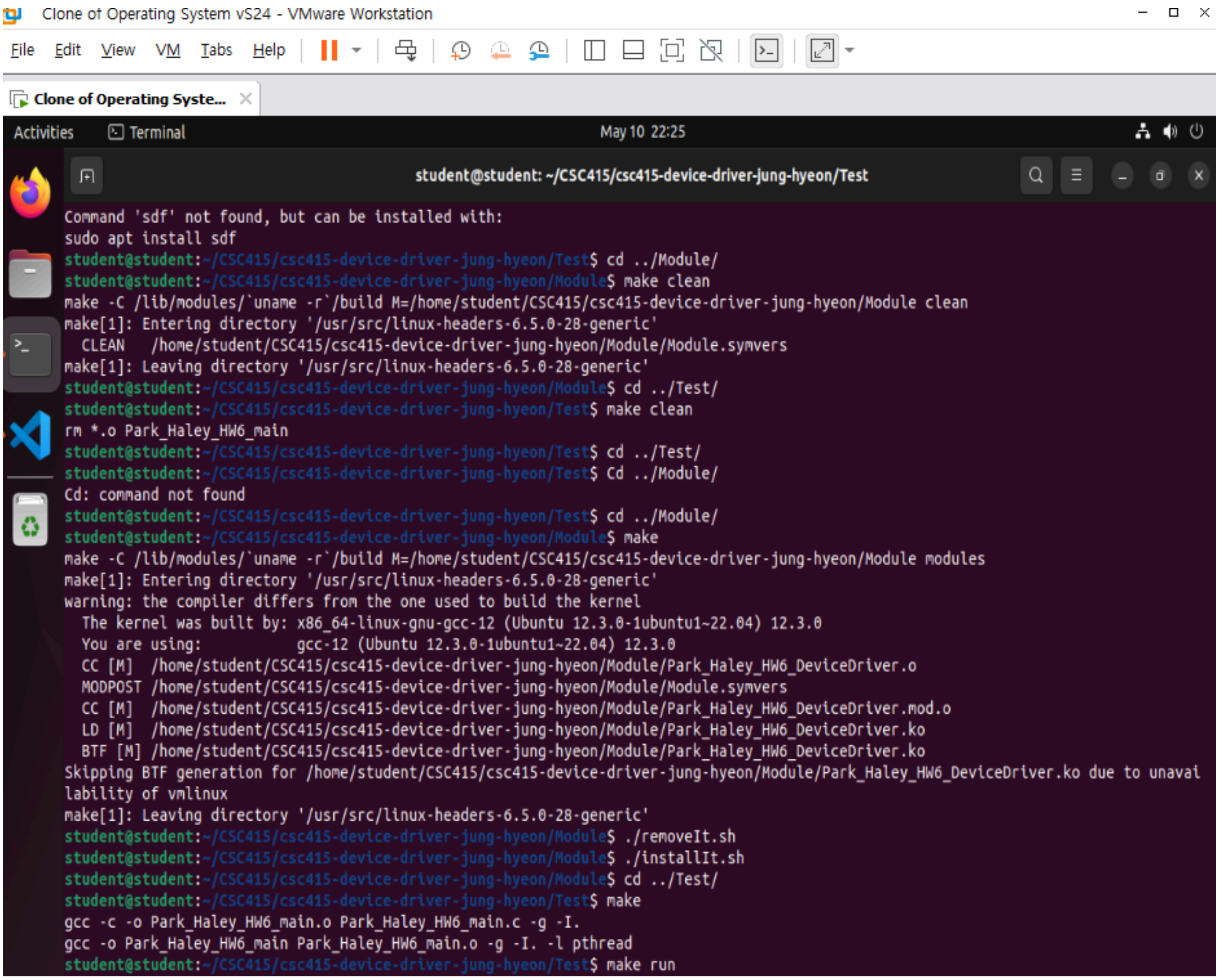
Haley Park
Github: jung-hyeon
ID: 923812276
CSC415 Operating Systems

