DEVICE DRIVERS – LAB EXERCISE 5

Submitted By:

Kiran Thomas Cherian

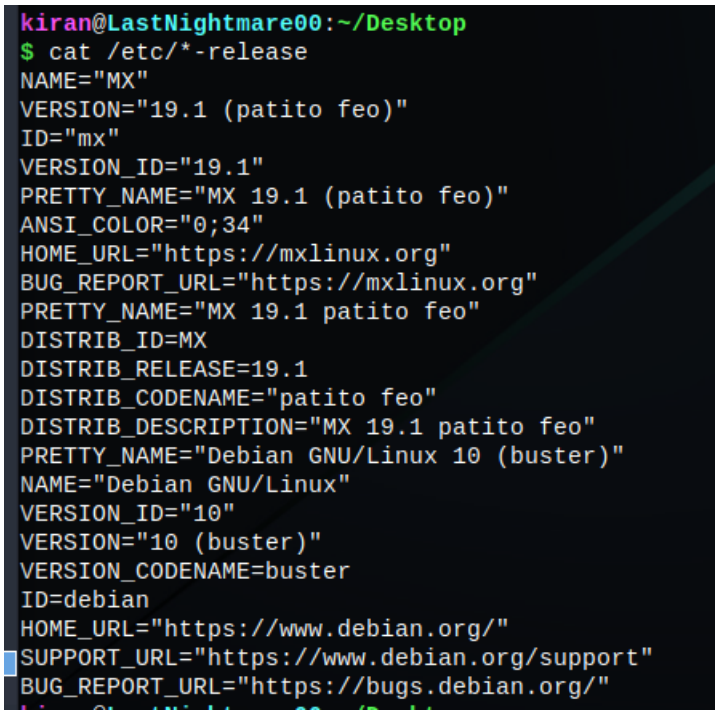
CED18I028

OBJECTIVE:

Write a C program. Compile. Insert the compiled program as a Kernel Module in Kernel. Remove the module from Kernel. Verify whether the insertion and deletion happened properly.

Linux Distribution Used:

MX Linux 19.1 (Running on Virtual machine)



Code:

```
#include <linux/module.h>     /* Needed by all modules */
#include <linux/kernel.h>     /* Needed for KERN_INFO */
#include <linux/init.h>       /* Needed for the macros */

///< The license type -- this affects runtime behavior
MODULE_LICENSE("GPL");

///< The author -- visible when you use modinfo
MODULE_AUTHOR("Kiran Thomas Cherian");

///< The description -- see modinfo
MODULE_DESCRIPTION("Hello World Kernal Modul Program");

///< The version of the module
MODULE_VERSION("0.1");

static int __init hello_start(void)
{
   printk(KERN_INFO "Loading my module\n");
   printk(KERN_INFO "Hello world by Kiran\n");
   return 0;
}
```

```
static void __exit hello_end(void)
{
   printk(KERN_INFO "Removing my module.bye..\n");
}

module_init(hello_start);
module_exit(hello_end);
```

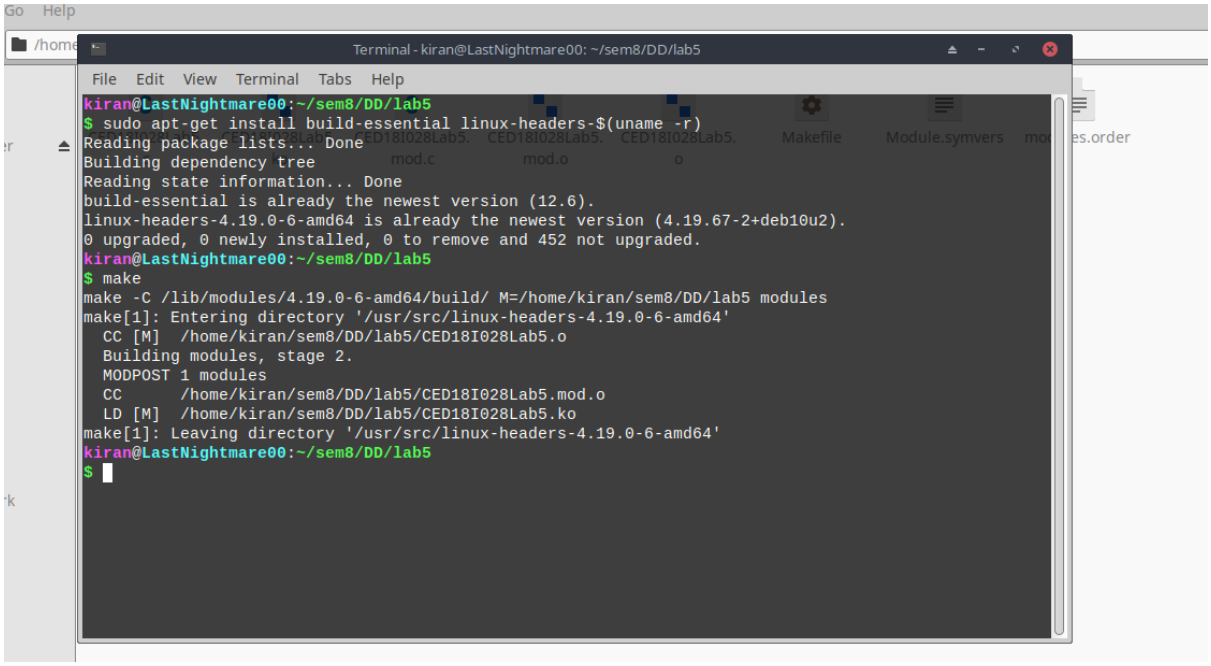
Makefile Contents:

```
obj-m = CED18I028Lab5.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
```

