Fayoum unveracity

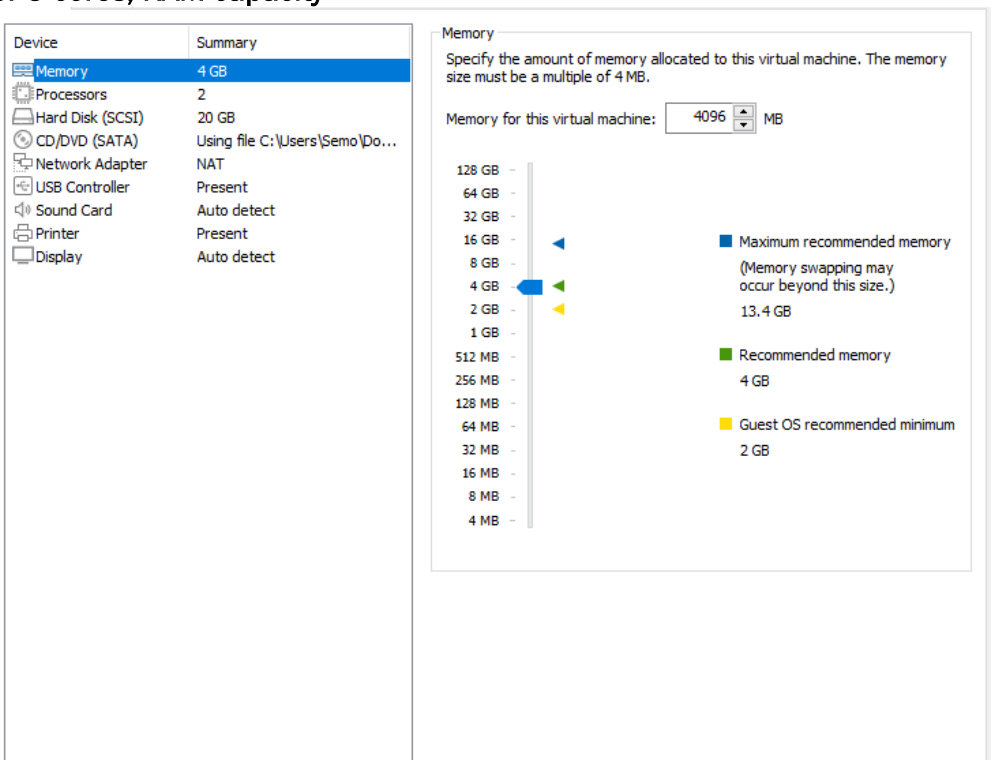
Operating System report

how to add your system call the Linux OS kernel.

Name Here
Sama Tarek Ahmed Ead G1
Rahma Mostafa G1

Description of my system

CPU cores, RAM capacity



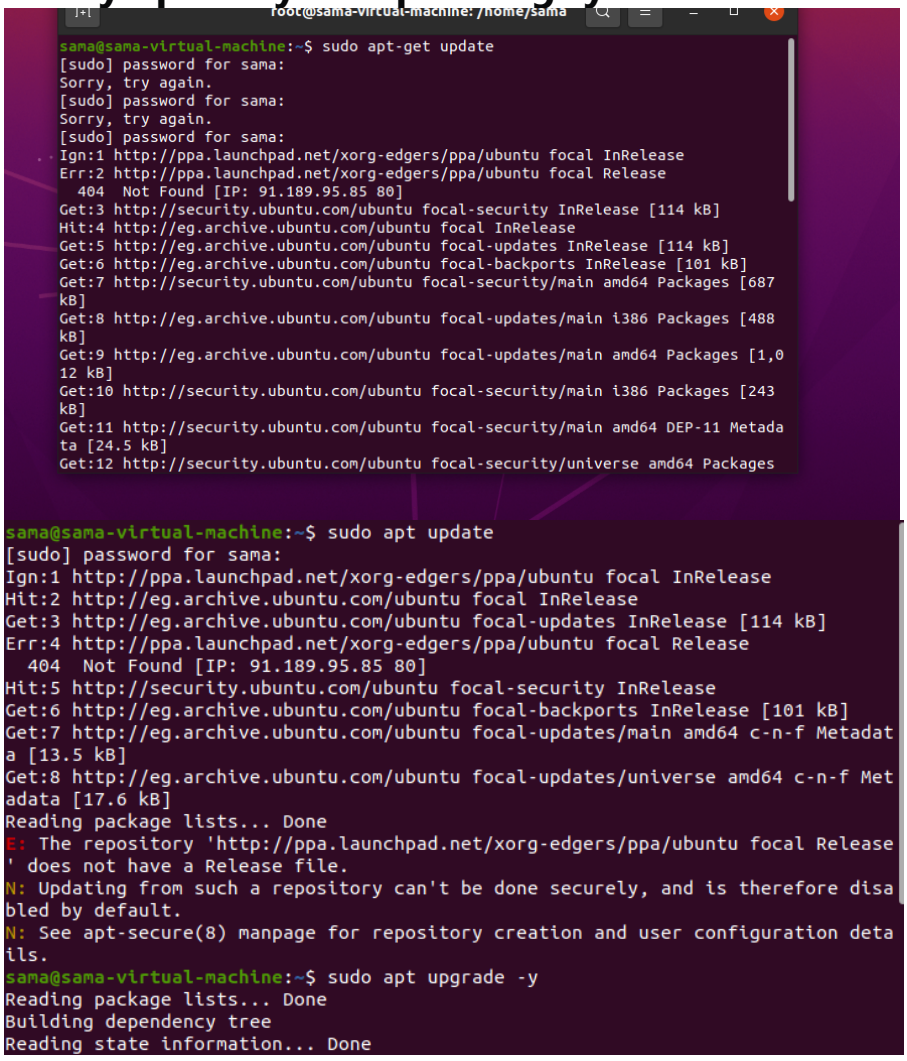
Kernel version
5.8.0.55-generic

```
sama@sama-virtual-machine:~$ uname -r
5.8.0-55-generic
sama@sama-virtual-machine:~$ cd ~
sama@sama-virtual-machine:~$ nano report.c
sama@sama-virtual-machine:~$
```