Assignment 6 – Device Driver

Description:
This file implements a device driver for a simple encryption program. It provides functionalities to encrypt and decrypt data, read and write strings, and use IOCTL to set encryption/decryption mode and key. The encryption algorithm subtracts a fixed key value from each character of the string, while the decryption algorithm adds the key value to each character. If IOCTL is set to encryption mode, the driver expects a string to be written, then it reads back the encrypted data. If IOCTL is set to decryption mode, the driver expects encrypted data to be written, and it reads back the original message (decrypted).

How to build & load:

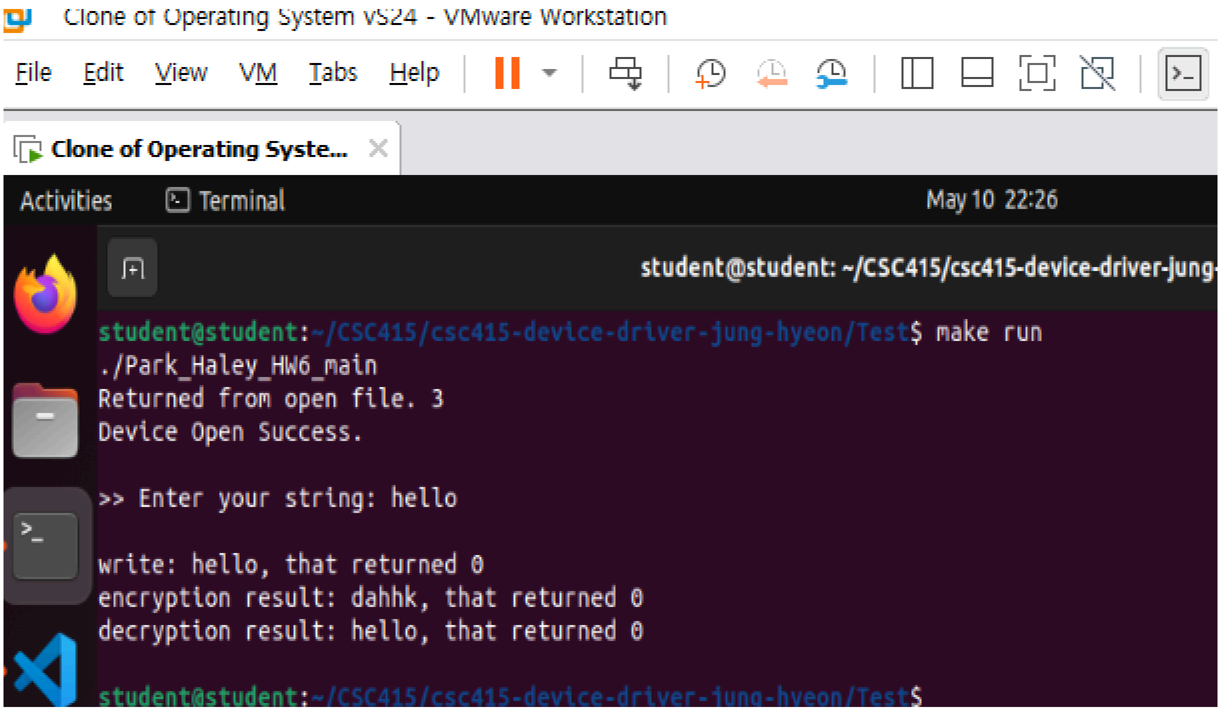
1) go to ../Module
2) run make
3) run ./removeIt.sh
4) run ./installIt.sh
5) go to ../Test
6) run make
7) run make run