The system must be prepared to build kernel code, and to do this you must have the Linux headers installed on your device. Under 64-bit Debian you can use:

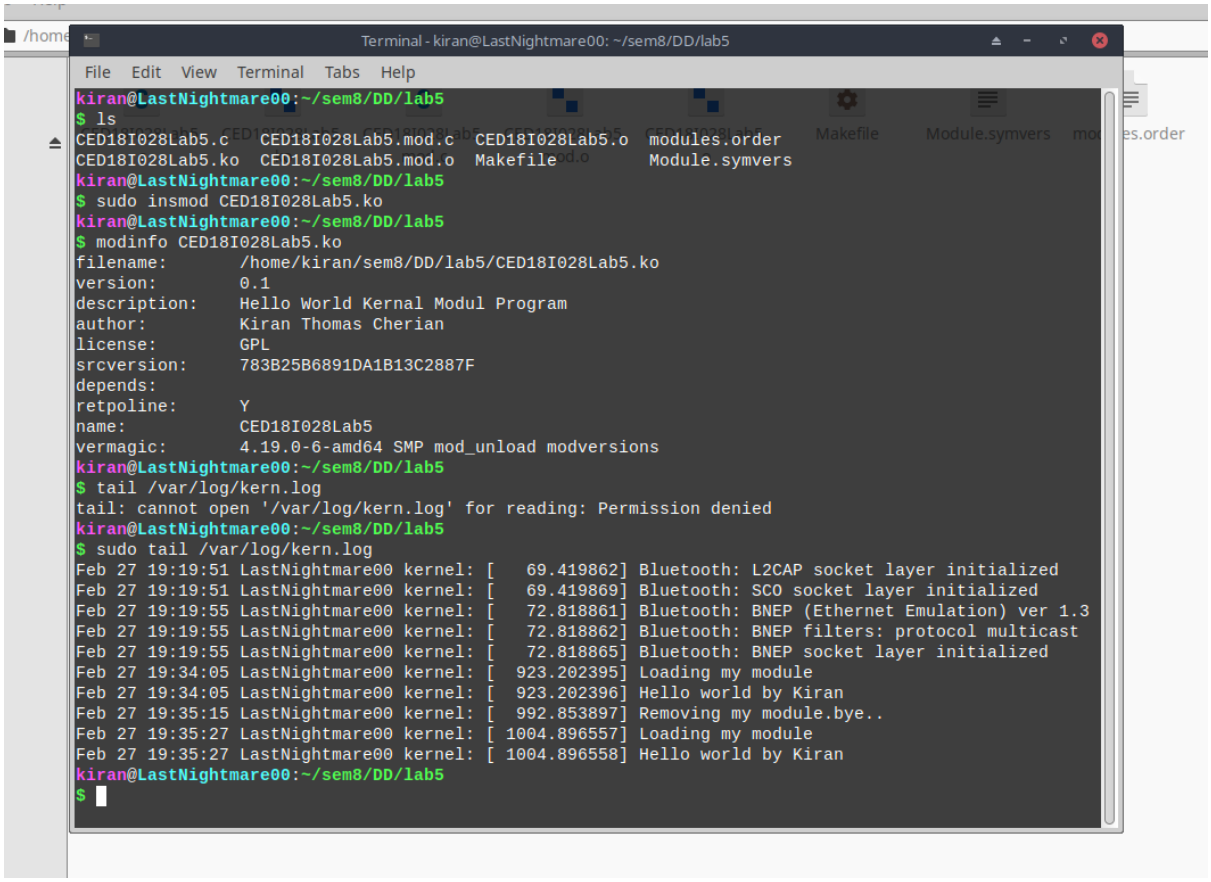
$ sudo apt-get install build-essential linux-headers-$(uname -r)

Run the make command to compile the source code.



Then use insmod to load the module.

You can get information about the module using the modinfo command, which will identify the description, author and any module parameters that are defined. And to see the message, we need to read the kern.log in /var/log directory



To unload the module, we run rmmod:

Now run the tail command to get the exit message.