Adding and testing a new system call to Linux kernel - Report

RollNo: 2022201009 **Name: Jeet Shah**

AOS-Assignment-2

STEPS PERFORMED TO MAKE THE ASSIGNMENT:-

1. First step is to download the kernel using linux-4.19.210.tar-1.gz file.
2. Now moving to the /usr/bin/linux-4.19.210/ path make one folder
3. Now make one ".c" file where we need to the write the code to be performed when system call is called.
4. Now make one file named "Makefile" where we will write code for ensuring that, The ".c" file is compiled and included into the kernel source code.
5. Now go back to the parent directory and then add the location of the folder into "Makefile".
6. Now go to ~/usr/bin/linux-4.19.210/arch/x86/entry/syscalls/ and make an entry into the syscall_64.tbl, individually for each new ".c" file.

Install packages first:

```
sudo apt-get install gcc
sudo apt-get install libncurses5-dev
sudo apt-get install bison
sudo apt-get install flex
sudo apt-get install libssl-dev
sudo apt-get install libelf-dev
sudo apt-get update
sudo apt-get upgrade
```

7. Now next time is to configure the kernel using the "sudo make menuconfig" command.
8. Now compile the kernel using "sudo make".
9. Now update the kernel using the "sudo make modules_install install" command.
10. Reboot the VM using the "shutdown -r now" command.
11. Testing the system call by making one ".c" file to call the system call made.
12. Now compile this file and run using ./a.out the output message will be printed.
13. Now type "dmesg" command to see the output of the system call invoked.

References:

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

2. https://www.stolaf.edu/people/rab/os/lab/newsyscall.html

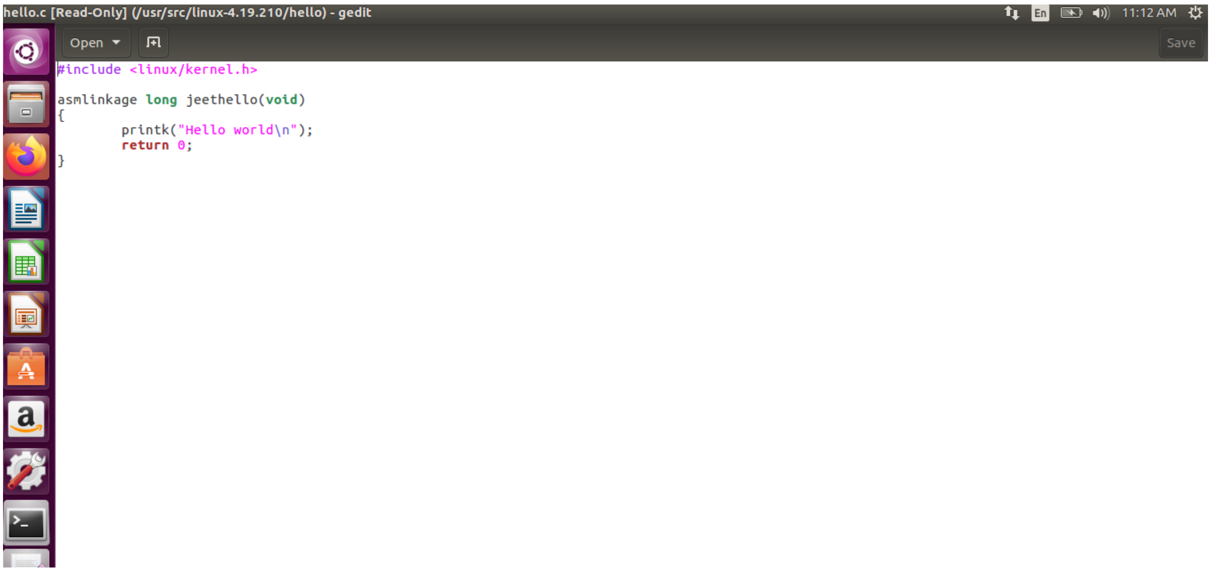
printk() → This function is used to print the kernel logs.

Adding and testing a new system call to Linux kernel - Report

RollNo: 2022201009 **Name: Jeet Shah**

Q1

STEP-1: mkdir hello
STEP-2: cd hello
STEP-3: gedit hello.c
Desc: In this file we have to write the steps that need to performed when the system call is invoked.