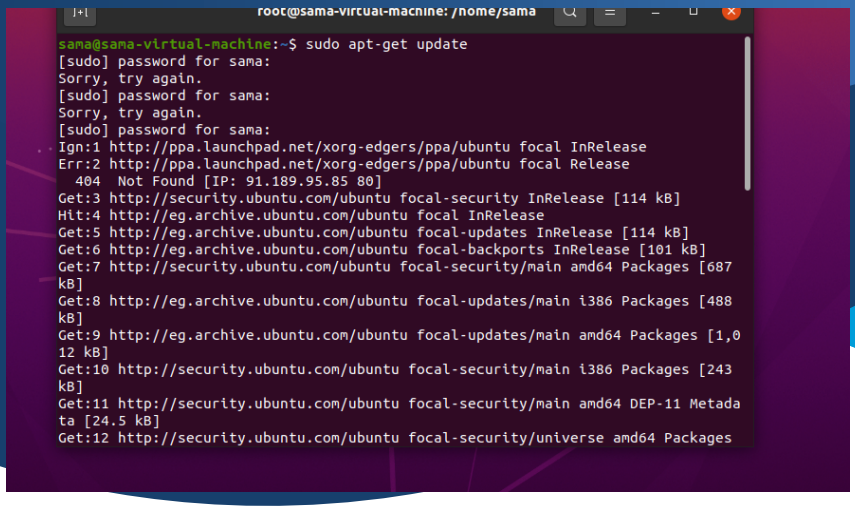
Steps of how adding the system call & Screenshots.

1 - Preparation

1.1 - Fully update your operating system



1.2 - Download and install the essential packages to compile kernels.



2 - Creation

2.1 **- Check the version of your current kernel.**

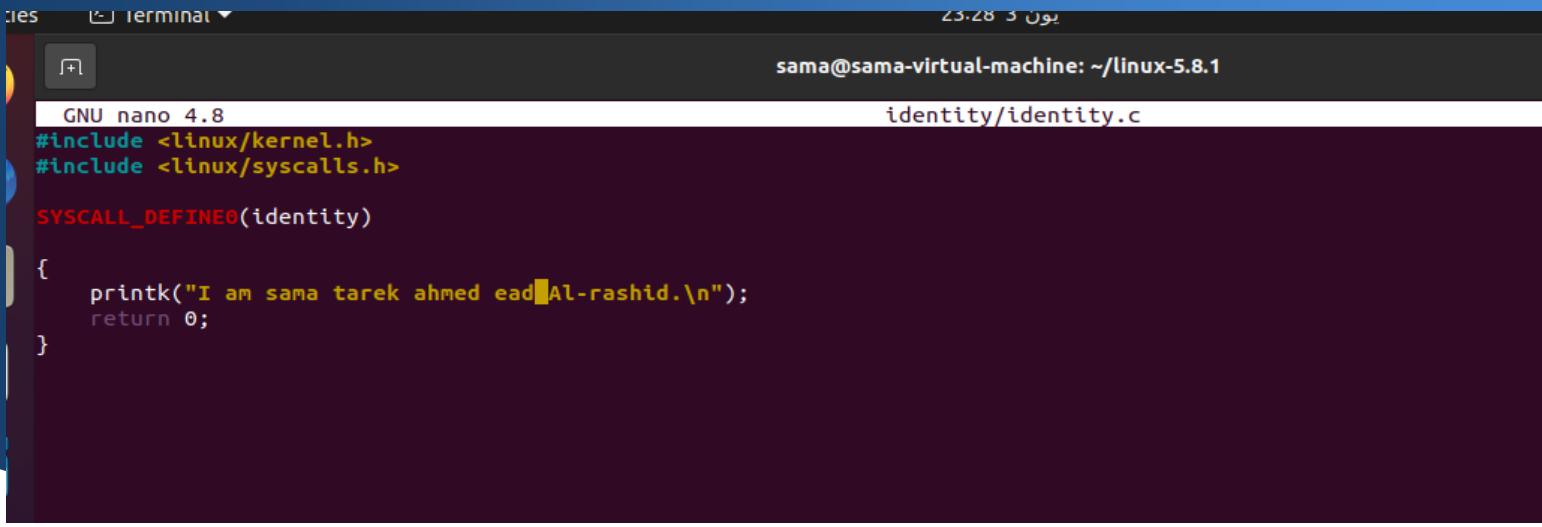
2.2 **- Change your working directory to the root directory of the recently unpacked source code.**
That is cd ~/linux-5.8.1/

