** file, you can type in a Terminal as root:

```
rhino-22# kill -HUP 1
```

This tells the **init** process to reread the **ttys** file.

Warning: specifying **off** and **secure** on ttyp0 (for example) is permissible but not advisable. This would allow users to **rlogin** or **telnet** as root to the machine on pseudo port 0 and can present a security risk. See Chapter 16, ªSecurity,º in the Release 2 *NeXT Network and System Administration* or Chapter 14, ªSecurity,º in the Release 3 *NeXTSTEP Network and System Administration* manual for more information on security in general.

QA592

Valid for 1.0, 2.0, 3.0, 3.1