CHAPTER 4

Building an HP-UX Kernel

Introduction

You may need to modify your HP-UX 11i kernel in some way, such as changing a kernel parameter, and then rebuild your kernel. You may need to create a new HP-UX kernel in order to add device drivers or subsystems, to tune the kernel to get improved performance, to alter configurable parameters, or to change the dump and swap devices. If you update or modify a dynamic element of your kernel, as shown in the example in this chapter, a reboot is not required. Updating or modifying a static element requires a reboot and may also require some additional steps.

With HP-UX 11i it is not necessary to rebuild your kernel for all changes that take place to it. In 11i, there are many *Dynamically Tunable Kernel Parameters* and *Dynamically Loadable Kernel Modules* that will modify your kernel but not require a reboot. Combined with many *Dynamic Patches* that are available in 11i, you will need to reboot your system less often. We'll cover the following two topics in this chapter:

- Manually build an HP-UX kernel - In the next section, we'll modify a *Dynamically Tunable Kernel Parameter*, thereby modifying the kernel, and do not have to reboot the system in order for the change to take place. We'll then make a change to the kernel and fully rebuild it so you can see the process of a complete rebuild, including a reboot. In this chapter, I discuss various commands related to kernel generation and cover the process by which you would manually create a kernel.

- Use **kcweb** to view, modify, and monitor the kernel - **kcweb** is a new Web-based kernel just becoming available at the time of this writing. Many such Web-based tools are planned for HP-UX, so we'll work with **kcweb** to perform many kernel-related functions.

Building a Kernel

New with 11i (first introduced in 11.0) was the introduction of dynamically loadable kernel modules. In 11.x, the infrastructure for this feature was put into place, providing a separate system file for each module. With 11.0 is provided the ability of specially created modules to be loaded or unloaded into the kernel without having to reboot the system as long as the module is not being used. HP-UX 11i continues to support all of this dynamic functionality. This new mechanism provides great flexibility and improved system uptime. Detailed information about this advanced feature can be reviewed in the H*P-UX 11.x Release Notes.* Most of the dynamically loadable kernel modules available at the time of this writing are third party. The *IT Resource Center* Web site (*itrc.hp.com*) contains information on this topic, including a developer's guide.

To begin, let's take a look at an existing kernel running on an HP-UX 11i L-Class system used in many of the examples throughout this book. The **sysdef** command is used to analyze and report tunable parameters of a currently running system. You can specify a particular file to analyze if you don't wish to use the currently running system. The following is a *partial* listing of having run **sysdef** on an 11i L-Class system:

```
# /usr/sbin/sysdef
NAME                     VALUE       BOOT       MIN-MAX        UNITS   FLAGS
acctresume                   4          -      -100-100                 -
acctsuspend                  2          -      -100-100                 -
allocate_fs_swapmap          0          -          -                    -
bufpages                 32074          -        0-            Pages    -
create_fastlinks             0          -          -                    -
dbc_max_pct                 50          -          -                    -
dbc_min_pct                  5          -          -                    -
default_disk_ir              0          -          -                    -
dskless_node                 0          -        0-1                    -
eisa_io_estimate           768          -          -                    -
eqmemsize                   23          -          -                    -
file_pad                    10          -        0-                     -
fs_async                     0          -        0-1                    -
hpux_aes_override            0          -          -                    -
maxdsiz                      2          -        0-655360      Pages    -
maxdsiz_64bit            16384          -      256-1048576     Pages    -
maxfiles                    60          -       30-2048                 -
maxfiles_lim              1024          -       30-2048                 -
maxssiz                  65536          -        0-655360      Pages    -
maxssiz_64bit           262144          -      256-1048576     Pages    -
maxswapchunks              512          -        1-16384                -
maxtsiz                   2048          -        0-655360      Pages    -
maxtsiz_64bit             2048          -      256-1048576     Pages    -
```

```
maxuprc               75          -         3-                   -
maxvgs                10          -          -                   -
msgmap           2555904          -         3-                   -
nbuf               18720          -         0-                   -
ncallout             515          -         6-                   -
ncdnode              150          -          -                   -
ndilbuffers           30          -         1-                   -
netisr_priority       -1          -        -1-127                -
netmemmax              0          -          -                   -
nfile                920          -        14-                   -
nflocks              200          -         2-                   -
ninode               476          -        14-                   -
no_lvm_disks           0          -          -                   -
nproc                400          -        10-                   -
npty                  60          -         1-                   -
nstrpty               60          -          -                   -
nswapdev              10          -         1-25                 -
nswapfs               10          -         1-25                 -
public_shlibs          1          -          -                   -
remote_nfs_swap        0          -          -                   -
rtsched_numpri        32          -          -                   -
sema                   0          -         0-1                  -
semmap           4128768          -         4-                   -
shmem                  0          -         0-1                  -
shmmni               200          -         3-1024               -
streampipes            0          -         0-                   -
swapmem_on             1          -          -                   -
swchunk             2048          -       2048-16384     kBytes  -
timeslice             10          -        -1-2147483648 Ticks   -
unlockable_mem      1800          -         0-           Pages   -
#
```