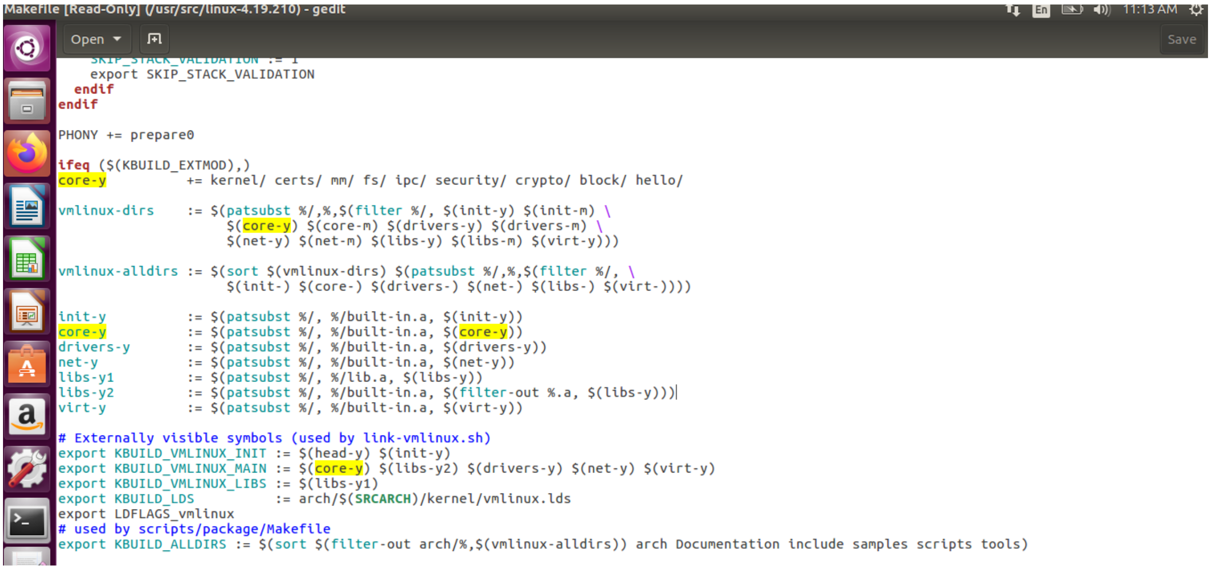


hello.c [Read-Only] (/usr/src/linux-4.19.210/hello) - gedit

```
#include <linux/kernel.h>

asmlinkage long jeethello(void)
{
        printk("Hello world\n");
        return 0;
}
```

STEP-4: gedit Makefile
Makefile content: obj-y := hello.o
Desc: These step is to ensure that the compiled object file of hello.c is included into the Kernel source code.

STEP-5: cd ../Makefile
Desc: These step is to add the folder path and tell the kernel that where our new system Is placed.

