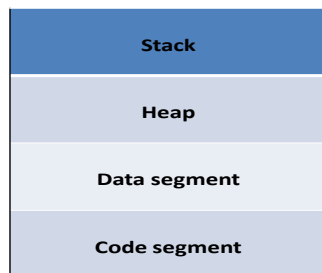


Process Control Block (PCB) - Class Notes

- Every process has an address space in user mode and PCB in kernel mode as shown below.
- User mode address space consists of code segment, data segment, heap and stack.



- **PCB** is a kernel structure whose instance is created for each process to store process information.
- PCB is created when the process is loaded and de-allocated when the application is fully terminated
- Linux kernel uses a structure called **task_struct** to implement process control block (PCB). This structure can be found in the Linux source code

`$vim include/linux/sched.h`

Important elements of PCB:

- **task_struct** contains following
 - Process identification information (PID ,PPID ,CPID)
 - Process scheduling policy and state
 - Information about number of user level threads mapped to the process.
- **mm_struct** gives process address space information
- **tty_struct** gives the active terminal information

- **files_struct** gives the information about the files which the application currently using. It is called file descriptor table.
 - **signal_struct** gives the signal information.
- Users can see PCB through **/proc** directory. In order to understand more let's run a simple C file and execute the output. We use a `getchar()` to put it in a wait state.

```
veda@veda: ~  
veda@veda: ~  
#include<stdio.h>  
  
int main()  
{  
    int a = 400;  
    printf("%d\n", a);  
    getchar();  
    return 0;  
}  
~  
~  
~
```

Compile and run the above program.

```
veda@veda: ~  
veda@veda: ~  
veda@veda:~$ gcc test1.c -o test1  
veda@veda:~$ ./test1  
400  
█
```

- **Ps -Af** is the command used to see the current running process in the kernel. It also shows the PID of the processes. It gives the **PID** of our program **test1**

```
veda@veda: ~
veda@veda: ~
veda@veda: /proc/2196

veda 1472 1 0 15:05 ? 00:00:00 /usr/lib/bamf/bamfdemon
veda 1473 1299 0 15:05 ? 00:00:00 telepathy-indicator
veda 1475 1 0 15:05 ? 00:00:00 /usr/lib/telepathy/mission-control-5
veda 1480 1 1 15:05 ? 00:01:02 /usr/lib/unity/unity-panel-service
veda 1488 1 0 15:05 ? 00:00:21 /usr/lib/indicator-application/indicator-application-service
veda 1489 1 0 15:05 ? 00:00:00 /usr/lib/indicator-session/indicator-session-service
veda 1491 1 0 15:05 ? 00:00:00 /usr/lib/indicator-sound/indicator-sound-service
veda 1496 1 0 15:05 ? 00:00:00 /usr/lib/indicator-messages/indicator-messages-service
veda 1497 1 0 15:05 ? 00:00:00 /usr/lib/indicator-datetime/indicator-datetime-service
veda 1517 1 0 15:05 ? 00:00:00 /usr/lib/geoclue/geoclue-master
veda 1527 1299 0 15:05 ? 00:00:00 /usr/lib/gnome-disk-utility/gdu-notification-daemon
veda 1529 1299 0 15:05 ? 00:00:00 zeitgeist-datahub
veda 1535 1 0 15:05 ? 00:00:00 /usr/bin/python /usr/bin/zeitgeist-daemon
veda 1536 1535 0 15:05 ? 00:00:00 /bin/cat
veda 1547 1299 0 15:05 ? 00:00:10 /usr/bin/python /usr/share/system-config-printer/applet.py
veda 1559 1 0 15:05 ? 00:00:07 /usr/bin/python /usr/lib/ubuntuone-client/ubuntuone-syncdaemon
veda 1588 1299 0 15:06 ? 00:00:00 update-notifier
root 1603 1 0 15:06 ? 00:00:00 /usr/bin/python /usr/lib/system-service/system-service-d
veda 1617 1 0 15:06 ? 00:00:00 /usr/lib/unity-lens-applications/unity-applications-daemon
veda 1619 1 0 15:06 ? 00:00:00 /usr/lib/unity-lens-music/unity-music-daemon
veda 1621 1 0 15:06 ? 00:00:00 /usr/lib/unity-lens-files/unity-files-daemon
veda 1651 1 0 15:06 ? 00:00:00 /usr/lib/unity-lens-music/unity-musicstore-daemon
veda 1690 1299 0 15:07 ? 00:00:00 /usr/lib/deja-dup/deja-dup/deja-dup-monitor
veda 1701 1395 0 15:09 ? 00:00:00 /bin/sh -c gnome-terminal
veda 1702 1701 0 15:09 ? 00:00:03 gnome-terminal
veda 1708 1702 0 15:09 ? 00:00:00 gnome-pty-helper
veda 1709 1702 0 15:09 pts/0 00:00:00 bash
root 1954 2 0 15:49 ? 00:00:00 [kworker/0:2]
root 1970 2 0 16:00 ? 00:00:00 [kworker/0:0]
root 1971 2 0 16:01 ? 00:00:00 [kworker/u:1]
root 1987 2 0 16:07 ? 00:00:00 [kworker/u:2]
root 2078 2 0 16:10 ? 00:00:00 [kworker/1:0]
root 2079 967 0 16:11 ? 00:00:00 /sbin/dhclient -d -4 -sf /usr/lib/NetworkManager/nm-dhcp-client.action -pf /var/run/dhclient-eth0.pid
root 2188 2 0 16:14 ? 00:00:00 [kworker/0:1]
veda 2197 1702 0 16:15 pts/1 00:00:00 bash
veda 2252 1702 0 16:15 pts/2 00:00:00 bash
veda 2342 1709 0 16:17 pts/0 00:00:00 ./test1
veda 2363 2197 0 16:17 pts/1 00:00:00 ps -Af
veda@veda:~$
```