Clone of Operating System vS24 - VMware Workstation
File
Edit
View
VM
Tabs
Help
Clone of Operating Syste...
Activities
Terminal
May 10 22:25
student@student: ~/CSC415/csc415-device-driver-jung-hyeon/Test

```
Command 'sdf' not found, but can be installed with:
sudo apt install sdf
student@student:~/CSC415/csc415-device-driver-jung-hyeon/Test$ cd ../Module/
student@student:~/CSC415/csc415-device-driver-jung-hyeon/Module$ make clean
make -C /lib/modules/`uname -r`/build M=/home/student/CSC415/csc415-device-driver-jung-hyeon/Module clean
make[1]: Entering directory '/usr/src/linux-headers-6.5.0-28-generic'
  CLEAN   /home/student/CSC415/csc415-device-driver-jung-hyeon/Module/Module.symvers
make[1]: Leaving directory '/usr/src/linux-headers-6.5.0-28-generic'
student@student:~/CSC415/csc415-device-driver-jung-hyeon/Module$ cd ../Test/
student@student:~/CSC415/csc415-device-driver-jung-hyeon/Test$ make clean
rm *.o Park_Haley_HW6_main
student@student:~/CSC415/csc415-device-driver-jung-hyeon/Test$ cd ../Test/
student@student:~/CSC415/csc415-device-driver-jung-hyeon/Test$ Cd ../Module/
Cd: command not found
student@student:~/CSC415/csc415-device-driver-jung-hyeon/Test$ cd ../Module/
student@student:~/CSC415/csc415-device-driver-jung-hyeon/Module$ make
make -C /lib/modules/`uname -r`/build M=/home/student/CSC415/csc415-device-driver-jung-hyeon/Module modules
make[1]: Entering directory '/usr/src/linux-headers-6.5.0-28-generic'
warning: the compiler differs from the one used to build the kernel
  The kernel was built by: x86_64-linux-gnu-gcc-12 (Ubuntu 12.3.0-1ubuntu1~22.04) 12.3.0
  You are using:           gcc-12 (Ubuntu 12.3.0-1ubuntu1~22.04) 12.3.0
  CC [M]  /home/student/CSC415/csc415-device-driver-jung-hyeon/Module/Park_Haley_HW6_DeviceDriver.o
  MODPOST /home/student/CSC415/csc415-device-driver-jung-hyeon/Module/Module.symvers
  CC [M]  /home/student/CSC415/csc415-device-driver-jung-hyeon/Module/Park_Haley_HW6_DeviceDriver.mod.o
  LD [M]  /home/student/CSC415/csc415-device-driver-jung-hyeon/Module/Park_Haley_HW6_DeviceDriver.ko
  BTF [M] /home/student/CSC415/csc415-device-driver-jung-hyeon/Module/Park_Haley_HW6_DeviceDriver.ko
Skipping BTF generation for /home/student/CSC415/csc415-device-driver-jung-hyeon/Module/Park_Haley_HW6_DeviceDriver.ko due to unavai
lability of vmlinux
make[1]: Leaving directory '/usr/src/linux-headers-6.5.0-28-generic'
student@student:~/CSC415/csc415-device-driver-jung-hyeon/Module$ ./removeIt.sh
student@student:~/CSC415/csc415-device-driver-jung-hyeon/Module$ ./installIt.sh
student@student:~/CSC415/csc415-device-driver-jung-hyeon/Module$ cd ../Test/
student@student:~/CSC415/csc415-device-driver-jung-hyeon/Test$ make
gcc -c -o Park_Haley_HW6_main.o Park_Haley_HW6_main.c -g -I.
gcc -o Park_Haley_HW6_main Park_Haley_HW6_main.o -g -I. -l pthread
student@student:~/CSC415/csc415-device-driver-jung-hyeon/Test$ make run
```

How to interact:

1) enter the string that user wants to encrypt
2) the result is
    a) encryption data from the string entered by user
    b) decryption data from the above encrypt data