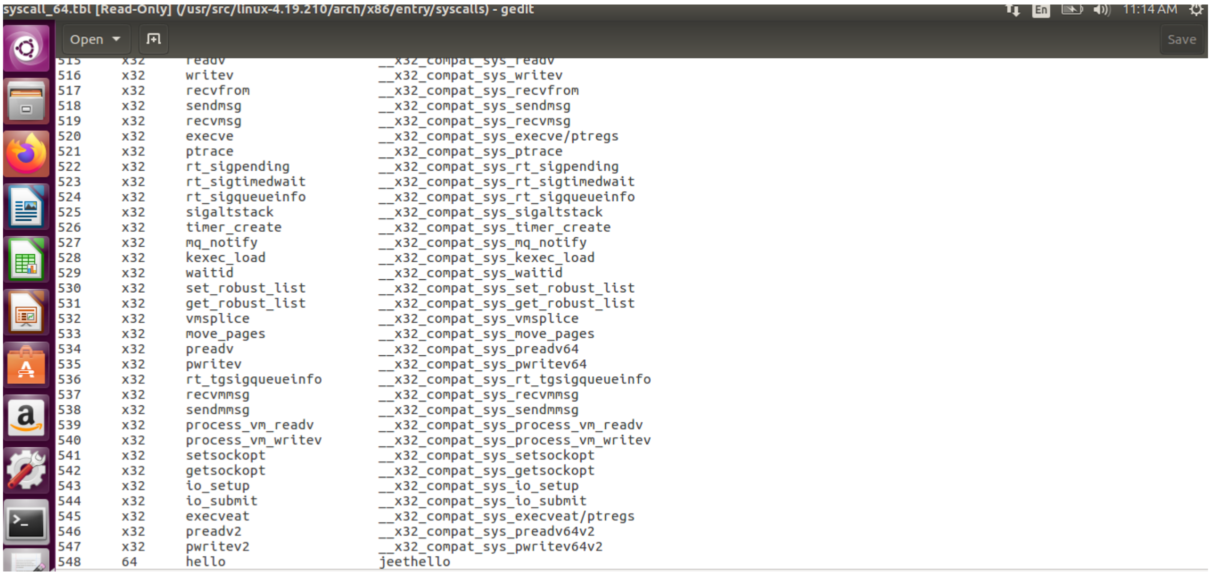


Makefile [Read-Only] (/usr/src/linux-4.19.210) - gedit

```
    SKIP_STACK_VALIDATION := 1
    export SKIP_STACK_VALIDATION
  endif
endif

PHONY += prepare0

ifeq ($(KBUILD_EXTMOD),)
core-y          += kernel/ certs/ mm/ fs/ ipc/ security/ crypto/ block/ hello/

vmlinux-dirs    := $(patsubst %/,%,$(filter %/, $(init-y) $(init-m) \
                     $(core-y) $(core-m) $(drivers-y) $(drivers-m) \
                     $(net-y) $(net-m) $(libs-y) $(libs-m) $(virt-y)))

vmlinux-alldirs := $(sort $(vmlinux-dirs) $(patsubst %/,%,$(filter %/, \
                     $(init-) $(core-) $(drivers-) $(net-) $(libs-) $(virt-))))

init-y          := $(patsubst %/, %/built-in.a, $(init-y))
core-y          := $(patsubst %/, %/built-in.a, $(core-y))
drivers-y       := $(patsubst %/, %/built-in.a, $(drivers-y))
net-y           := $(patsubst %/, %/built-in.a, $(net-y))
libs-y1         := $(patsubst %/, %/lib.a, $(libs-y))
libs-y2         := $(patsubst %/, %/built-in.a, $(filter-out %.a, $(libs-y)))
virt-y          := $(patsubst %/, %/built-in.a, $(virt-y))

# Externally visible symbols (used by link-vmlinux.sh)
export KBUILD_VMLINUX_INIT := $(head-y) $(init-y)
export KBUILD_VMLINUX_MAIN := $(core-y) $(libs-y2) $(drivers-y) $(net-y) $(virt-y)
export KBUILD_VMLINUX_LIBS := $(libs-y1)
export KBUILD_LDS          := arch/$(SRCARCH)/kernel/vmlinux.lds
export LDFLAGS_vmlinux
# used by scripts/package/Makefile
export KBUILD_ALLDIRS := $(sort $(filter-out arch/%,$(vmlinux-alldirs)) arch Documentation include samples scripts tools)
```

STEP-6: Make entry inside syscall_64.tbl

Desc: This entry is used to map the system call number “548” to the “jeethello” system call.