2.3 - Create the home directory of your system call. Name is `identity`

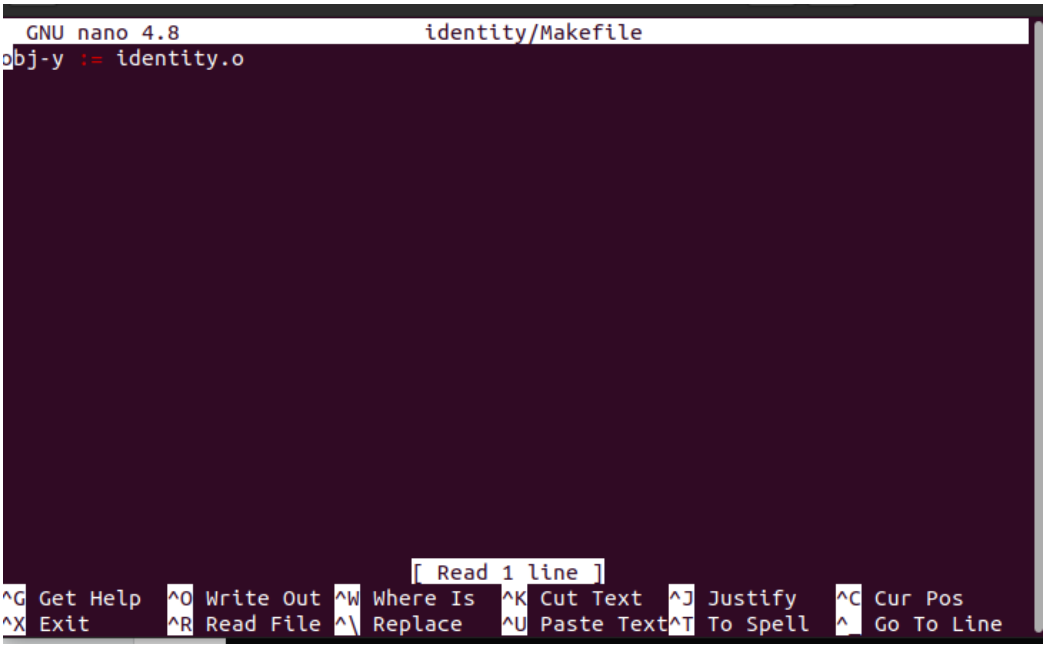
2.4 - Create a C file for your system call.