Clone of Operating System vS24 - VMware Workstation
File
Edit
View
VM
Tabs
Help
Clone of Operating Syste...
Activities
Terminal
May 10 22:26
student@student: ~/CSC415/csc415-device-driver-jung-

```
student@student:~/CSC415/csc415-device-driver-jung-hyeon/Test$ make run
./Park_Haley_HW6_main
Returned from open file. 3
Device Open Success.

>> Enter your string: hello

write: hello, that returned 0
encryption result: dahhk, that returned 0
decryption result: hello, that returned 0

student@student:~/CSC415/csc415-device-driver-jung-hyeon/Test$
```

Approach:

[Device Driver]

1. Header Files Inclusion:
    a. Necessary header files such as <linux/module.h>, <linux/kernel.h>, <linux/fs.h>, <linux/cdev.h>, and <linux/vmalloc.h> are included.
2. Module Information:
    a. The module metadata such as author, description, and license is specified using macros like MODULE_AUTHOR, MODULE_DESCRIPTION, and MODULE_LICENSE.
3. Device Information:
    a. Major and minor numbers are defined for the device.
    b. The device name is specified.
4. Custom Data Structure:
    a. struct myds is defined to keep track of read/write counts and encryption mode.
5. Global Variables:
    a. kernel_buffer is a pointer to store data written by user space.
    b. actual_rx_size keeps track of the actual size of data written.
    c. my_cdev is an instance of struct cdev representing the character device.
6. File Operations Functions:
    a. myOpen: Allocates memory for struct myds and initializes it.
    b. myClose: Releases memory allocated in myOpen.
    c. myWrite: Writes data from user space to kernel space, encrypts it, and keeps track of the count of write operations.

   d. myRead: Reads encrypted data from kernel space, decrypts it, and sends it to user space, while keeping track of the count of read operations.
   e. myIoCtl: Handles IOCTL commands to set encryption mode (encrypt or decrypt).
7. Encryption/Decryption Functions:
   a. encryptOrDecrypt: Performs encryption or decryption based on the mode set in myIoCtl.
8. Initialization and Cleanup Functions:
   a. init_module: Registers the device, initializes the character device structure, and adds it to the system.
   b. cleanup_module: Unregisters the device and cleans up resources.
9. Main Function:
   a. The entry point for the module is init_module, and it calls the necessary initialization functions.
   b. The exit point is cleanup_module, which handles the cleanup tasks.

[application (main.c)]

This C file serves as a test program to interact with the device driver created in the previous code (Park_Haley_HW6_DeviceDriver.c).

1. Header Files Inclusion:
   - Standard C library headers like <stdio.h>, <string.h>, <stdlib.h>, and POSIX headers like <sys/ioctl.h>, <fcntl.h>, <unistd.h>, and <errno.h> are included.
2. Macro Definitions:
   - Macros are defined for the device name (DEV_NAME) and the maximum length of input strings (MAX_LEN).
3. Main Function:
   - The main function is the entry point of the program.
   - A file descriptor fd is declared to interact with the device driver.
   - An array read_buf is allocated to store the read data.
   - Variables read_ret and write_ret are used to store the return values of read and write operations, respectively.
   - An array mystr is declared to store the user input string.
   - An integer key is declared to store the encryption/decryption key.
4. Device File Opening:
   - The device file is opened using open system call with the specified device name (DEV_NAME).
   - Error handling is performed to check if the file descriptor fd is valid.
5. User Input:
   - The user is prompted to enter a string via printf and scanf functions, which is stored in mystr.
6. Writing to Device:
   - The user input string (mystr) is written to the device using the write system call.

- The return value of the write operation is printed for debugging.