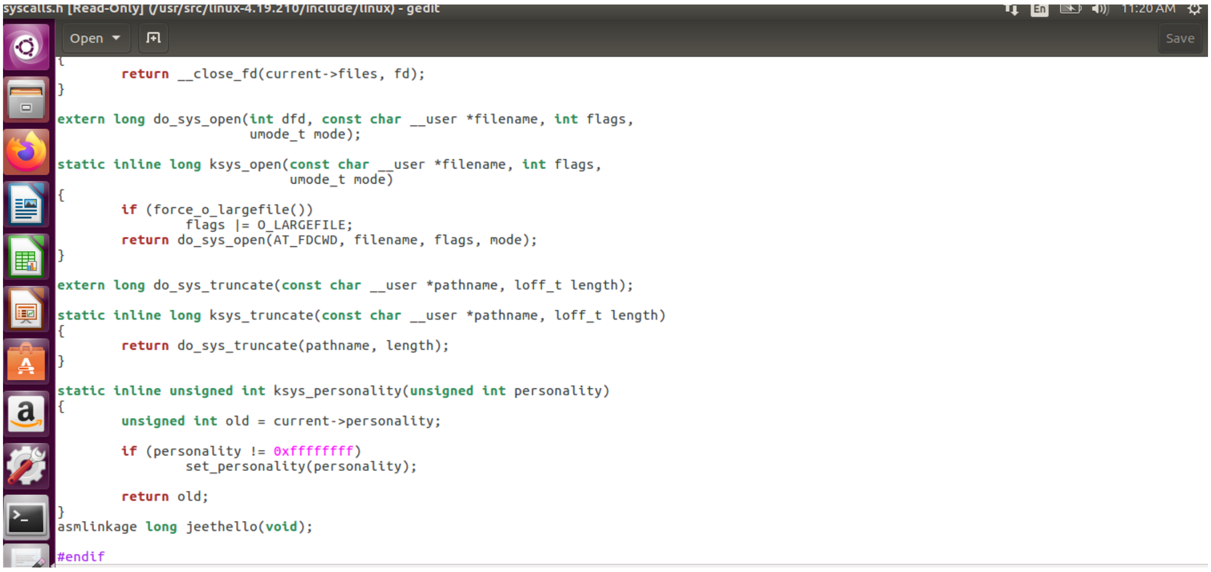


syscall_64.tbl [Read-Only] (/usr/src/linux-4.19.210/arch/x86/entry/syscalls) - gedit
11:14 AM
Open
Save

```
515     x32     readv                   __x32_compat_sys_readv
516     x32     writev                  __x32_compat_sys_writev
517     x32     recvfrom                __x32_compat_sys_recvfrom
518     x32     sendmsg                 __x32_compat_sys_sendmsg
519     x32     recvmsg                 __x32_compat_sys_recvmsg
520     x32     execve                  __x32_compat_sys_execve/ptregs
521     x32     ptrace                  __x32_compat_sys_ptrace
522     x32     rt_sigpending           __x32_compat_sys_rt_sigpending
523     x32     rt_sigtimedwait         __x32_compat_sys_rt_sigtimedwait
524     x32     rt_sigqueueinfo         __x32_compat_sys_rt_sigqueueinfo
525     x32     sigaltstack             __x32_compat_sys_sigaltstack
526     x32     timer_create            __x32_compat_sys_timer_create
527     x32     mq_notify               __x32_compat_sys_mq_notify
528     x32     kexec_load              __x32_compat_sys_kexec_load
529     x32     waitid                  __x32_compat_sys_waitid
530     x32     set_robust_list         __x32_compat_sys_set_robust_list
531     x32     get_robust_list         __x32_compat_sys_get_robust_list
532     x32     vmsplice                __x32_compat_sys_vmsplice
533     x32     move_pages              __x32_compat_sys_move_pages
534     x32     preadv                  __x32_compat_sys_preadv64
535     x32     pwritev                 __x32_compat_sys_pwritev64
536     x32     rt_tgsigqueueinfo       __x32_compat_sys_rt_tgsigqueueinfo
537     x32     recvmmsg                __x32_compat_sys_recvmmsg
538     x32     sendmmsg                __x32_compat_sys_sendmmsg
539     x32     process_vm_readv        __x32_compat_sys_process_vm_readv
540     x32     process_vm_writev       __x32_compat_sys_process_vm_writev
541     x32     setsockopt              __x32_compat_sys_setsockopt
542     x32     getsockopt              __x32_compat_sys_getsockopt
543     x32     io_setup                __x32_compat_sys_io_setup
544     x32     io_submit               __x32_compat_sys_io_submit
545     x32     execveat                __x32_compat_sys_execveat/ptregs
546     x32     preadv2                 __x32_compat_sys_preadv64v2
547     x32     pwritev2                __x32_compat_sys_pwritev64v2
548     64      hello                   jeethello
```

STEP-7: Make entry inside syscalls.h

Desc: This line added defines the prototype of the system call jeethello.



syscalls.h [Read-Only] (/usr/src/linux-4.19.210/include/linux) - gedit
11:20 AM
Open
Save

```
{
        return __close_fd(current->files, fd);
}

extern long do_sys_open(int dfd, const char __user *filename, int flags,
                        umode_t mode);

static inline long ksys_open(const char __user *filename, int flags,
                             umode_t mode)
{
        if (force_o_largefile())
                flags |= O_LARGEFILE;
        return do_sys_open(AT_FDCWD, filename, flags, mode);
}

extern long do_sys_truncate(const char __user *pathname, loff_t length);

static inline long ksys_truncate(const char __user *pathname, loff_t length)
{
        return do_sys_truncate(pathname, length);
}

static inline unsigned int ksys_personality(unsigned int personality)
{
        unsigned int old = current->personality;

        if (personality != 0xffffffff)
                set_personality(personality);

        return old;
}
asmlinkage long jeethello(void);

#endif
```