By nano identity/identity.c & Write the following code in it.

```
done
Removing linux-modules-5.8.0-43-generic (5.8.0-43.49~20.04.1) ...
Processing triggers for libc-bin (2.31-0ubuntu9.2) ...
sama@sama-virtual-machine:~$ sudo apt autoremove -y
Reading package lists... Done
Building dependency tree
Reading state information... Done
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
sama@sama-virtual-machine:~$ wget -P ~/ https://cdn.kernel.org/pub/linux/kernel/v5.x/linux-5.8.1.tar.xz
--2021-06-03 23:18:51--  https://cdn.kernel.org/pub/linux/kernel/v5.x/linux-5.8.1.tar.xz
Resolving cdn.kernel.org (cdn.kernel.org)... 199.232.81.176, 2a04:4e42:54::432
Connecting to cdn.kernel.org (cdn.kernel.org)|199.232.81.176|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 114458544 (109M) [application/x-xz]
Saving to: '/home/sama/linux-5.8.1.tar.xz'

linux-5.8.1.tar.xz                100%[=================================================================>] 109.16M  1.57MB/s    in 67s

2021-06-03 23:19:58 (1.62 MB/s) - '/home/sama/linux-5.8.1.tar.xz' saved [114458544/114458544]

sama@sama-virtual-machine:~$ uname -r
5.8.0-53-generic
sama@sama-virtual-machine:~$ cd ~/linux-5.8.1/
bash: cd: /home/sama/linux-5.8.1/: No such file or directory
sama@sama-virtual-machine:~$ l
Desktop/  Documents/  Downloads/  linux-5.8.1.tar.xz  Music/  Pictures/  Public/  snap/  Templates/  Videos/
sama@sama-virtual-machine:~$ ls
Desktop  Documents  Downloads  linux-5.8.1.tar.xz  Music  Pictures  Public  snap  Templates  Videos
sama@sama-virtual-machine:~$ cd ~/linux-5.8.1/
sama@sama-virtual-machine:~/linux-5.8.1$ mkdir identity
sama@sama-virtual-machine:~/linux-5.8.1$ nano identity/identity.c
sama@sama-virtual-machine:~/linux-5.8.1$ nano identity/Makefile
```

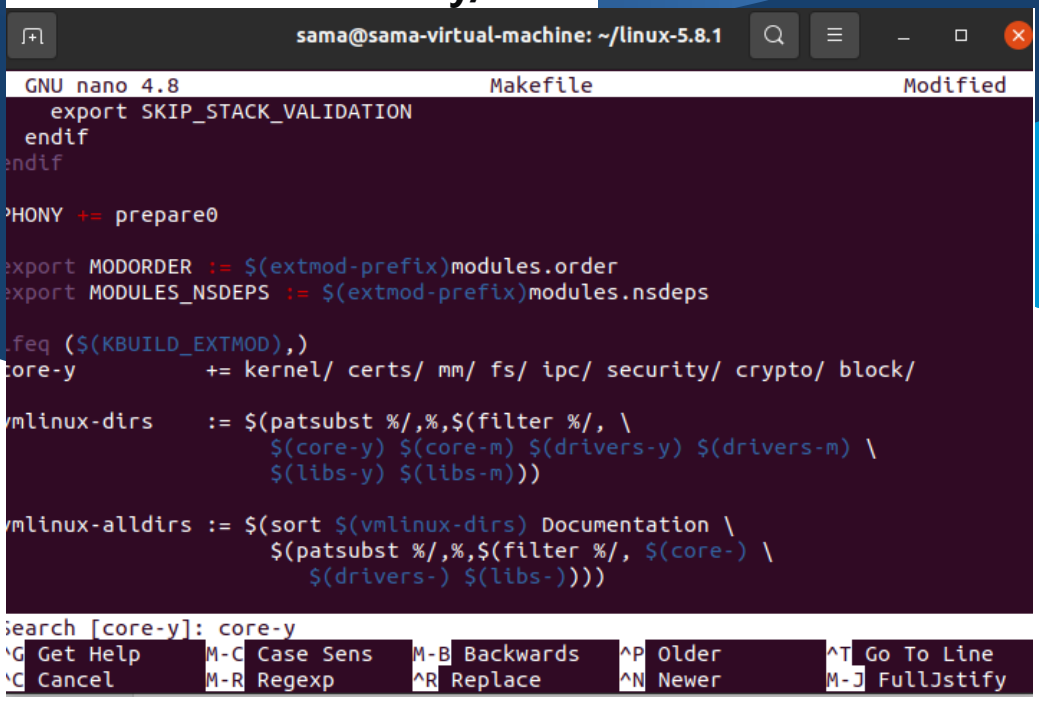


2.5 – Create a Makefile for your system call name identity Write the following code in it.



2.6 **- Add the home directory of your system call to the main Makefile of the kernel**

Search for core-y. In the second result, you will see a series of directories.Add "identity/" to it



2.7 - Add a corresponding function prototype for your system call to the header file of system calls.

