



**INSTITUTO POLITÉCNICO NACIONAL
ESCUELA SUPERIOR DE COMPUTO**

UNIDAD DE APRENDIZAJE: SISTEMAS OPERATIVOS

PRACTICA 06: “MAPAS DE MEMORIA DE UN PROCESO”

INTEGRANTES:

| NO. LISTA | NOMBRE |
|-----------|---------------------------------|
| 3 | CADENAS ACEVEDO JESÚS ALEJANDRO |
| 8 | FARRERA MENDEZ EMMANUEL SINAI |
| 27 | RAMOS GÓMEZ SERGIO |
| 28 | RENDON LECHUGA DIEGO |
| 32 | SERRANO GUILLÉN IVÁN URIEL |
| 34 | VALLEJO ACOSTA ANA PAULINA |

PROFESOR: ARAUJO DIAZ DAVID

GRUPO: 4CM3

FECHA DE ENTREGA: 09 MAYO 2023

Indice

OBJETIVO 3

DESARROLLO. 3

 Procesos en LINUX..... 3

 Procesos en UNIX 4

CONCLUSIONES. 8

REFERENCIAS..... 8

OBJETIVO

Comprender la forma como se almacena los procesos en memoria dentro de los sistemas LINUX/UNIX

DESARROLLO.

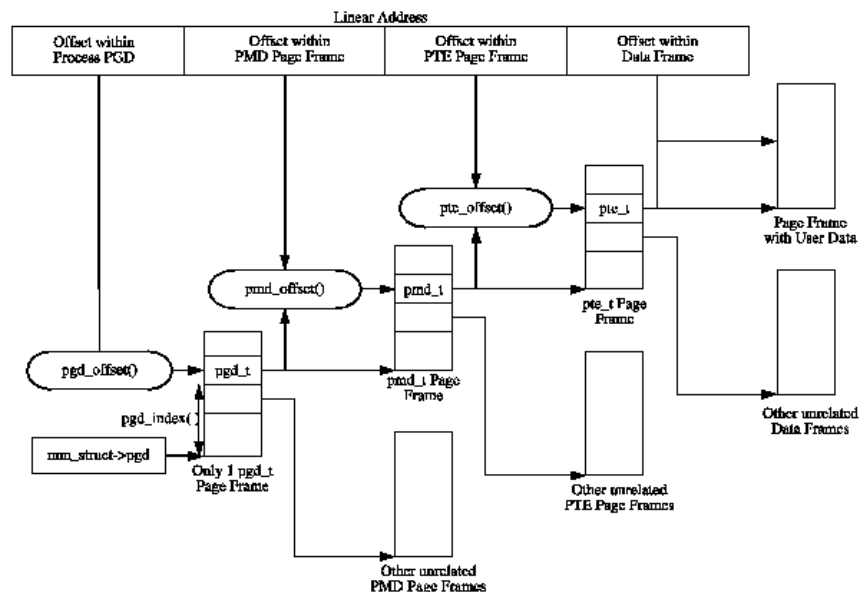
Procesos en LINUX

Linux emplea direcciones virtuales en lugar de direcciones físicas para aislar el espacio de direcciones de cada proceso y brindar mayor seguridad. Pero, en última instancia, acceder a la memoria física requerirá que el controlador de memoria disponga de una dirección física para acceder a la memoria física.

Para lograr esto, Linux almacena una tabla de asignaciones que contiene asignaciones de memoria virtual a física de cada proceso. Pero esto no es un simple mapeo uno a uno ya que requeriría mucho espacio en la memoria física. En su lugar, Linux utiliza un enfoque de tabla de 3 páginas. Los bits de dirección virtual están separados en cuatro campos distintos, cada uno de los cuales ayuda a navegar por las estructuras de la tabla de páginas. Las 3 tablas de páginas son:

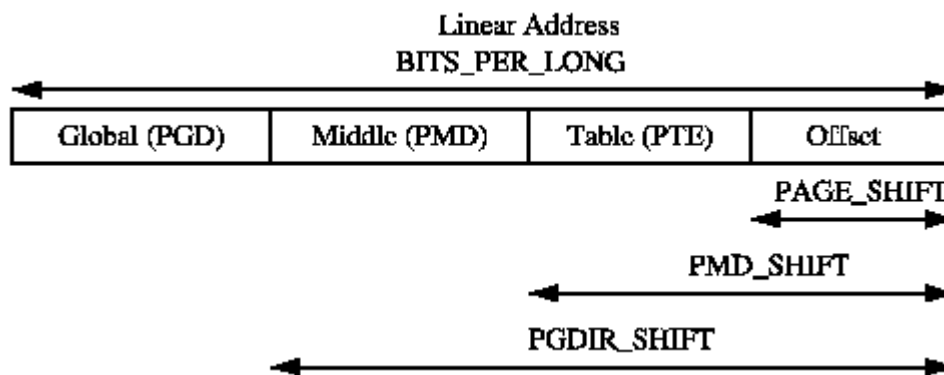
- Directorio global de páginas (**PGD**)
- Directorio medio de páginas (**PMD**)
- Entradas de la tabla de páginas (**PTE**)

El siguiente diagrama ayuda a explicar la estructura de la tabla de mejor manera



PGD es una matriz de tipo `pgd_t`, PMD es una matriz de tipo `pmd_t` y PTE es una matriz de tipo `pte_t`. Cada proceso está representado por una estructura `task_struct` que contiene un puntero a `mm_struct`, otra estructura que contiene información sobre las áreas de memoria virtual del proceso. Dentro de `mm_struct` existe un puntero a su propio directorio global de página. Antes de leer las tablas de páginas, se adquiere el miembro de bloqueo de la tabla de páginas de `mm_struct` mediante la API `spinlock`.