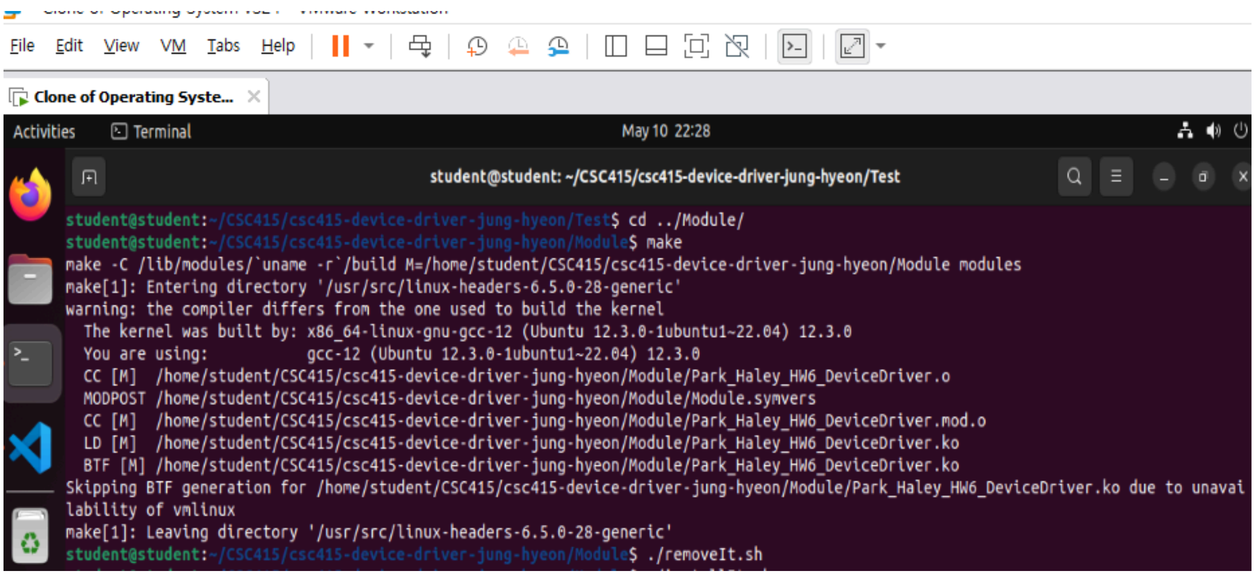
7. Encryption and Decryption:
   - The IOCTL command ioctl(fd, 1, 1) is used to set the encryption mode.
   - The encrypted data is read from the device using the read system call, and the result is printed.
   - The IOCTL command ioctl(fd, 1, 2) is used to set the decryption mode.
   - The decrypted data is read from the device using the read system call, and the result is printed.
8. Device File Closing:
   - The device file is closed using the close system call.
9. Return:
   - The main function returns 0 to indicate successful execution.

Issues and Resolutions:

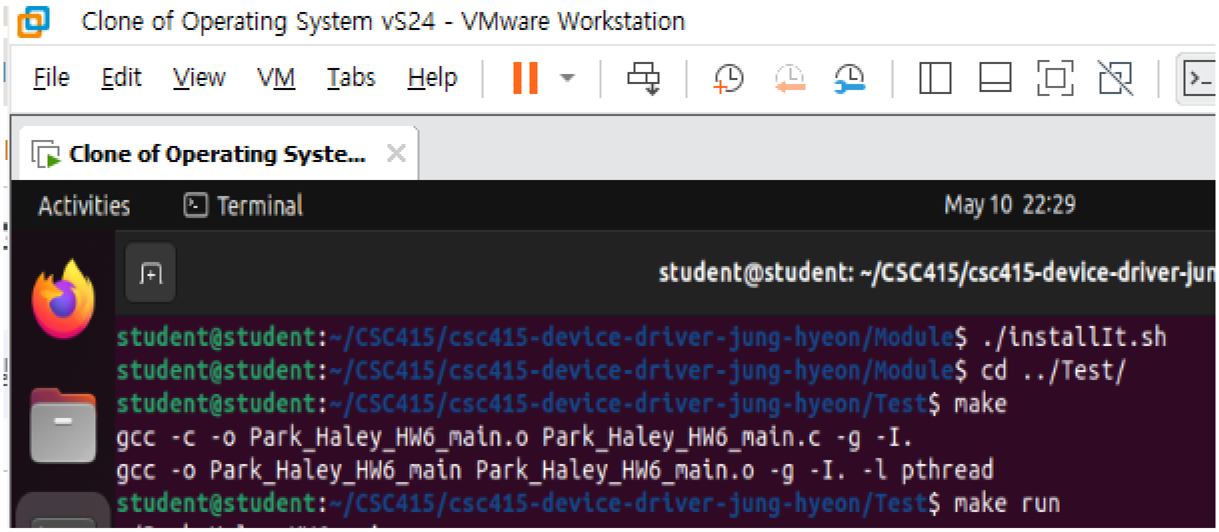
1. copy_from_user() & copy_to_user()
   a. I tried to write to the kernel and read from the kernel. So, I just tried to pass through the parameters. But It didn't work. We have to use that functions (copy_from_user(), copy_to_user()) to interact between user and kernel,
2. using ioctl()
   a. I created the myIoCtl function. But when I use this function, I confused the parameters. To write something in the kernel in that function, I need to pass the 1 by second parameter.
3. build process
   a. After compiling the device driver file, I need to connect the device file again.
      i. sudo insmod Park_Haley_HW6_DeviceDriver.ko
      ii. sudo mknod /dev/Park_Haley_HW6_DeviceDriver c 415 0
      iii. sudo chmod 666 /dev/Park_Haley_HW6_DeviceDriver
   b. I didn't know that. I wasted a lot of time.