In addition to the tunable parameters, you may want to see a report of all the hardware found on your system. The **ioscan** command does this for you. Using **sysdef** and **ioscan,** you can see what your tunable parameters are set to and what hardware exists on your system. You will then know how your system is set up and can then make changes to your kernel. The following is an **ioscan** output of the same HP-UX 11i L-Class system for which **sysdef** was run:

```
# /usr/sbin/ioscan -f
Class       I  H/W Path     Driver    S/W State   H/W Type     Description
===========================================================================
root        0               root      CLAIMED     BUS_NEXUS
ioa         0  0            sba       CLAIMED     BUS_NEXUS    System Bus Adapter (582)
ba          0  0/0          lba       CLAIMED     BUS_NEXUS    Local PCI Bus Adapter (782)
lan         0  0/0/0/0      btlan     CLAIMED     INTERFACE    HP PCI 10/100Base-TX Core
ext_bus     0  0/0/1/0      c720      CLAIMED     INTERFACE    SCSI C896 Fast Wide LVD
target      0  0/0/1/0.7    tgt       CLAIMED     DEVICE
ctl         0  0/0/1/0.7.0  sctl      CLAIMED     DEVICE       Initiator
ext_bus     1  0/0/1/1      c720      CLAIMED     INTERFACE
                                                     SCSI C896 Ultra Wide Single-Ended
target      1  0/0/1/1.2    tgt       CLAIMED     DEVICE
disk        1  0/0/1/1.2.0  sdisk     CLAIMED     DEVICE       SEAGATE ST318203LC
target      2  0/0/1/1.7    tgt       CLAIMED     DEVICE
ctl         1  0/0/1/1.7.0  sctl      CLAIMED     DEVICE       Initiator
ext_bus     2  0/0/2/0      c720      CLAIMED     INTERFACE    SCSI C875 Ultra
Wide Single-Ended
```

```
target       3  0/0/2/0.2    tgt       CLAIMED     DEVICE
disk         2  0/0/2/0.2.0  sdisk     CLAIMED     DEVICE       SEAGATE ST318203LC
target       4  0/0/2/0.7    tgt       CLAIMED     DEVICE
ctl          2  0/0/2/0.7.0  sctl      CLAIMED     DEVICE       Initiator
ext_bus      3  0/0/2/1      c720      CLAIMED     INTERFACE
                                                        SCSI C875 Ultra Wide Single-Ended
target       5  0/0/2/1.4    tgt       CLAIMED     DEVICE
disk         3  0/0/2/1.4.0  sdisk     CLAIMED     DEVICE       TOSHIBA CD-ROM XM-6201TA
target       6  0/0/2/1.7    tgt       CLAIMED     DEVICE
ctl          3  0/0/2/1.7.0  sctl      CLAIMED     DEVICE       Initiator
tty          0  0/0/4/0      asio0     CLAIMED     INTERFACE    PCI Serial (103c1048)
tty          1  0/0/5/0      asio0     CLAIMED     INTERFACE    PCI Serial (103c1048)
ba          1  0/1          lba      CLAIMED     BUS_NEXUS    Local PCI Bus Adapter (782)
ba          2  0/2          lba      CLAIMED     BUS_NEXUS    Local PCI Bus Adapter (782)
ba          3  0/3          lba      CLAIMED     BUS_NEXUS    Local PCI Bus Adapter (782)
lan          1  0/3/0/0      btlan     CLAIMED     INTERFACE
                                              HP A5230A/B5509BA PCI 10/100Base-TX Addon
ba          4  0/4          lba      CLAIMED     BUS_NEXUS    Local PCI Bus Adapter (782)
ext_bus      4  0/4/0/0      c720      CLAIMED     INTERFACE
                                                           C875 Fast Wide Differential
target       7  0/4/0/0.7    tgt       CLAIMED     DEVICE
ctl          4  0/4/0/0.7.0  sctl      CLAIMED     DEVICE       Initiator
ext_bus      5  0/4/0/1      c720      CLAIMED     INTERFACE    SCSI C875 Fast
Wide Differential
target       8  0/4/0/1.7    tgt       CLAIMED     DEVICE
ctl          5  0/4/0/1.7.0  sctl      CLAIMED     DEVICE       Initiator
ba          5  0/5          lba      CLAIMED     BUS_NEXUS    Local PCI Bus Adapter (782)
ba          6  0/6          lba      CLAIMED     BUS_NEXUS    Local PCI Bus Adapter (782)
ba          7  0/7          lba      CLAIMED     BUS_NEXUS    Local PCI Bus Adapter (782)
ext_bus      6  0/7/0/0      c720      CLAIMED     INTERFACE
                                                      SCSI C875 Fast Wide Differential
target       9  0/7/0/0.7    tgt       CLAIMED     DEVICE
ctl          6  0/7/0/0.7.0  sctl      CLAIMED     DEVICE       Initiator
ext_bus      7  0/7/0/1      c720      CLAIMED     INTERFACE
                                                      SCSI C875 Fast Wide Differential
target      10  0/7/0/1.7    tgt       CLAIMED     DEVICE
ctl          7  0/7/0/1.7.0  sctl      CLAIMED     DEVICE       Initiator
memory       0  8            memory    CLAIMED     MEMORY       Memory
processor    0  160          processor CLAIMED     PROCESSOR    Processor
processor    1  166          processor CLAIMED     PROCESSOR    Processor
#
```

I normally run **ioscan** with the *-f* option because it includes the *Driver, S/W State,* and *H/W Type* columns. I am interested in the driver associated with the hardware in the system that the *-f* option produces.

The **ioscan** output shows all of the hardware that comprises the system, including the two processors in the system.