STEP-8: sudo make menuconfig

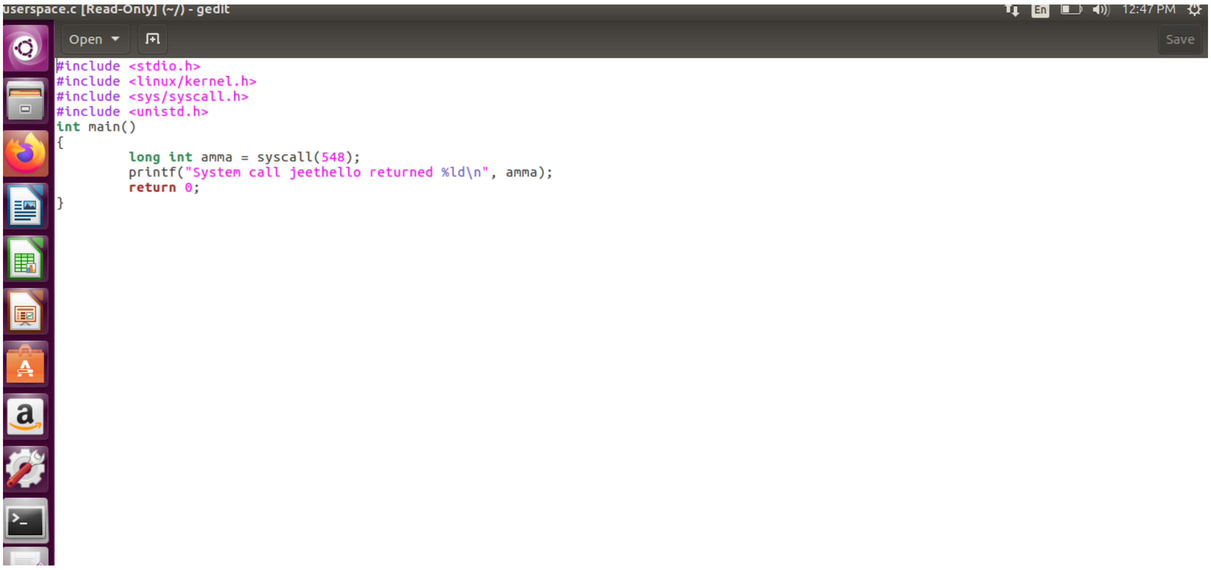
STEP-9: sudo make — To compile the code

STEP-10: sudo make modules_install install

STEP-11: shutdown -r now

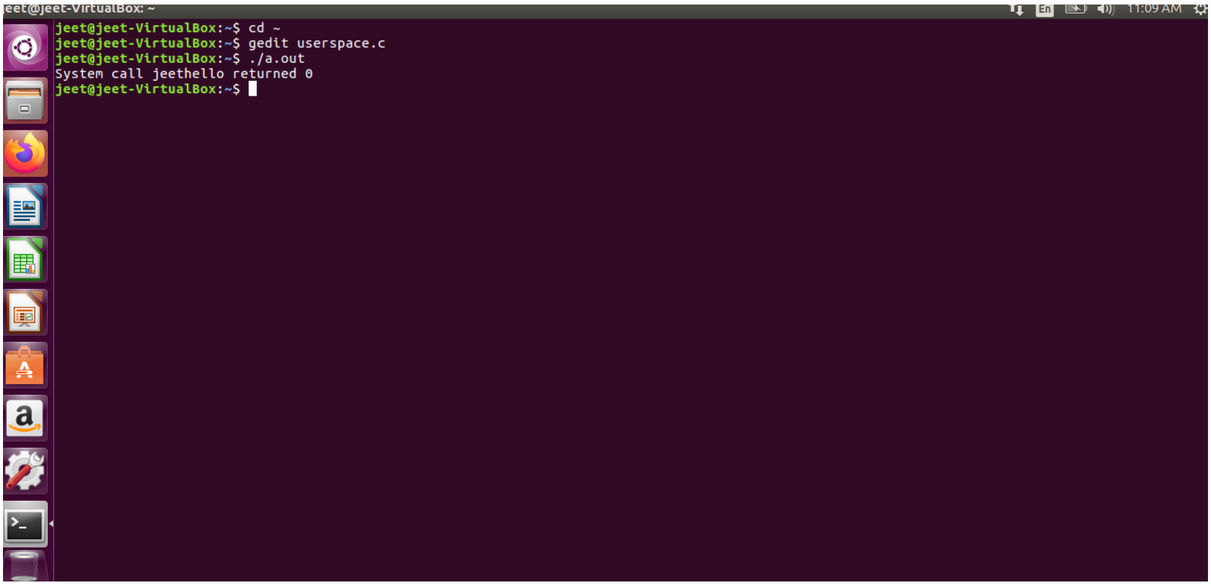
STEP-12: gedit userspace.c

Desc: Here the code for calling the system call is written by giving the reference of the system call number(entry number of syscall_64.tbl).



```
#include <stdio.h>
#include <linux/kernel.h>
#include <sys/syscall.h>
#include <unistd.h>
int main()
{
         long int amma = syscall(548);
         printf("System call jeethello returned %ld\n", amma);
         return 0;
}
```