By this command

nano include/linux/syscalls.h

following code just above #endif

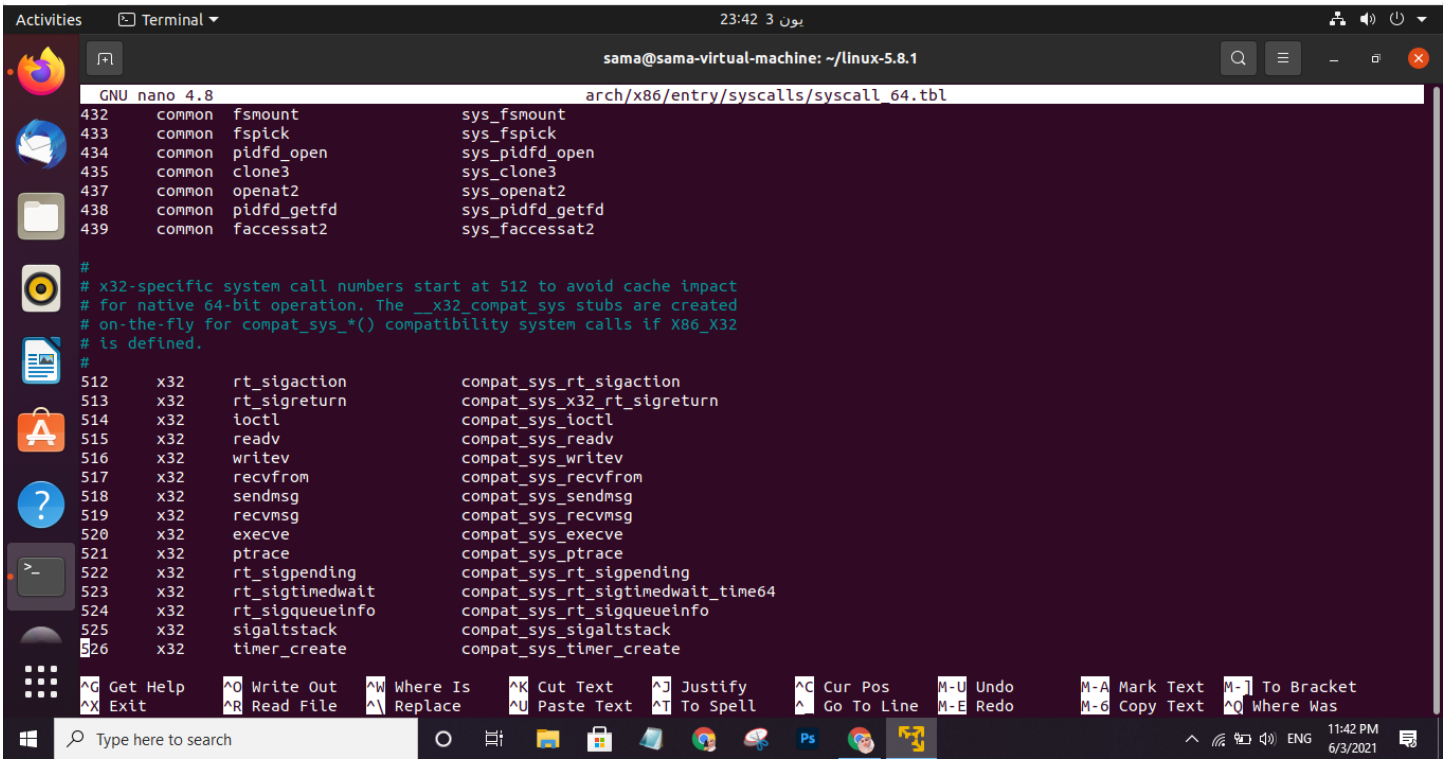
asmlinkage long sys_identity(void);

```
403 }
404
405 /* for __ARCH_WANT_SYS_IPC */
406 long ksys_semtimedop(int semid, struct sembuf __user *tsops,
407                      unsigned int nsops,
408                      const struct __kernel_timespec __user *timeout);
409 long ksys_semget(key_t key, int nsems, int semflg);
410 long ksys_old_semctl(int semid, int semnum, int cmd, unsigned long arg);
411 long ksys_msgget(key_t key, int msgflg);
412 long ksys_old_msgctl(int msqid, int cmd, struct msqid_ds __user *buf);
413 long ksys_msgrcv(int msqid, struct msgbuf __user *msgp, size_t msgsz,
414                  long msgtyp, int msgflg);
415 long ksys_msgsnd(int msqid, struct msgbuf __user *msgp, size_t msgsz,
416                  int msgflg);
417 long ksys_shmget(key_t key, size_t size, int shmflg);
418 long ksys_shmdt(char __user *shmaddr);
419 long ksys_old_shmctl(int shmid, int cmd, struct shmid_ds __user *buf);
420 long compat_ksys_semtimedop(int semid, struct sembuf __user *tsems,
421                             unsigned int nsops,
422                             const struct old_timespec32 __user *timeout);
423
424
425 asmlinkage long sys_h
```