➤ Then go to **/proc**, there we will find our **pid**.

```
veda@veda: /proc
veda@veda: ~
veda@veda: /proc

veda@veda:/$ cd /proc
veda@veda:/proc$ ls
1 1039 13 1405 1438 1489 1588 1709 2188 26 6 924 bus execdomains kpagecount pagetypeinfo sysvipc
1000 1045 1350 1407 1441 1491 16 18 2197 27 620 934 cgroups fb kpageflags partitions timer_list
1003 1056 1353 1408 1452 1496 1603 19 2252 28 630 949 cmdline filesystems latency_stats sched_debug timer_stats
1006 1061 1359 1410 1457 1497 1617 1954 23 284 635 967 consoles fs loadavg schedstat tty
1007 1068 1361 1414 1459 15 1619 1970 2342 285 664 970 cpuinfo interrupts locks scsi uptime
1013 11 1366 1415 1468 1517 1621 1971 235 29 7 976 crypto iomem mdstat self version
1014 1172 1381 1420 1469 1527 1651 1987 237 3 801 977 devices ioports meminfo slabinfo version_signature
1016 12 1389 1422 1472 1529 1690 2 2383 337 803 996 device-tree irq misc softirqs vmallocinfo
1020 1227 1391 1424 1473 1535 17 20 24 341 864 997 diskstats kallsyms modules stat vmstat
1021 1258 1395 1428 1475 1536 1701 2078 241 37 865 acpi dma kcore mounts swaps zoneinfo
1027 1289 1397 1432 1480 1547 1702 2079 243 38 897 asound dri key-users mtrr sys
1028 1299 1400 1433 1488 1559 1708 21 25 572 9 buddyinfo driver kmsg net sysrq-trigger
veda@veda:/proc$
```

- If we go to our particular pid, it shows **task_struct** tree as shown below.

```
veda@veda: /proc/2342
veda@veda: ~
veda@veda: /proc/2342$ ls
attr      clear_refs  cpuset  fd      limits  mountinfo  ns      pagemap  schedstat  stack  syscall
autogroup cmdline    cwd     fdinfo  loginuid  mounts     oom_adj  personality seccomp_filter stat   task
auxv      comm       environ io      maps     mountstats oom_score  root     sessionid  statm  wchan
cgroup    coredump_filter exe      latency mem      net        oom_score_adj sched    smaps    status
veda@veda: /proc/2342$
```

- In **PID** directory we can see a file maps, if we execute **\$cat maps**, which shows **mm_struct** i.e. it shows the addresses for the particular process as shown below. It also shows the addresses of libraries linked to it.

```
veda@veda: /proc/2342$ cat maps
00217000-00393000 r-xp 00000000 08:09 3145877 /lib/i386-linux-gnu/libc-2.13.so
00393000-00395000 r--p 0017c000 08:09 3145877 /lib/i386-linux-gnu/libc-2.13.so
00395000-00396000 rw-p 0017e000 08:09 3145877 /lib/i386-linux-gnu/libc-2.13.so
00396000-00399000 rw-p 00000000 00:00 0
006f7000-006f8000 r-xp 00000000 00:00 0 [vdso]
00dec000-00e0a000 r-xp 00000000 08:09 3149854 /lib/i386-linux-gnu/ld-2.13.so
00e0a000-00e0b000 r--p 0001d000 08:09 3149854 /lib/i386-linux-gnu/ld-2.13.so
00e0b000-00e0c000 rw-p 0001e000 08:09 3149854 /lib/i386-linux-gnu/ld-2.13.so
08048000-08049000 r-xp 00000000 08:09 4459559 /home/veda/test1
08049000-0804a000 r--p 00000000 08:09 4459559 /home/veda/test1
0804a000-0804b000 rw-p 00001000 08:09 4459559 /home/veda/test1
b7723000-b7724000 rw-p 00000000 00:00 0
b7733000-b7737000 rw-p 00000000 00:00 0
bfd88000-bfda9000 rw-p 00000000 00:00 0 [stack]
veda@veda: /proc/2342$
```

- Thus **/proc** is a directory which shows the process table and addresses mapped to a particular process.