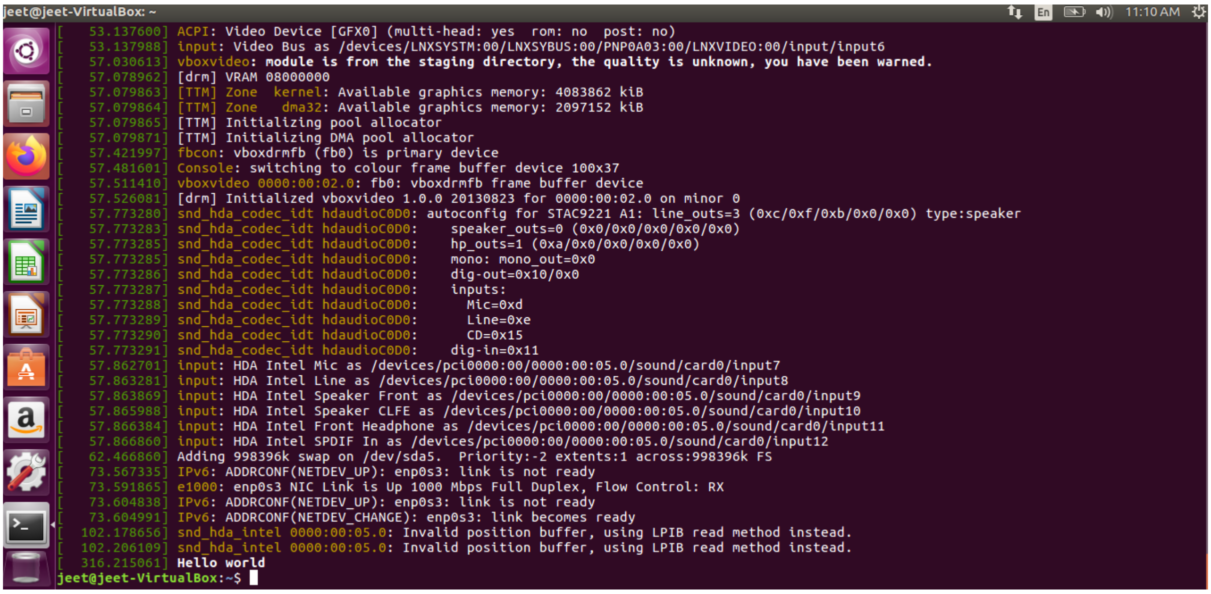
STEP-13: gcc userspace.c
./a.out



```
jeet@jeet-VirtualBox:~$ cd ~
jeet@jeet-VirtualBox:~$ gedit userspace.c
jeet@jeet-VirtualBox:~$ ./a.out
System call jeethello returned 0
jeet@jeet-VirtualBox:~$ 
```

STEP-14: dmesg

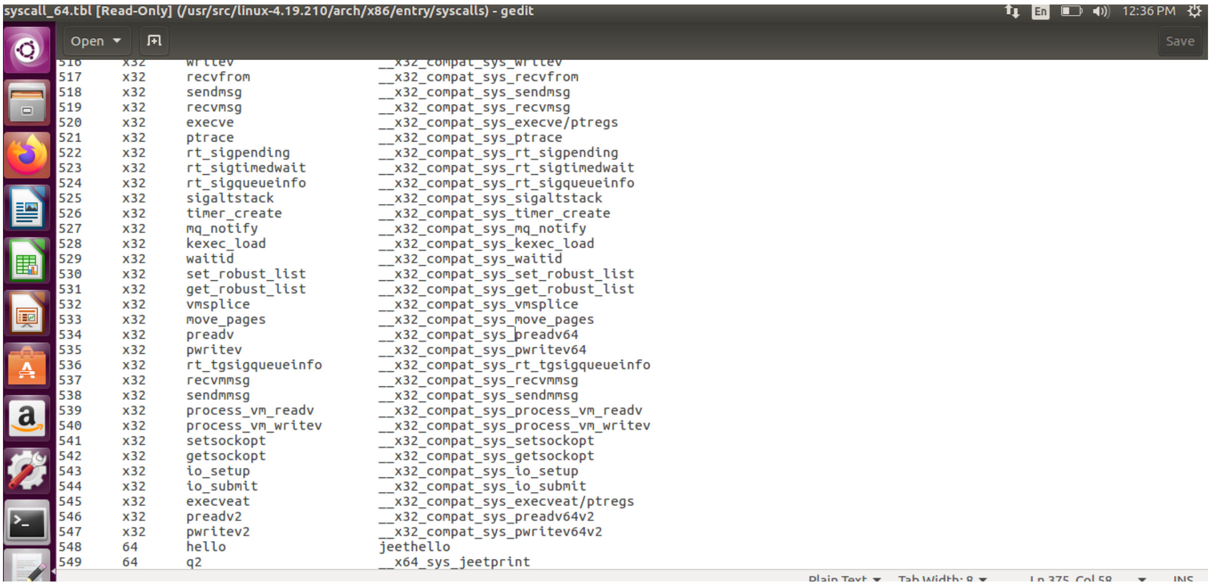
Desc: These command is used to print the kernel logs.



```
[   53.137600] ACPI: Video Device [GFX0] (multi-head: yes  rom: no  post: no)
[   53.137988] input: Video Bus as /devices/LNXSYSTM:00/LNXSYBUS:00/PNP0A03:00/LNXVIDEO:00/input/input6
[   57.030613] vboxvideo: module is from the staging directory, the quality is unknown, you have been warned.
[   57.078962] [drm] VRAM 08000000
[   57.079863] [TTM] Zone  kernel: Available graphics memory: 4083862 kiB
[   57.079864] [TTM] Zone   dma32: Available graphics memory: 2097152 kiB
[   57.079865] [TTM] Initializing pool allocator
[   57.079871] [TTM] Initializing DMA pool allocator
[   57.421997] fbcon: vboxdrmfb (fb0) is primary device
[   57.481601] Console: switching to colour frame buffer device 100x37
[   57.511410] vboxvideo 0000:00:02.0: fb0: vboxdrmfb frame buffer device
[   57.526081] [drm] Initialized vboxvideo 1.0.0 20130823 for 0000:00:02.0 on minor 0
[   57.773280] snd_hda_codec_idt hdaudioC0D0: autoconfig for STAC9221 A1: line_outs=3 (0xc/0xf/0xb/0x0/0x0) type:speaker
[   57.773283] snd_hda_codec_idt hdaudioC0D0:    speaker_outs=0 (0x0/0x0/0x0/0x0/0x0)
[   57.773285] snd_hda_codec_idt hdaudioC0D0:    hp_outs=1 (0xa/0x0/0x0/0x0/0x0)
[   57.773285] snd_hda_codec_idt hdaudioC0D0:    mono: mono_out=0x0
[   57.773286] snd_hda_codec_idt hdaudioC0D0:    dig-out=0x10/0x0
[   57.773287] snd_hda_codec_idt hdaudioC0D0:    inputs:
[   57.773288] snd_hda_codec_idt hdaudioC0D0:      Mic=0xd
[   57.773289] snd_hda_codec_idt hdaudioC0D0:      Line=0xe
[   57.773290] snd_hda_codec_idt hdaudioC0D0:      CD=0x15
[   57.773291] snd_hda_codec_idt hdaudioC0D0:    dig-in=0x11
[   57.862701] input: HDA Intel Mic as /devices/pci0000:00/0000:00:05.0/sound/card0/input7
[   57.863281] input: HDA Intel Line as /devices/pci0000:00/0000:00:05.0/sound/card0/input8
[   57.863869] input: HDA Intel Speaker Front as /devices/pci0000:00/0000:00:05.0/sound/card0/input9
[   57.865988] input: HDA Intel Speaker CLFE as /devices/pci0000:00/0000:00:05.0/sound/card0/input10
[   57.866384] input: HDA Intel Front Headphone as /devices/pci0000:00/0000:00:05.0/sound/card0/input11
[   57.866860] input: HDA Intel SPDIF In as /devices/pci0000:00/0000:00:05.0/sound/card0/input12
[   62.466860] Adding 998396k swap on /dev/sda5.  Priority:-2 extents:1 across:998396k FS
[   73.567335] IPv6: ADDRCONF(NETDEV_UP): enp0s3: link is not ready
[   73.591865] e1000: enp0s3 NIC Link is Up 1000 Mbps Full Duplex, Flow Control: RX
[   73.604838] IPv6: ADDRCONF(NETDEV_UP): enp0s3: link is not ready
[   73.604991] IPv6: ADDRCONF(NETDEV_CHANGE): enp0s3: link becomes ready
[  102.178656] snd_hda_intel 0000:00:05.0: Invalid position buffer, using LPIB read method instead.
[  102.206109] snd_hda_intel 0000:00:05.0: Invalid position buffer, using LPIB read method instead.
[  316.215061] Hello world
jeet@jeet-VirtualBox:~$ 
```