Cualquier dirección lineal dada puede dividirse en partes para generar compensaciones dentro de estos tres niveles de tabla de páginas y una compensación dentro de la página real. Para ayudar a dividir la dirección lineal en sus componentes, se proporcionan varias macros en tripletes para cada nivel de la tabla de páginas, a saber, una macro `SHIFT`, `SIZE` y `MASK`. Las macros `SHIFT` especifican la longitud en bits que son mapeados por cada nivel de las tablas de páginas como se ilustra en la imagen:



Como se mencionó, cada entrada se describe mediante las estructuras `pte_t`, `pmd_t` y `pgd_t` para PTE, PMD y PGD, respectivamente. Aunque a menudo estos son solo números enteros sin signo, se definen como estructuras por dos razones. El primero es para la protección de tipos para que no se utilicen de manera inapropiada. El segundo es para funciones como PAE en x86, donde se usan 4 bits adicionales para direccionar más de 4 GiB de memoria. Para almacenar los bits de protección, se define `pgprot_t` que contiene los indicadores relevantes y generalmente se almacena en los bits inferiores de una entrada de la tabla de páginas.

Procesos en UNIX

Las páginas en un modelo de memoria no son más que la memoria física que se divide en bloques de igual tamaño. Estos bloques se denominan marcos y la memoria lógica se divide en bloques de igual tamaño que se denominan páginas.

Los tamaños de página generalmente oscilan entre 512 bytes y 4K bytes. La paginación evita la fragmentación y la necesidad de compactación.

Cuando el núcleo decide establecer páginas físicas de memoria en una región, no es necesario asignar las páginas de forma continua o en un orden específico. Es lo mismo que los bloques de disco que no se asignan de forma contigua para reducir la fragmentación.

El kernel forma un mapeo de números de página lógicos a físicos en una tabla que se puede ver de la siguiente manera:

| Logical Page Number | Physical Page number |
|---------------------|----------------------|
| 0 | 177 |
| 1 | 17 |
| 2 | 50 |
| 3 | 100 |

Fig. Page table

Las tablas anteriores se conocen como tablas de página. La entrada de la tabla de regiones tiene punteros a las tablas de páginas. Entonces, el espacio de direcciones lógicas es contiguo y proporciona el índice para una matriz de números de página físicos.

Las tablas de páginas también contienen información relacionada con el hardware, como los permisos para las páginas. Hoy en día, existe un hardware especial para traducir la dirección porque la implementación del software de dicha traducción sería demasiado lenta. Por lo tanto, cuando un proceso comienza a ejecutarse, el kernel le dice al hardware dónde residen sus tablas de páginas. El mecanismo de paginación mediante el uso de hardware se explica más claramente en el siguiente diagrama.