Analysis:

Screen shot of compilation:

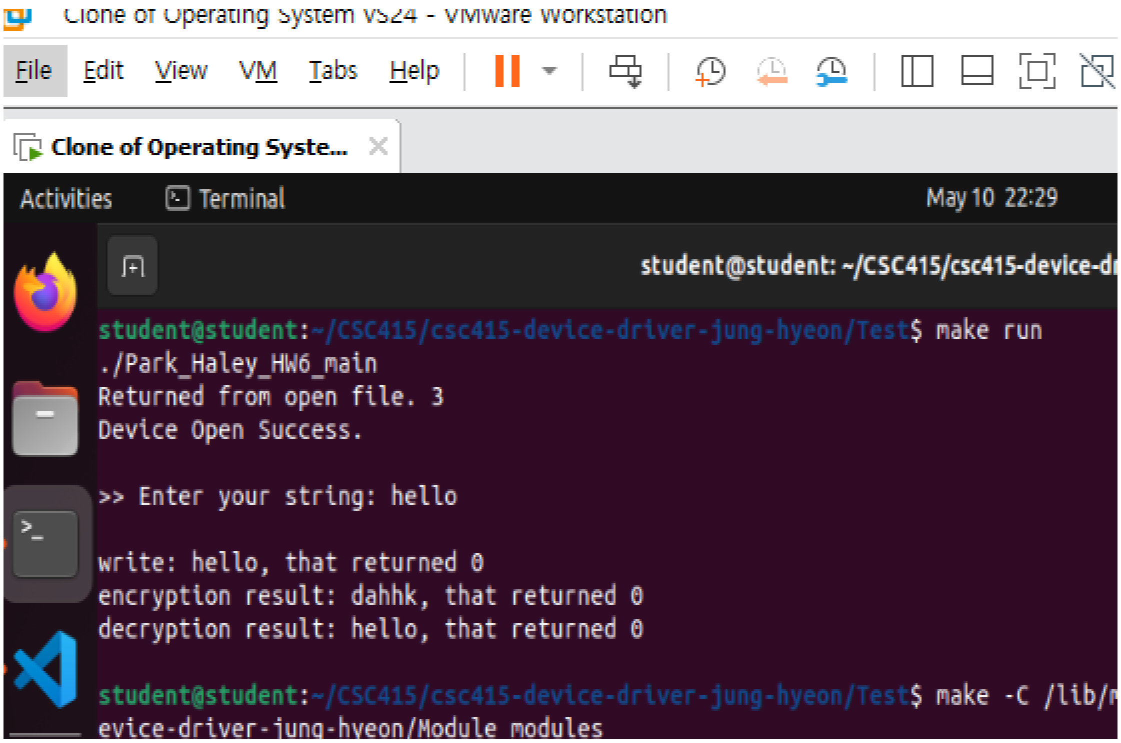

File Edit View VM Tabs Help
Clone of Operating Syste...
Activities
Terminal
May 10 22:28
student@student: ~/CSC415/csc415-device-driver-jung-hyeon/Test

```
student@student:~/CSC415/csc415-device-driver-jung-hyeon/Test$ cd ../Module/
student@student:~/CSC415/csc415-device-driver-jung-hyeon/Module$ make
make -C /lib/modules/`uname -r`/build M=/home/student/CSC415/csc415-device-driver-jung-hyeon/Module modules
make[1]: Entering directory '/usr/src/linux-headers-6.5.0-28-generic'
warning: the compiler differs from the one used to build the kernel
  The kernel was built by: x86_64-linux-gnu-gcc-12 (Ubuntu 12.3.0-1ubuntu1~22.04) 12.3.0
  You are using:           gcc-12 (Ubuntu 12.3.0-1ubuntu1~22.04) 12.3.0
  CC [M]  /home/student/CSC415/csc415-device-driver-jung-hyeon/Module/Park_Haley_HW6_DeviceDriver.o
  MODPOST /home/student/CSC415/csc415-device-driver-jung-hyeon/Module/Module.symvers
  CC [M]  /home/student/CSC415/csc415-device-driver-jung-hyeon/Module/Park_Haley_HW6_DeviceDriver.mod.o
  LD [M]  /home/student/CSC415/csc415-device-driver-jung-hyeon/Module/Park_Haley_HW6_DeviceDriver.ko
  BTF [M] /home/student/CSC415/csc415-device-driver-jung-hyeon/Module/Park_Haley_HW6_DeviceDriver.ko
Skipping BTF generation for /home/student/CSC415/csc415-device-driver-jung-hyeon/Module/Park_Haley_HW6_DeviceDriver.ko due to unavai
lability of vmlinux