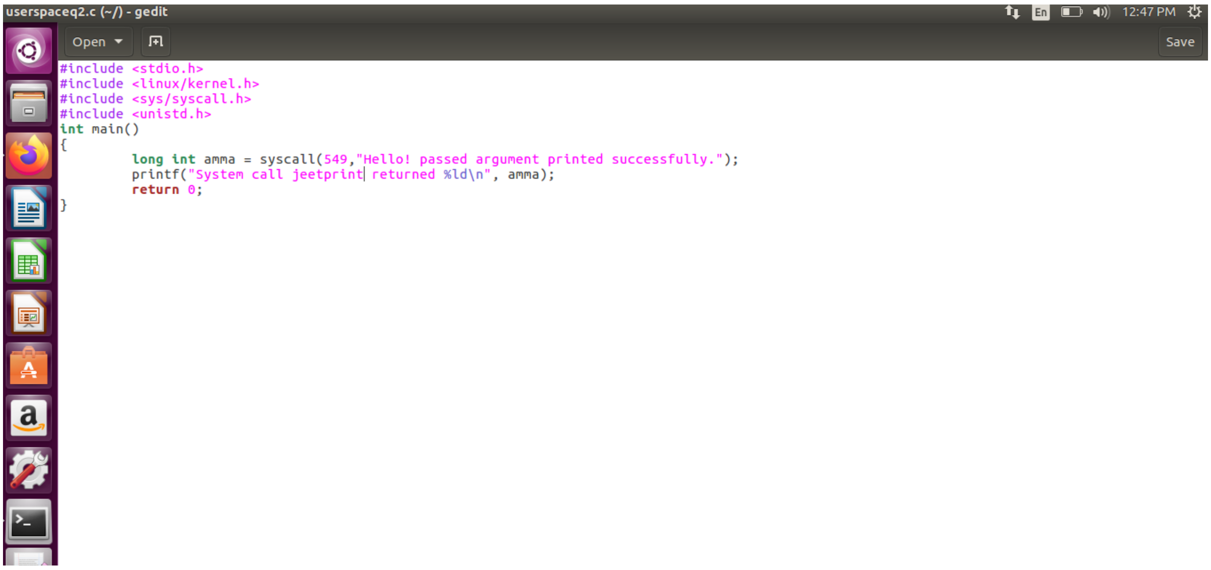
Q2 – All the steps of Q1 are the same, just that step - 7 is to be skipped here.

