Kernel Compilation:

```
johnlol@johnlol-VirtualBox:~/Desktop/linux$ uname -a
Linux johnlol-VirtualBox 6.1.0-os-313552052 #3 SMP PREEMPT_DYNAMIC Tue Oct
 15 17:14:14 CST 2024 x86_64 x86_64 x86_64 GNU/Linux
johnlol@johnlol-VirtualBox:~/Desktop/linux$ cat /etc/os-release
PRETTY_NAME="Ubuntu 24.04.1 LTS"
NAME="Ubuntu"
VERSION_ID="24.04"
VERSION="24.04.1 LTS (Noble Numbat)"
VERSION_CODENAME=noble
ID=ubuntu
ID_LIKE=debian
HOME_URL="https://www.ubuntu.com/"
SUPPORT_URL="https://help.ubuntu.com/"
BUG_REPORT_URL="https://bugs.launchpad.net/ubuntu/"
PRIVACY_POLICY_URL="https://www.ubuntu.com/legal/terms-and-policies/privac
y-policy"
UBUNTU_CODENAME=noble
LOGO=ubuntu-logo
johnlol@johnlol-VirtualBox:~/Desktop/linux$
```

The steps of kernel compilation process:

Step 1.新增 revstr 資料夾，並在該資料夾中新增 sys_revstr.c，實作 system call

Step 2.在 revstr 資料夾中建立 Makefile，並寫入指令:

```
obj-y := sys_revstr.o
~
```

Step 3.在 Kbuild 中新增一行:

```
obj-y                   += revstr/
"Kbuild" 100L, 2608B
```

Step 4.在 include/linux/syscalls.h 新增一行:

```
asmlinkage long sys_revstr(char *str, int len);
#endif
"include/linux/syscalls.h" 1389L, 56918B
```

Step 5.在 include/uapi/asm-generic/unistd.h 新增一行

```
#define __NR_revstr 451
__SYSCALL(__NR_revstr, sys_revstr)
```

Step 6.在 kernel/sys_ni.c 新增一行:

```
COND_SYSCALL(revstr);
```

Step 7.在 arch/x86/entry/syscalls/syscall_64.tbl 新增一行:

```
451     common  revstr                  sys_revstr
```

Step 8. make menuconfig 更改 local version 並 configure kernel

(optional)過程中遇到 No rules to make target ‘debian/canonical-certs.pem’, needed by ‘certs/x509_certificate_list’:

處理: 將.config 檔中 CONFIG_SYSTEM_TRUSTED_KEYS 和 CONFIG_SYSTEM_REVOCATION_KEYS 的值清除

Step 9. sudo make 編譯

Step 10. sudo make modules_install

Step 11. sudo make install

Step 12. sudo update-grub

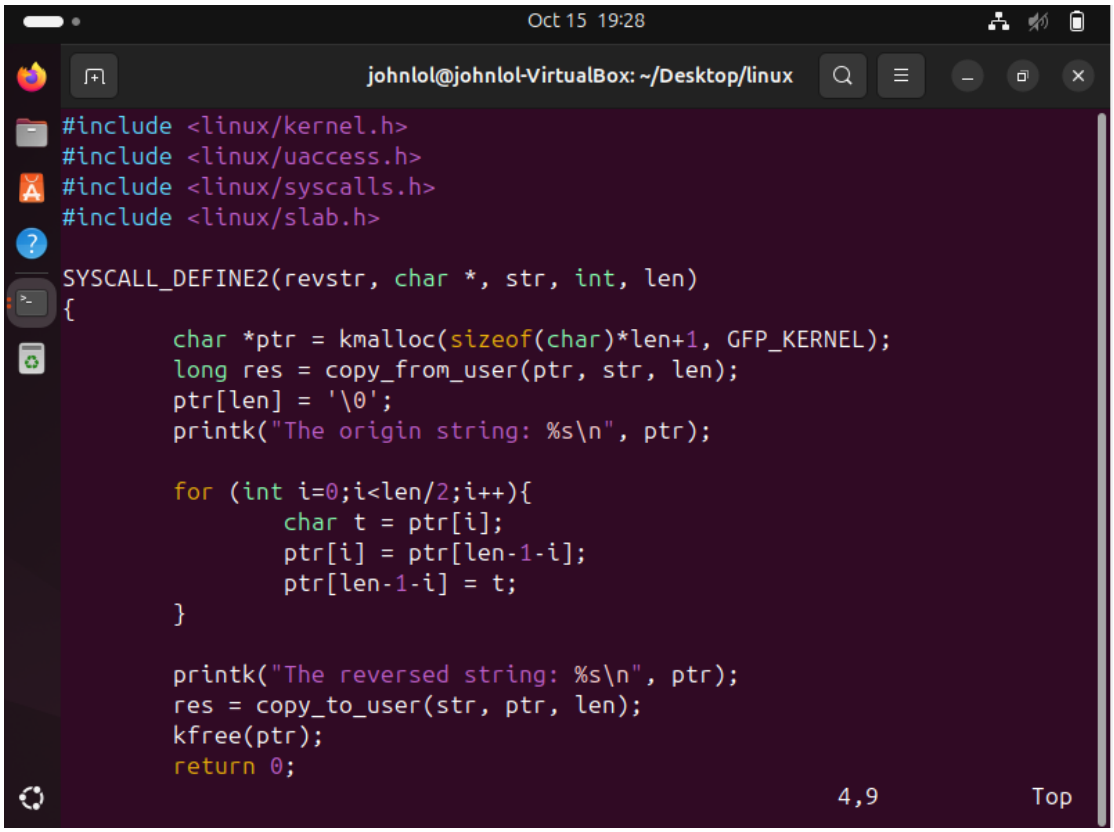
Step 13. sudo reboot

System call implementation:

```
johnlol@johnlol-VirtualBox:~/Desktop/linux$ ./main
Ori: hello
Rev: olleh
Ori: Operating System
Rev: metsyS gnitarepO
johnlol@johnlol-VirtualBox:~/Desktop/linux$
```

```
[ 4143.394076] The origin string: hello
[ 4143.394080] The reversed string: olleh
[ 4143.394082] The origin string: Operating System
[ 4143.394083] The reversed string: metsyS gnitarepO
johnlol@johnlol-VirtualBox:~$
```

Implementation for the system call:



先利用 kmalloc()取得一塊 kernel memory，大小為 len + 1 bytes，接著利用 copy_from_user()將 str 從 user space 複製到 kernel space，因兩者之間的資料是不共用的。字串反轉的實作方式是利用頭尾做 swap，並在最後一格插上終止符號，最後使用 copy_to_user()複製到 user space 的 str，實現直接更改 str 的方式，system call 結束後呼叫 kfree()釋放原先取得的 kernel memory。