The file **/stand/vmunix** is the currently running kernel. Here is a long listing of the directory **/stand** on the L-Class system, which shows the file **/stand/vmunix**:

```
# ls -l
total 74274
-rw-r--r--   1 root    sys          19 Aug  4 11:37 bootconf
drwxr-xr-x   4 root    sys        2048 Aug 25 11:24 build
drwxr-xr-x   5 root    sys        1024 Aug 24 13:00 dlkm
drwxr-xr-x   5 root    sys        1024 Aug  4 12:45 dlkm.vmunix.prev
-rw-r--r--   1 root    sys        3024 Aug  4 12:26 ioconfig
-r--r--r--   1 root    sys          82 Aug  4 12:27 kernrel
```

```
drwxr-xr-x   2 root    sys       1024 Aug 29 11:39 krs
drwxr-xr-x   2 root    root      1024 Aug 29 11:33 krs_lkg
drwxr-xr-x   2 root    root      1024 Aug 29 11:39 krs_tmp
drwxr-xr-x   2 root    root      8192 Aug  4 11:36 lost+found
-rw-------   1 root    root        12 Aug 29 11:33 rootconf
-rw-rw-rw-   1 root    sys       1180 Aug 24 12:52 system
-r--r--r--   1 root    sys       1026 Aug  4 12:21 system.prev
-rwxr-xr-x   1 root    sys   14774416 Aug 24 12:53 vmunix
-rwxr-xr-x   1 root    sys   23184584 Aug  4 12:22 vmunix.prev
#
```

Notice that among the directories shown are two related to Dynamically Loadable Kernel Modules (DLKM). These are kernel modules that can be included in the kernel without having to reboot the system.

In order to make a change to the kernel, we would change to the **/stand/build** directory, where all work in creating a new kernel is performed, and issue the **system_prep** command as shown below:

```
# cd /stand/build
# /usr/lbin/sysadm/system_prep  -s  system
```

We can now proceed to make the desired changes to the kernel, including adding a driver or subsystem such as cdfs for a CD-ROM file system. With the dynamically loadable kernel module (DLKM) structure in place with 11i, we must use **kmsystem** and **kmtune** to make changes to the kernel system and system description files.

You can use **kmtune** to view the value and parameters related to existing kernel parameters as well as to make proposed modifications to the kernel. The following listing shows issuing **kmtune** (without the *-l* option to view details) to view a summary of the currently running kernel:

```
# kmtune

Parameter             Current Dyn Planned                    Module     Version
===============================================================================
NSTRBLKSCHED                -  -  2
NSTREVENT                  50  -  50
NSTRPUSH                   16  -  16
NSTRSCHED                   0  -  0
STRCTLSZ                 1024  -  1024
STRMSGSZ                65535  -  65535
acctresume                  4  -  4
acctsuspend                 2  -  2
aio_listio_max            256  -  256
aio_max_ops              2048  -  2048
aio_physmem_pct            10  -  10
```

```
aio_prio_delta_max          20  -  20
allocate_fs_swapmap          0  -  0
alwaysdump                   1  -  1
bootspinlocks                -  -  256
bufcache_hash_locks        128  -  128
bufpages                     0  -  0
chanq_hash_locks           256  -  256
create_fastlinks             0  -  0
dbc_max_pct                 50  -  50
dbc_min_pct                  5  -  5
default_disk_ir              0  -  0
desfree                      -  -  0
disksort_seconds             0  -  0
dnlc_hash_locks            512  -  512
dontdump                     0  -  0
dskless_node                 -  -  0
dst                          1  -  1
effective_maxpid             -  -  ((NPROC<22500)?30000:(NPROC*5/4))
eisa_io_estimate             -  -  0x300
enable_idds                  0  -  0
eqmemsize                   15  -  15
executable_stack             1  -  1
fcp_large_config             0  -  0
file_pad                     -  -  10
fs_async                     0  -  0
ftable_hash_locks           64  -  64
hdlpreg_hash_locks         128  -  128
hfs_max_ra_blocks            8  -  8
hfs_max_revra_blocks         8  -  8
hfs_ra_per_disk             64  -  64
hfs_revra_per_disk          64  -  64
hp_hfs_mtra_enabled          1  -  1
hpux_aes_override            -  -  0
initmodmax                  50  -  50
io_ports_hash_locks         64  -  64
iomemsize                    -  -  40000
ksi_alloc_max             2208  -  2208
ksi_send_max                32  -  32
lotsfree                     -  -  0
max_async_ports             50  -  50
max_fcp_reqs               512  -  512
max_mem_window               0  -  0
max_thread_proc             64  -  64
maxdsiz             0x10000000  -  0x10000000
maxdsiz_64bit       0x40000000  -  0X40000000
maxfiles                    60  -  60
maxfiles_lim              1024  Y  1024
maxqueuetime                 -  -  0
maxssiz               0x800000  -  0X800000
maxssiz_64bit         0x800000  -  0X800000
maxswapchunks              512  -  512
maxtsiz              0x4000000  Y  0X4000000
maxtsiz_64bit       0x40000000  Y  0x40000000
maxuprc                     77  Y  77
maxusers                    32  -  32
maxvgs                      10  -  10
mesg                         1  -  1
minfree                      -  -  0
modstrmax                  500  -  500
msgmap                      42  -  42
msgmax                    8192  Y  8192
msgmnb                   16384  Y  16384
msgmni                      50  -  50
msgseg                    2048  -  2048
msgssz                       8  -  8
msgtql                      40  -  40
nbuf                         0  -  0
ncallout                   515  -  515
ncdnode                    150  -  150
nclist                     612  -  612
ncsize                    5596  -  5596
ndilbuffers                 30  -  30
netisr_priority              -  -  -1
netmemmax                    -  -  0
nfile                      910  -  910
```

```
nflocks                    200  -  200
nhtbl_scale                  0  -  0
ninode                     476  -  476
nkthread                   499  -  499
nni                          -  -  2
no_lvm_disks                 0  -  0
nproc                      400  -  500
npty                        60  -  60
nstrpty                     60  -  60
nstrtel                     60  -  60
nswapdev                    10  -  10
nswapfs                     10  -  10
nsysmap                    800  -  800
nsysmap64                  800  -  800
num_tachyon_adapters         0  -  0
o_sync_is_o_dsync            0  -  0
page_text_to_local           -  -  0
pfdat_hash_locks           128  -  128
public_shlibs                1  -  1
region_hash_locks          128  -  128
remote_nfs_swap              0  -  0
rtsched_numpri              32  -  32
scroll_lines               100  -  100
scsi_maxphys           1048576  -  1048576
sema                         1  -  1
semaem                   16384  -  16384
semmap                      66  -  66
semmni                      64  -  64
semmns                     128  -  128
semmnu                      30  -  30
semume                      10  -  10
semvmx                   32767  -  32767
sendfile_max                 0  -  0
shmem                        1  -  1
shmmax               0x4000000  Y  0X4000000
shmmni                     200  -  200
shmseg                     120  Y  120
st_ats_enabled               1  -  1
st_fail_overruns             0  -  0
st_large_recs                0  -  0
streampipes                  0  -  0
swapmem_on                   1  -  1
swchunk                   2048  -  2048
sysv_hash_locks            128  -  128
tcphashsz                    0  -  0
timeslice                   10  -  10
timezone                   420  -  420
unlockable_mem               0  -  0
vas_hash_locks             128  -  128
vnode_cd_hash_locks        128  -  128
vnode_hash_locks           128  -  128
vps_ceiling                 16  -  16
vps_chatr_ceiling      1048576  -  1048576
vps_pagesize                 4  -  4
vx_fancyra_enable            0  -  0
vx_maxlink               32767  -  32767
vx_ncsize                 1024  -  1024
vxfs_max_ra_kbytes        1024  -  1024
vxfs_ra_per_disk          1024  -  1024
#
```