make[1]: Leaving directory '/usr/src/linux-headers-6.5.0-28-generic'
student@student:~/CSC415/csc415-device-driver-jung-hyeon/Module$ ./removeIt.sh
```


Clone of Operating System vS24 - VMware Workstation
File Edit View VM Tabs Help
Clone of Operating Syste...
Activities
Terminal
May 10 22:29
student@student: ~/CSC415/csc415-device-driver-jun

```
student@student:~/CSC415/csc415-device-driver-jung-hyeon/Module$ ./installIt.sh
student@student:~/CSC415/csc415-device-driver-jung-hyeon/Module$ cd ../Test/
student@student:~/CSC415/csc415-device-driver-jung-hyeon/Test$ make
gcc -c -o Park_Haley_HW6_main.o Park_Haley_HW6_main.c -g -I.
gcc -o Park_Haley_HW6_main Park_Haley_HW6_main.o -g -I. -l pthread
student@student:~/CSC415/csc415-device-driver-jung-hyeon/Test$ make run
```

Screen shot(s) of the execution of the program:



Clone of Operating System vs24 - VMware Workstation
File
Edit
View
VM
Tabs
Help
Clone of Operating Syste...
Activities
Terminal
May 10 22:29
student@student: ~/CSC415/csc415-device-d

```
student@student:~/CSC415/csc415-device-driver-jung-hyeon/Test$ make run
./Park_Haley_HW6_main
Returned from open file. 3
Device Open Success.

>> Enter your string: hello

write: hello, that returned 0
encryption result: dahhk, that returned 0
decryption result: hello, that returned 0

student@student:~/CSC415/csc415-device-driver-jung-hyeon/Test$ make -C /lib/
evice-driver-jung-hyeon/Module modules
```