2.8 - **Add your system call to the kernel's system call table.**
By this command

nano arch/x86/entry/syscalls/syscall_64.tbl

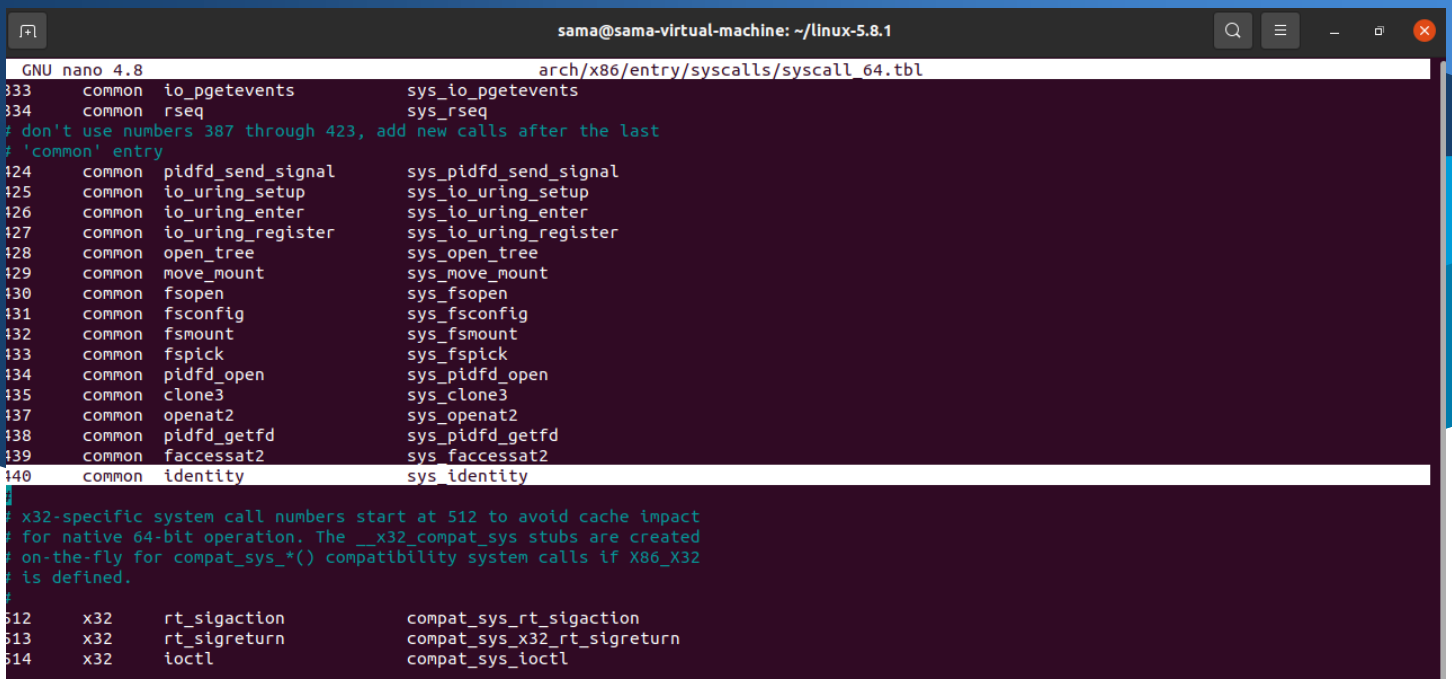
```
sama@sama-virtual-machine:~/linux-5.8.1$ nano arch/x86/entry/syscalls/syscall_64.tbl
sama@sama-virtual-machine:~/linux-5.8.1$
```



You will find a series of x32 system calls. Scroll to this section

440 common identity sys_identity

above it.