Issuing **kmtune** with the *-l* option produces a detailed listing of the kernel. The following shows just the output for one of the parameters:

```
# kmtune -l
Parameter:     maxuprc
Current:       77
Planned:       77
Default:       75
Minimum:       -
Module:        -
Version:       -
Dynamic:       Yes
#
```

This parameter is *Dynamic* (*Yes*) meaning that the kernel can be dynamically updated. After having viewed this output we can now modify the value of this dynamic parameter. The following command changes the value of the following parameter from *77*, which is the existing value, to *80*:

```
# kmtune -s maxuprc=80
#
```

We can now issue the **kmtune** to again view the existing and proposed value of the *maxuprc* parameter:

```
# kmtune
Parameter              Current Dyn Planned                    Module     Version
================================================================================
NSTRBLKSCHED                 -  -  2
NSTREVENT                   50  -  50
NSTRPUSH                    16  -  16
NSTRSCHED                    0  -  0
STRCTLSZ                  1024  -  1024
STRMSGSZ                 65535  -  65535
acctresume                   4  -  4
acctsuspend                  2  -  2
aio_listio_max             256  -  256
aio_max_ops               2048  -  2048
aio_physmem_pct             10  -  10
aio_prio_delta_max          20  -  20
allocate_fs_swapmap          0  -  0
alwaysdump                   1  -  1
bootspinlocks                -  -  256
bufcache_hash_locks        128  -  128
bufpages                     0  -  0
chanq_hash_locks           256  -  256
create_fastlinks             0  -  0
dbc_max_pct                 50  -  50
dbc_min_pct                  5  -  5
```

```
default_disk_ir              0  -  0
desfree                      -  -  0
disksort_seconds             0  -  0
dnlc_hash_locks            512  -  512
dontdump                     0  -  0
dskless_node                 -  -  0
dst                          1  -  1
effective_maxpid             -  -  ((NPROC<22500)?30000:(NPROC*5/4))
eisa_io_estimate             -  -  0x300
enable_idds                  0  -  0
eqmemsize                   15  -  15
executable_stack             1  -  1
fcp_large_config             0  -  0
file_pad                     -  -  10
fs_async                     0  -  0
ftable_hash_locks           64  -  64
hdlpreg_hash_locks         128  -  128
hfs_max_ra_blocks            8  -  8
hfs_max_revra_blocks         8  -  8
hfs_ra_per_disk             64  -  64
hfs_revra_per_disk          64  -  64
hp_hfs_mtra_enabled          1  -  1
hpux_aes_override            -  -  0
initmodmax                  50  -  50
io_ports_hash_locks         64  -  64
iomemsize                    -  -  40000
ksi_alloc_max             2208  -  2208
ksi_send_max                32  -  32
lotsfree                     -  -  0
max_async_ports             50  -  50
max_fcp_reqs               512  -  512
max_mem_window               0  -  0
max_thread_proc             64  -  64
maxdsiz             0x10000000  -  0x10000000
maxdsiz_64bit       0x40000000  -  0X40000000
maxfiles                    60  -  60
maxfiles_lim              1024  Y  1200
maxqueuetime                 -  -  0
maxssiz               0x800000  -  0X800000
maxssiz_64bit         0x800000  -  0X800000
maxswapchunks              512  -  512
maxtsiz              0x4000000  Y  0X4000000
maxtsiz_64bit       0x40000000  Y  0X40000000
maxuprc                     77  Y  80
maxusers                    32  -  32
maxvgs                      10  -  10
mesg                         1  -  1
minfree                      -  -  0
modstrmax                  500  -  500
msgmap                      42  -  42
msgmax                    8192  Y  8192
msgmnb                   16384  Y  16384
msgmni                      50  -  50
msgseg                    2048  -  2048
msgssz                       8  -  8
msgtql                      40  -  40
nbuf                         0  -  0
ncallout                   515  -  515
ncdnode                    150  -  150
nclist                     612  -  612
ncsize                    5596  -  5596
ndilbuffers                 30  -  30
netisr_priority              -  -  -1
netmemmax                    -  -  0
nfile                      910  -  910
nflocks                    200  -  200
nhtbl_scale                  0  -  0
ninode                     476  -  476
nkthread                   499  -  499
nni                          -  -  2
no_lvm_disks                 0  -  0
nproc                      400  -  400
npty                        60  -  60
nstrpty                     60  -  60
nstrtel                     60  -  60
```

```
nswapdev                  10  -  10
nswapfs                   10  -  10
nsysmap                  800  -  800
nsysmap64                800  -  800
num_tachyon_adapters       0  -  0
o_sync_is_o_dsync          0  -  0
page_text_to_local         -  -  0
pfdat_hash_locks         128  -  128
public_shlibs              1  -  1
region_hash_locks        128  -  128
remote_nfs_swap            0  -  0
rtsched_numpri            32  -  32
scroll_lines             100  -  100
scsi_maxphys         1048576  -  1048576
sema                       1  -  1
semaem                 16384  -  16384
semmap                    66  -  66
semmni                    64  -  64
semmns                   128  -  128
semmnu                    30  -  30
semume                    10  -  10
semvmx                 32767  -  32767
sendfile_max               0  -  0
shmem                      1  -  1
shmmax             0x4000000  Y  0X4000000
shmmni                   200  -  200
shmseg                   120  Y  120
st_ats_enabled             1  -  1
st_fail_overruns           0  -  0
st_large_recs              0  -  0
streampipes                0  -  0
swapmem_on                 1  -  1
swchunk                 2048  -  2048
sysv_hash_locks          128  -  128
tcphashsz                  0  -  0
timeslice                 10  -  10
timezone                 420  -  420
unlockable_mem             0  -  0
vas_hash_locks           128  -  128
vnode_cd_hash_locks      128  -  128
vnode_hash_locks         128  -  128
vps_ceiling               16  -  16
vps_chatr_ceiling    1048576  -  1048576
vps_pagesize               4  -  4
vx_fancyra_enable          0  -  0
vx_maxlink             32767  -  32767
vx_ncsize               1024  -  1024
vxfs_max_ra_kbytes      1024  -  1024
vxfs_ra_per_disk        1024  -  1024
#
```