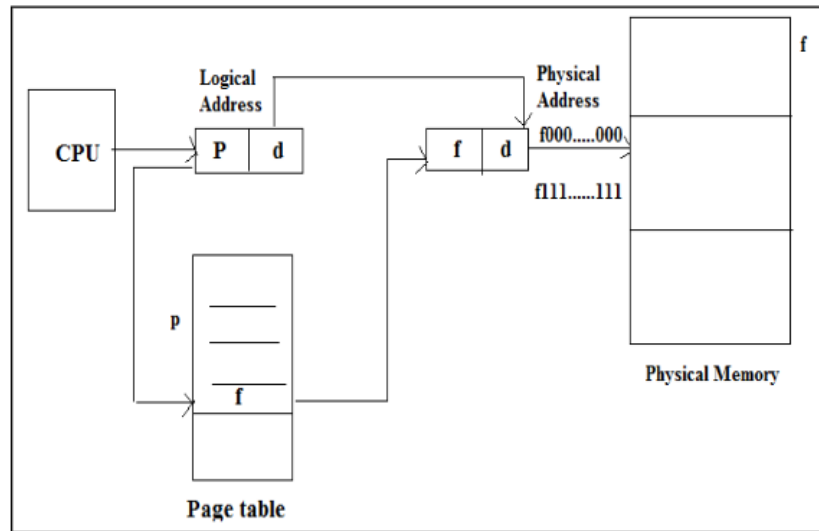


Fig. Paging Using hardware

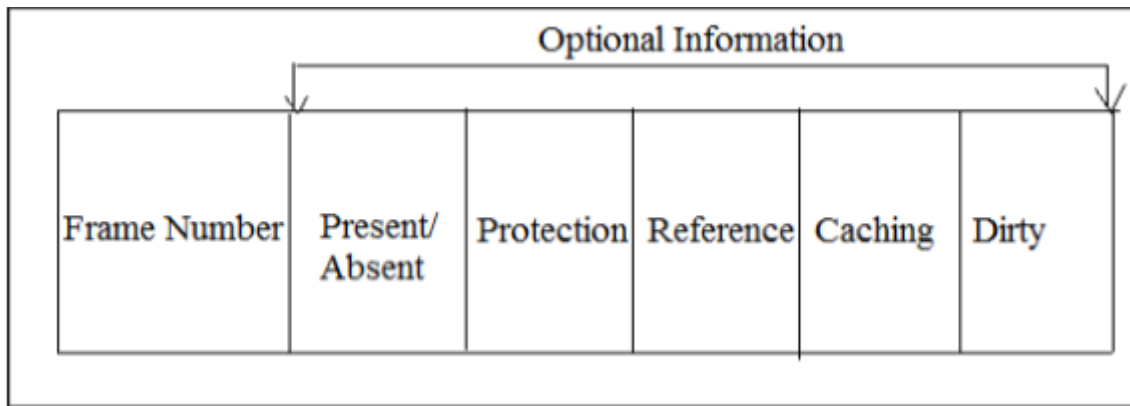
La CPU genera la dirección que se divide en dos partes, la primera es el número de página (p) y la segunda es el desplazamiento de página (d). El número de página se utiliza como índice en la tabla de páginas. La tabla de páginas almacena la dirección base de cada página en la memoria física. Esta dirección base se combina con el desplazamiento de página para definir la dirección de memoria física que se envía a la unidad de memoria.

Por ser independiente del hardware, solo considere que el hardware tiene registros triples y usos del kernel para la administración de memoria. El primer registro en el triple almacena la dirección de la tabla de páginas, el segundo registro contiene la primera dirección virtual asignada por la tabla de páginas y el tercer registro contiene la información de control, como el número de páginas en las tablas de páginas y los permisos de acceso a las páginas. Cuando se está ejecutando un proceso, el kernel carga dichos registros triples con los datos en las entradas de la región p.

Si un proceso accede a una dirección fuera de su espacio de direcciones virtuales, se produce una excepción. Supongamos que un proceso tiene de 0 a 16 Kbytes de espacio de direcciones y el proceso accede a la dirección virtual de 20 K, se generará una excepción y el sistema operativo la detectará. De manera similar, si un proceso intenta acceder a una página sin tener suficientes permisos, se generará una excepción. Y si se presenta tal situación, el proceso normalmente termina.

La tabla de páginas tiene entradas de tabla de páginas que consisten en un número de marco y bits de estado opcionales (como protección). La mayoría de los bits de estado utilizados en el sistema de memoria virtual. La parte más importante de PTE es el número de cuadro.

Una página tiene la siguiente información:



- **Bit presente/ausente:** el bit presente o ausente indica que una página en particular que está buscando está presente o ausente. Si no está presente, se llama Fallo de página. Se establece en 0 si la página respectiva no está en la memoria. A veces, este bit también se conoce como bits válidos/inválidos.
- **Bit de protección:** el bit de protección indica qué tipo de protección desea en esa página. Entonces, estos bits son para la protección del marco de la página (lectura, escritura, etc.).
- **Bit referenciado:** el bit referenciado le informará si esta página ha sido referenciada en el último ciclo de reloj o no. Se establece en 1 por hardware cuando se accede a la página.
- **Almacenamiento en caché habilitado/deshabilitado:** en algunos casos, queremos los datos nuevos. Solo considere que el usuario está escribiendo información desde el teclado y su programa debería ejecutarse de acuerdo con la entrada proporcionada por el usuario. En ese caso, la información llegará primero a la memoria principal. Por lo tanto, la memoria principal contiene la información actual que escribe el usuario.
- **Bit modificado:** El bit modificado dice si la página ha sido modificada o no. Modificado significa que a veces está tratando de hacer cambios o escribir algo en la página. Si se modifica una página, siempre que deba reemplazar esa página con alguna otra página, la información modificada debe mantenerse en el disco duro o debe volver a escribirse o guardarse nuevamente. El hardware lo establece en 1 en el acceso de escritura a la página que se usa para evitar la escritura cuando se intercambia. A veces, este bit modificado también se denomina bit sucio.

CONCLUSIONES.

Dentro de las operaciones más básicas y la vez más complejas de nuestra PC encontramos los procesos. Un proceso, debido a que está destinado a ser ejecutadas por el microprocesador, el sistema operativo almacena los estados de ejecución en un momento dado, lo cual permite realizar las operaciones con mucha facilidad y rapidez.

REFERENCIAS.

<https://www.kernel.org/doc/gorman/html/understand/understand006.html>

<https://www.ques10.com/p/43690/page-table-in-unix/>

<https://www.geeksforgeeks.org/processes-in-linuxunix/>

<https://www.tecmint.com/linux-process-management/>