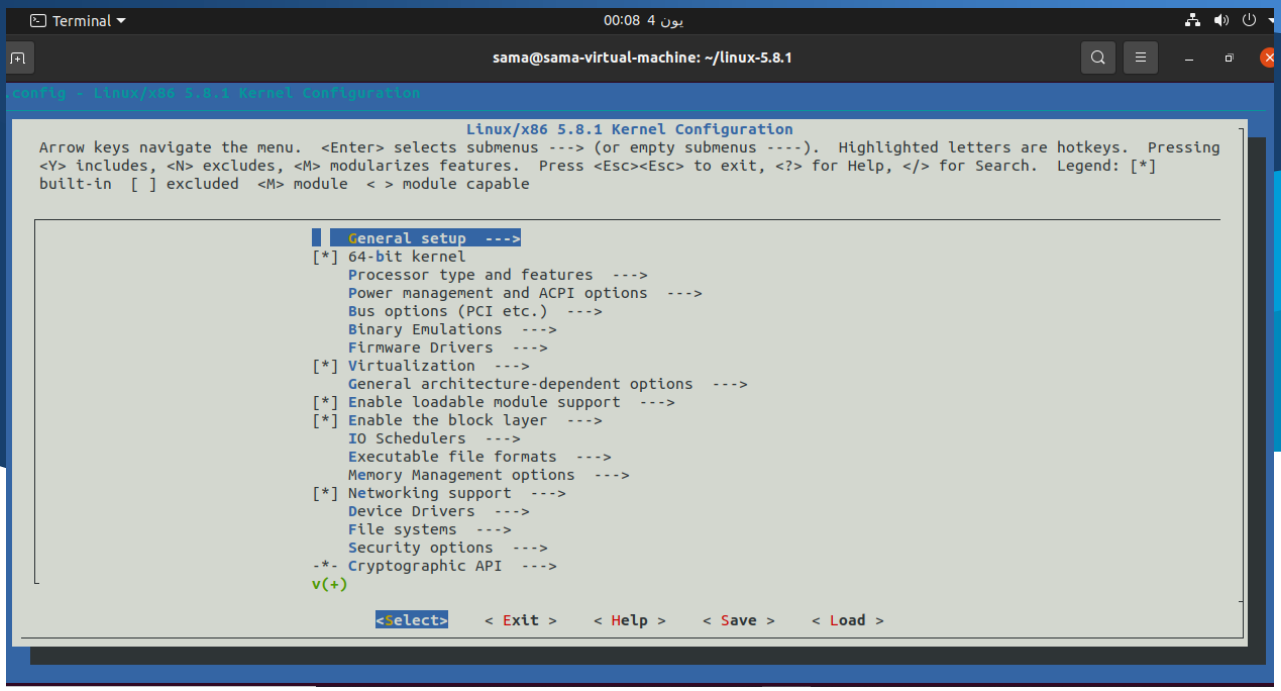


3 - Installation

3.1 - Configure the kernel.
configuration window with the following command.

make menuconfig

```
sama@sama-virtual-machine:~/linux-5.8.1$ make menuconfig
  HOSTCC  scripts/basic/fixdep
  UPD     scripts/kconfig/mconf-cfg
  HOSTCC  scripts/kconfig/mconf.o
  HOSTCC  scripts/kconfig/lxdialog/checklist.o
  HOSTCC  scripts/kconfig/lxdialog/inputbox.o
  HOSTCC  scripts/kconfig/lxdialog/menubox.o
  HOSTCC  scripts/kconfig/lxdialog/textbox.o
  HOSTCC  scripts/kconfig/lxdialog/util.o
  HOSTCC  scripts/kconfig/lxdialog/yesno.o
  HOSTCC  scripts/kconfig/confdata.o
  HOSTCC  scripts/kconfig/expr.o
  LEX     scripts/kconfig/lexer.lex.c
  YACC    scripts/kconfig/parser.tab.[ch]
  HOSTCC  scripts/kconfig/lexer.lex.o
  HOSTCC  scripts/kconfig/parser.tab.o
  HOSTCC  scripts/kconfig/preprocess.o
  HOSTCC  scripts/kconfig/symbol.o
  HOSTCC  scripts/kconfig/util.o
  HOSTLD  scripts/kconfig/mconf
scripts/kconfig/mconf  Kconfig
#
# using defaults found in /boot/config-5.8.0-53-generic
#
/boot/config-5.8.0-53-generic:8468:warning: symbol value 'm' invalid for ASHMEM
/boot/config-5.8.0-53-generic:9477:warning: symbol value 'm' invalid for ANDROID_BINDER_IPC
/boot/config-5.8.0-53-generic:9478:warning: symbol value 'm' invalid for ANDROID_BINDERFS


*** End of the configuration.
*** Execute 'make' to start the build or try 'make help'.
```



3.2 - Find out how many logical cores you have by nproc

3.2 - Find out how many logical cores you have by make -j12

```
sama@sama-virtual-machine:~/linux-5.8.1$ nproc
2
sama@sama-virtual-machine:~/linux-5.8.1$ make -j12
  DESCEND  objtool
  CALL     scripts/atomic/check-atomics.sh
  CALL     scripts/checksyscalls.sh
  CHK      include/generated/compile.h
make[1]: *** No rule to make target 'debian/canonical-certs.pem', needed by 'certs/x509_certificate_list'.  Stop.
make[1]: *** Waiting for unfinished jobs....
make: *** [Makefile:1756: certs] Error 2
make: *** Waiting for unfinished jobs....
  CHK      kernel/kheaders_data.tar.xz
```