This output shows that the change to our parameter is pending.

We can apply the change to the dynamic parameter *maxuprc* from *77* to *80* by issuing **kmtune** with the *-u* option:

```
# kmtune -u
The kernel's value of maxuprc has been set to 80 (0x50).
#
```

This output shows that the change we wanted made to the kernel has been made. We can confirm this by running **kmtune** again and searching for *maxuprc*:

```
# kmtune | grep maxuprc
maxuprc                    80  Y  80
#
```

Both the *Current* and *Planned* values have been updated to *80*. This dynamic update can be done using **kmsystem** to add dynamic drivers to your kernel.

There are many other procedures for which you would have to perform additional steps to include modifications in the kernel and rebuild it. With these non-dynamic changes you would create a new kernel, which will be generated as **/stand/build/vmunix_test,** using the command shown below:

```
# mk_kernel -s system
Compiling conf.c...
Loading the kernel...
Generating kernel symbol table...
#
```

At this point, the new kernel exists in the **/stand/build** directory. The existing kernel is updated with the newly generated kernel with **kmupdate**. **kmupdate** moves the new kernel files into the **/stand** directory. I would first recommend moving the existing **/stand/system** kernel file to a backup file, and then updating the new kernel as shown below:

```
# mv /stand/system /stand/system.prev      (may want to move additional
# kmupdate /stand/build/vmunix_test        files shown in Figure 4-1)

  Kernel update request is scheduled.

  Default kernel /stand/vmunix will be updated by
  newly built kernel /stand/build/vmunix_test
  at next system shutdown or startup time.
#
```

kmupdate will automatically create backup copies of **/stand/vmunix** and **/stand/dlkm** for you. These will be created as **/stand/vmunix.prev** and **/stand/dlkm.vmunix.prev,** respectively.

You can now shut down the system and automatically boot from the new kernel if your update did not take place dynamically and requires a reboot.

Figure 4-1 summarizes the process of building a new kernel in HP-UX 11i.

Step	Comments
1) run **sysdef** and **ioscan -f**	Analyzes and reports tunable parameters of currently running kernel.
2) perform long listing of **/stand** directory	The file **vmunix** is the existing kernel, and **system** is used to build a new kernel.
3) **cd /stand/build**	This is the directory where the new kernel will be built.
4) **/usr/lbin/sysadm/system_prep -s system**	This extracts the **system** file from the currently running kernel.
5) use **kmsystem** and **kmtune** to make changes	Takes place in the **/stand/build** directory. Dyamic update complete here.
6) **mk_kernel -s system**	Makes a new kernel in the **/stand/build** directory called **vmunix_test**. DLKM files are produced in **dlkm.vmunix_test/***.
7) **mv /stand/system /stand/system.prev** **mv /stand/vmunix /stand/vmunix.prev** **mv /stand/dlkm /stand/dlkm.vmunix.prev**	Saves the existing files as **.prev**.
8) **mv /stand/build/system /stand/system** **kmupdate /stand/build/vmunix_test**	Updates the kernel with the newly generated kernel. Automatically saves the old versions in **/stand** as follows: **vmunix** as **/stand/vmunix.prev** **dlkm** as **/dlkm.vmunix.prev**
9) **cd /** **shutdown -r 0**	Changes directory to **/** and shuts down the sytem so that it comes up with the new kernel. This may not be required if your change could be implemented dynamically.

Figure 4-1 Creating a Kernel in HP-UX 11i

There are really two different procedures for generating your kernel - one for dynamic elements, such as the parameter *maxuprc* shown in the earlier example, and one for static elements. The static procedure consists of several additional steps and a reboot. With HP-UX 11i, more and more kernel objects will be updated dynamically, resulting in fewer reboots when modifying your kernel.

kcweb

At the time of this writing **kcweb** is a stand-alone tool that is downloaded from *www.software.hp.com*. Future plans are for **kcweb** to be included with HP-UX distributions and for additional Web-based management to be part of HP-UX. At this time the tool is simple to download and install.

