

Tarea



Estructuras del S.O.

Alumno

César Mauricio Arellano Velásquez

Profesor

René Mac Kinney Romero

Materia

Sistemas Operativos

Desarrollo

Investigue cómo hacer las siguientes actividades en *linux*

- Obtener el tiempo que dura un pedazo de código
- Leer la estructura de un directorio

Investigue cual es el contenido de un *PCB* en linux

El PCB es creado por el SO cada vez que aparece un nuevo proceso. Los procesos son reconocidos por el sistema operativo y por lo tanto son aptos para la asignación de los recursos del sistema, sólo cuando existe un PCB activo asociado a ellos. Cuando el programa termina, el PCB es eliminado, dejando espacio libre en el registro, donde ahora pueden almacenarse otros PCB.

En el kernel de Linux, esto está implementado a través de una estructura de datos denominada **task_struct**, la cual es una **lista doblemente enlazada**, cuyo encabezado es **init_task (pid 0)**. Esto se conoce comúnmente como la **tabla de procesos**. Es el PCB de Linux, en ella se almacena toda la información relacionada con un proceso.

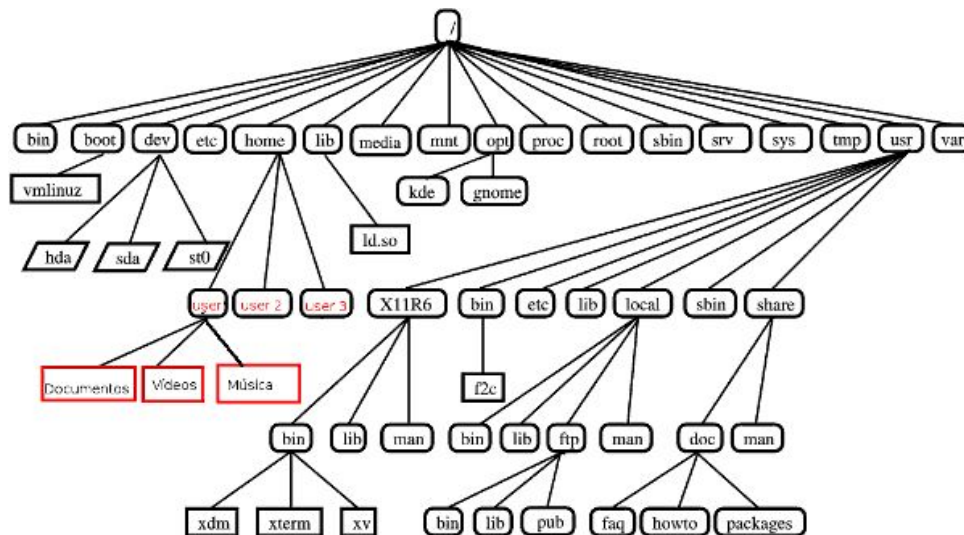
Los elementos de información asociados son:

- Estado actual del proceso.
- Contador de programa: indica la dirección de la siguiente instrucción que se ejecutará de ese proceso.
- Registros de CPU: acumuladores, registros índice, punteros de pila y registros generales.
- Información de planificación de CPU: prioridad del proceso, punteros a colas de planificación, etc.
- Información de gestión de memoria: valor de los registros de base y límite, tabla de páginas o tabla de segmentos.

| Puntero | Estado |
|--------------------------|--------|
| Identificador de proceso | |
| Contador de programa | |
| Registros | |
| Límites de memoria | |
| Estado de la E/S ... | |

- Información de contabilidad: tiempo de CPU, tiempo consumido, números de procesos, etc.
- Información de estado de E/S: dispositivos de E/S asignados a este proceso, lista de archivos abiertos, etc.

Investigue el contenido y la estructura de los directorios



/dev

Este directorio incluye todos los dispositivos de almacenamiento, **en forma de archivos**, conectados al sistema, es decir, cualquier disco duro conectado, partición, memoria USB, o CD ROM conectado al sistema y que el sistema pueda entender como un volumen lógico de almacenamiento.

Siendo esto así, verás que la ruta en la que se encuentra cualquier volumen (partición o dispositivo externo) conectado al sistema siempre empieza por **/dev**. Este es el directorio que contiene solo los puntos de montaje, pero no la información real de estos volúmenes.

Con el comando **sudo fdisk -l**, se puede visualizar la estructura de particiones del sistema. En una instalación típica de cualquier distro GNU/Linux suele ser la siguiente:

- /dev/sda1 - Partición principal

- /dev/sda2 - Partición extendida
- /dev/sda5 - Partición Swap

Eso en cuanto a particiones. Si se trata de un dispositivo externo, el volumen estará igualmente dentro de **/dev**, pero en este caso varía el nombre que el sistema le asigna a dicho volumen. Generalmente la estructura suele ser la siguiente.

- /dev/sdb1
- /dev/sdb2
- /dev/sdb3 ...

/proc

Este directorio contiene información de los procesos y aplicaciones que se están ejecutando en un momento determinado en el sistema, pero realmente no guarda nada como tal, ya que lo que almacena son archivos virtuales, por lo que el contenido de este directorio es nulo. Básicamente son listas de eventos del sistema operativo que se generan en el momento de acceder a ellos, y que no existen dentro del directorio como tal.

El directorio /proc en sí mismo es creado cada vez que arrancas tu equipo. Se requiere ser root para poder examinar completamente el directorio en su totalidad; algunos de los archivos (tales como los relacionados a procesos) son propiedad del usuario que los ejecutó.

Organización del directorio /proc

```
#> ls /proc
1      2432 3340 3715 3762 5441 815      devices  modules
129    2474 3358 3716 3764 5445 acpi     diskstats mounts
1290   248  3413 3717 3812 5459 asound  dma      mtrr
133    2486 3435 3718 3813 5479 bus     execdomains partitions
1420   2489 3439 3728 3814 557  dri     fb        self
165    276  3450 3731 39  5842 driver  filesystems slabinfo
166    280  36  3733 3973 5854 fs      interrupts splash
2      2812 3602 3734 4  6  ide     iomem     stat
2267   3  3603 3735 40  6381 irq     ioports   swaps
2268   326  3614 3737 4083 6558 net     kallsyms  sysrq-trigger
2282   327  3696 3739 4868 6561 scsi    kcore     timer_list
2285   3284 3697 3742 4873 6961 sys     keys      timer_stats
2295   329  3700 3744 4878 7206 sysvipc  key-users uptime
2335   3295 3701 3745 5  7207 tty     kmsg      version
2400   330  3706 3747 5109 7222 buddyinfo loadavg   vmcore
2401   3318 3709 3749 5112 7225 cmdline  locks     vmstat
2427   3329 3710 3751 541  7244 config.gz meminfo   zoneinfo
2428   3336 3714 3753 5440 752  cpuinfo  misc
```

Bibliografía

Escuela Universitaria de Informática de Segovia (2015), Gestión de Procesos, [Unidad 2: Gestión de Procesos \(uva.es\)](#) [1].

Flup (2014), ¿Dónde está el PCB en Linux?, [Process - Where is PCB on Linux - Unix & Linux Stack Exchange](#) [2].

Universidad de Sevilla (2020), Bloque de control de procesos, [Bloque de control de procesos - Wiki de Sistemas Operativos \(us.es\)](#) [3].

[ComputerNewAge](#) (2014), El Árbol de Directorios de Linux, [El Árbol de Directorios de Linux. Conoce las Principales Carpetas – Computer New Age](#) [4].

Binh Nguyen(2004), Linux Filesystem Hierarchy, [Linux Filesystem Hierarchy \(tldp.org\)](#) [5].