Descripción breve

Pequeño manual de la creacíon de los módulos

Manual del proceso de creación de los módulos



Herminio García 2003-1755

Iván López 2011-22826

Creación de módulos

Comandos utilizados para creación de módulos

apt-get update

apt-get upgrade

apt-get install

apt-get

 make clean  
ls -R /lib/modules/$(uname -r)

lsmod

cat /proc/modules

lspci -v

lsmod | grep vfat

sudo depmod

sudo modprobe -v vfat

Mostramos las funciones utilizadas

Sysinfo

struct [**sysinfo**](https://elixir.bootlin.com/linux/latest/C/ident/sysinfo) {

};

Retorna información sobre la memoria RAM

**List\_head**

Lista vinculada que acepta cualquier estructura.

**task\_struct**

struct task\_struct {

};

Describe un proceso o tarea en el sistema.

static int [**meminfo\_proc\_show**](https://elixir.bootlin.com/linux/latest/C/ident/meminfo_proc_show)(struct [**seq\_file**](https://elixir.bootlin.com/linux/latest/C/ident/seq_file) \*m, void \*v)

{

}

static int [**\_\_init**](https://elixir.bootlin.com/linux/latest/C/ident/__init) [**proc\_meminfo\_init**](https://elixir.bootlin.com/linux/latest/C/ident/proc_meminfo_init)(void)

{

[**proc\_create\_single**](https://elixir.bootlin.com/linux/latest/C/ident/proc_create_single)("meminfo", 0, NULL, [**meminfo\_proc\_show**](https://elixir.bootlin.com/linux/latest/C/ident/meminfo_proc_show));

return 0;

}

[**fs\_initcall**](https://elixir.bootlin.com/linux/latest/C/ident/fs_initcall)(**[proc\_meminfo\_init](https://elixir.bootlin.com/linux/latest/C/ident/proc_meminfo_init)**);

1. Para la creación un módulo define básicamente dos funciones importantes que son las que se utilizan para su carga y descarga, esto son el init y exit que se declararían así

module\_init(nombre \_funcion)

module\_exit(nombre\_funcion)

2. Los módulos que se crearon son lo que se encargan de leer información básica de la memoria RAM y lo escribirá en un archivo en el directorio proc, y un módulo que mostrara el listado de procesos y su información de la misma forma, estos se llamaran memo\_ 201122826 y cpu\_ 201122826

4. Para poder ejecutar estos modulos debemos de crear un Makefile para cada uno de ellos con el siguiente codigo, en el nombre del objeto salida variamos cpu por memo.

Con este fichero Makefile, bastaría con ejecutar el mandato make para compilar el módulo. Para ese paso no es necesario ser usuario administrador

root# insmo

Para comprobar si el módulo se ha cargado, puede usarse el mandato lsmod y para ver si ha escrito el mensaje de bienvenida puede usar dmesg.

Por último, para descargar el módulo debe usar el siguiente mandato:

root# rmmod nombremodulo

Con lsmod y dmesg puede comprobar si la descarga se ha realizado satisfactoriamente.

NombreModulo.o

all:

make -C /lib/modules/$(shell uname -r)/build M=$(PWD) modules

clean:

make -C /lib/modules/$(shell uname -r)/build M=$(PWD) clean

5. Una vez guardamos los Makefile ingresamos el siguiente comando “make” y generara un archivo con extension ko que sera el que cargaremos al kernel.

6. Para cargar los modulos al kernel basta con introducir el siguiente comando “insmod nombre.ko”.

7. Una vez ya no sea necesario el modulo podemos descargarlo con el siguiente comando “rmmod nomre.ko”

8. Para ver los mensajes de salida al momento de la carga y descarga utilizaremos el comando “dmesg | tail -1”, el 1 nos indica cuantos mensajes obtendra de la cola para mostrar.