In this section we are able to perform a variety of functions through the Web-based interface. In this section we'll perform the following in **kcweb**:

- View kernel parameters
- Get details on a specific kernel parameter in the bottom of the **kcweb** page and the man page.
- Modify a dynamic kernel parameter and apply the new value.
- Set an alarm to inform us when a kernel parameter exceeds the specified value.

At the time of this writing, **kcweb** is invoked at the command line with **kcweb**. On my system this opens the browser window shown in Figure 4-2. Figure 3-2 shows **kcweb** with several kernel parameters.

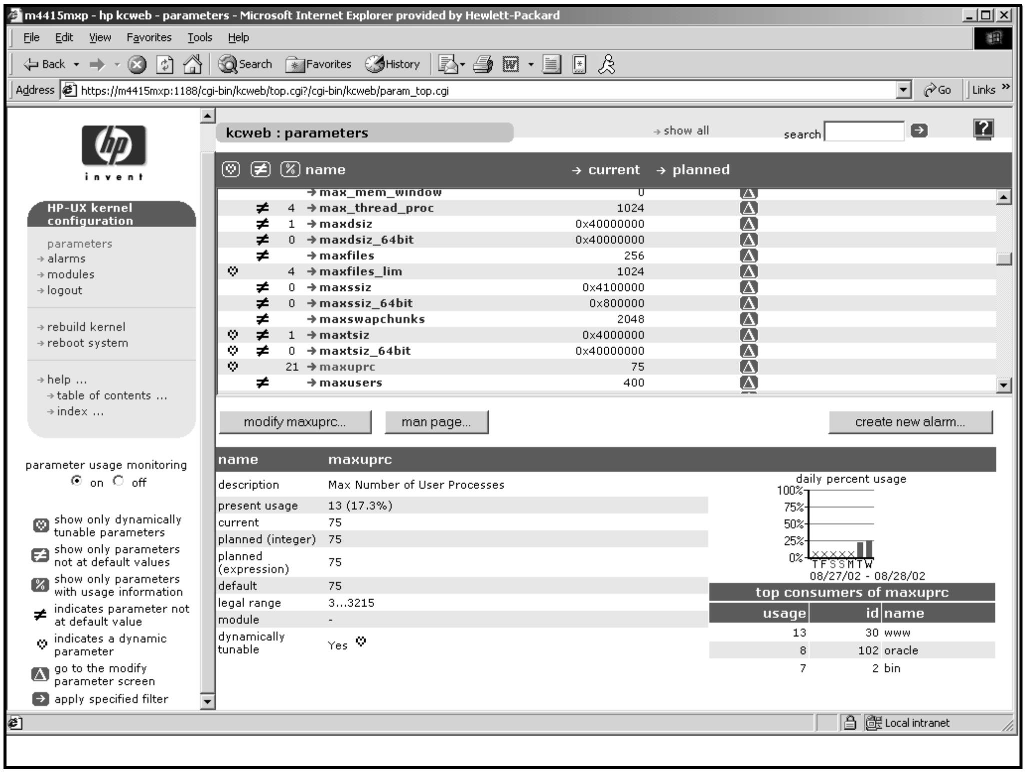


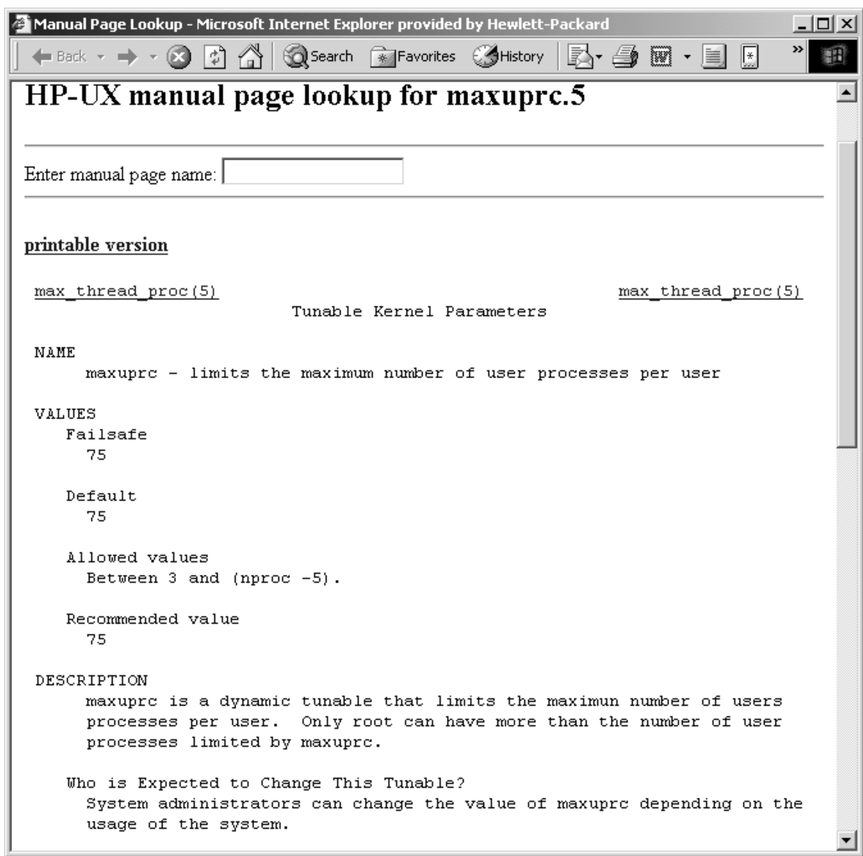
Figure 4-2 **kcweb** Showing a Variety of Parameters

The upper left of Figure 4-2 shows that **kcweb** has *parameters, alarms,* and *modules* functions. We'll cover an example of working with *parameters* and *alarms* in this section. There aren't too many dynamically loadable kernel modules at this time but the technique for working with dynamically loadable modules is similar to that of dynamically tunable parameters so the examples will give you a good idea of the way in which you work with **kcweb**.