3.5 - Install the kernel by sudo make install -j12

4 - Result

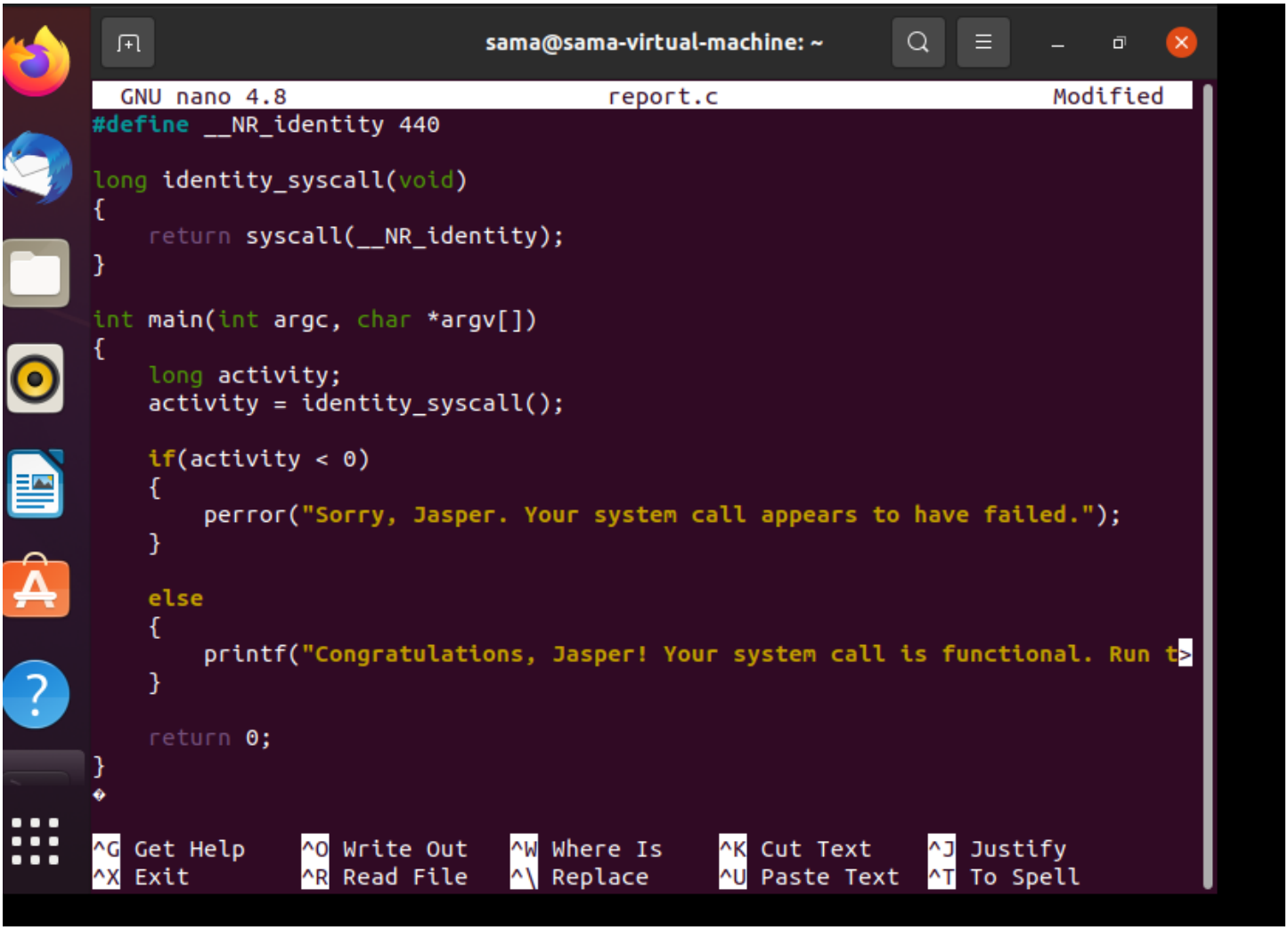
4.1 - Check the version of your current kernel.

4.2 - Change your working directory to your home directory.

4.3 - Create a C file to generate a report of the success or failure of your system call.

```
sama@sama-virtual-machine:~$ uname -r
5.8.0-55-generic
sama@sama-virtual-machine:~$ cd ~
sama@sama-virtual-machine:~$ nano report.c
sama@sama-virtual-machine:~$
```

Write the following code in it.



4.4 - Compile the C file you just created.

gcc -o report report.c

4.5 - Run the C file you just compiled. ./report

```
sama@sama-virtual-machine:~$ ./report
Congratulations, Jasper! Your system call is functional. Run the command dmesg
in the terminal and find out!
sama@sama-virtual-machine:~$
```

References

https://www.kernel.org/doc/html/latest/process/adding-syscalls.html

https://dev.to/jasper/adding-a-system-call-to-the-linux-kernel-5-8-1-in-ubuntu-20-04-lts-2ga8

https://medium.com/anubhav-shrimal/adding-a-hello-world-system-call-to-linux-kernel-dad32875872