Notice in the bottom left of the figure that there is a legend that includes descriptions of the symbols that are used in **kcweb**. Those parameters with a heart next to them are dynamically tunable parameters. If you select the heart on the bar across the top of the kernel parameters, then only dynamically tunable parameters will be shown. The "not equal to" sign indicates parameters that are not set to their default value. There are several other entries in the legend as well. This makes for viewing groups of icons easy and the legend helps identify the status of icons.

The bottom of the screen provides information about the kernel parameter selected: in this case *maxuprc*. There is a graph in the bottom right showing the usage of this parameter over time. For system-wide parameters the graph will show usage on a system basis. For user-specific or process-specific parameters, such as *maxuprc*, the graph includes the top five consumers of the parameter.

You can get detailed information about a kernel parameter by selecting the *man page...* button as shown in Figure 4-3:



HP-UX manual page lookup for maxuprc.5

Enter manual page name:

printable version

```
max_thread_proc(5)                                        max_thread_proc(5)
                          Tunable Kernel Parameters

NAME
     maxuprc - limits the maximum number of user processes per user

VALUES
   Failsafe
     75

   Default
     75

   Allowed values
     Between 3 and (nproc -5).

   Recommended value
     75

DESCRIPTION
     maxuprc is a dynamic tunable that limits the maximun number of users
     processes per user.  Only root can have more than the number of user
     processes limited by maxuprc.

   Who is Expected to Change This Tunable?
     System administrators can change the value of maxuprc depending on the
     usage of the system.
```

Figure 4-3 **kcweb** Showing *man page...* For *maxuprc*

You can also modify one of the parameters by highlighting the parameter and then selecting the *modify <parameter name>* as we've done for the *maxuprc* parameter as we've done in Figure 4-4:

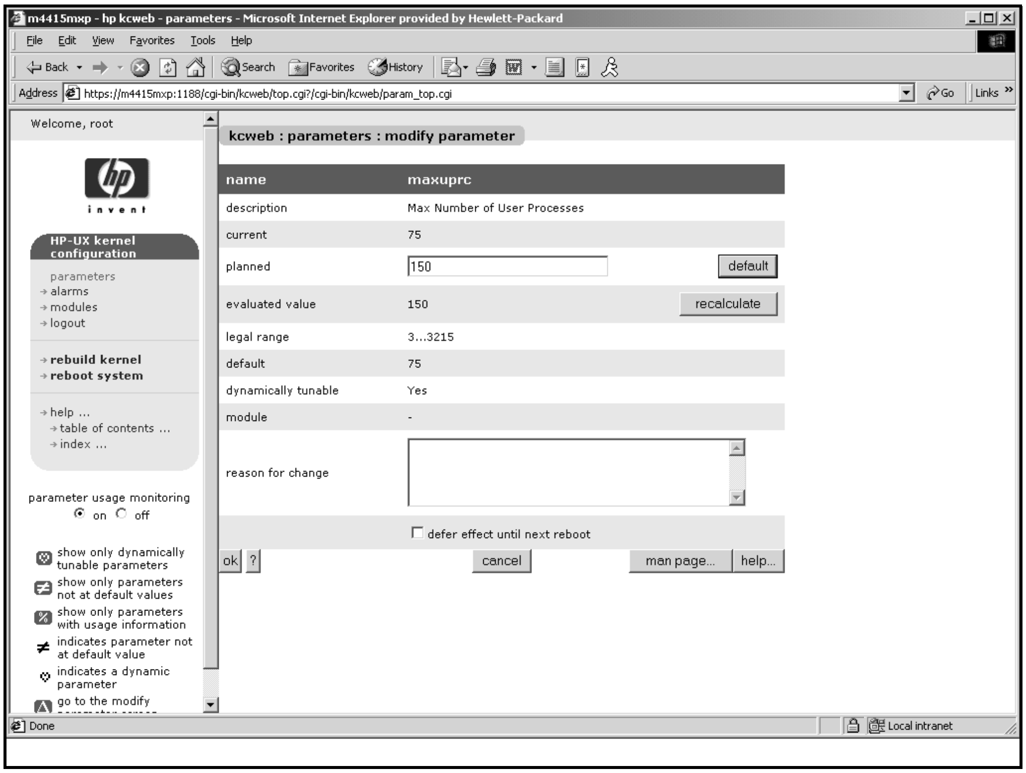


Figure 4-4 **kcweb** Showing *modify maxuprc*

We've chosen to increase *maxuprc* from *75* to *150* in Figure 4-4. To implement the change we unselect the *defer until next reboot* box and select *ok*. The change is then implemented as you can see in Figure 4-5:

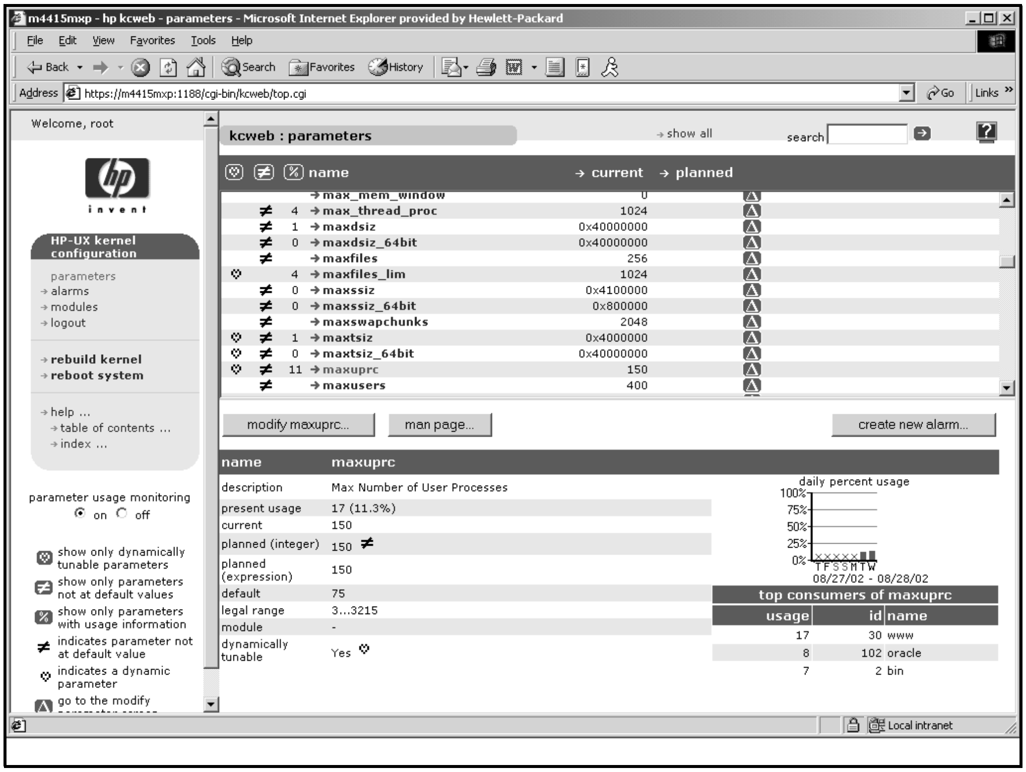


Figure 4-5 **kcweb** Showing *maxuprc planned* Change

This is a parameter that can be modified dynamically, so it is updated immediately. If this were not a dynamic parameter, we could *rebuild kernel* to update the kernel with the desired change.

Now that we have modified *maxuprc* we can set an alarm to inform us when the parameter reaches a specified threshold. Figure 4-6 shows setting up this alarm:

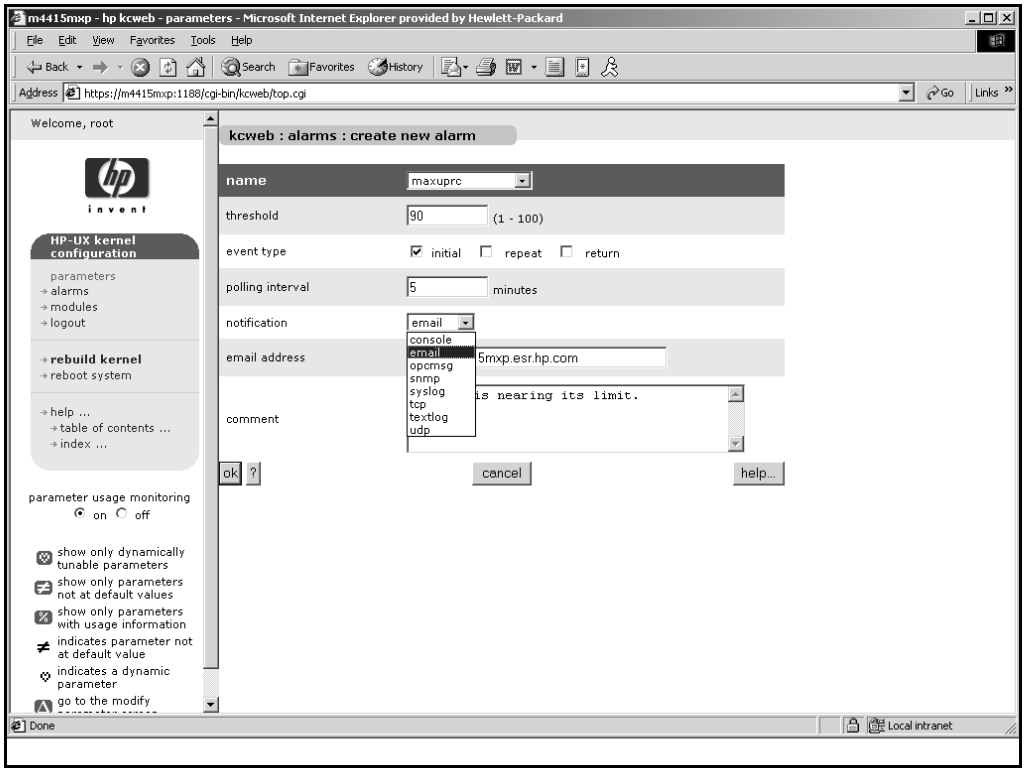


Figure 4-6 **kcweb** Showing *maxuprc* Alarm

We get to the *kcweb:alarms* page by selecting *create new alarm...* in the window shown earlier. All of the parameters related to the alarm are shown in Figure 4-6. The setup of the alarm specifies a *threshold* of *90*. All of the options for notification are shown. In this case we've selected email to *root@m4415mxp.esr.hp.com* and specified a comment to appear in the email address.

We can also work with kernel modules in the same way that we work with kernel parameters. Those that are dynamic can be loaded on-the-fly, and those that are not dynamic can be built into the kernel with a rebuild.

This was a quick overview of **kcweb** that included some of the most commonly performed tasks. Since this is a Web-based interface, it is easy to use and most of the screens and information are self explanatory. More Web based management tools will be included in HP-UX over time.