STEP-6: Make entry inside syscall_64.tbl


syscall_64.tbl [Read-Only] (/usr/src/linux-4.19.210/arch/x86/entry/syscalls) - gedit
En
12:36 PM
Open ▾
Save
Plain Text ▾
Tab Width: 8 ▾
Ln 375, Col 58
INS

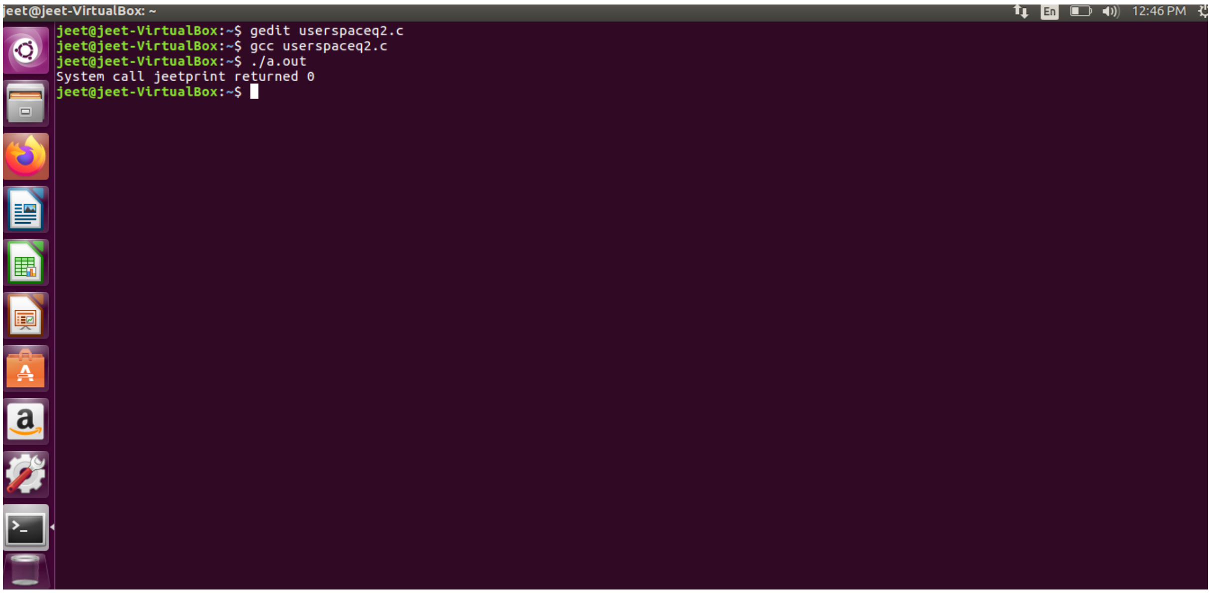
```
516     x32     writev                  __x32_compat_sys_writev
517     x32     recvfrom                __x32_compat_sys_recvfrom
518     x32     sendmsg                 __x32_compat_sys_sendmsg
519     x32     recvmsg                 __x32_compat_sys_recvmsg
520     x32     execve                  __x32_compat_sys_execve/ptregs
521     x32     ptrace                  __x32_compat_sys_ptrace
522     x32     rt_sigpending           __x32_compat_sys_rt_sigpending
523     x32     rt_sigtimedwait         __x32_compat_sys_rt_sigtimedwait
524     x32     rt_sigqueueinfo         __x32_compat_sys_rt_sigqueueinfo
525     x32     sigaltstack             __x32_compat_sys_sigaltstack
526     x32     timer_create            __x32_compat_sys_timer_create
527     x32     mq_notify               __x32_compat_sys_mq_notify
528     x32     kexec_load              __x32_compat_sys_kexec_load
529     x32     waitid                  __x32_compat_sys_waitid
530     x32     set_robust_list         __x32_compat_sys_set_robust_list
531     x32     get_robust_list         __x32_compat_sys_get_robust_list
532     x32     vmsplice                __x32_compat_sys_vmsplice
533     x32     move_pages              __x32_compat_sys_move_pages
534     x32     preadv                  __x32_compat_sys_preadv64
535     x32     pwritev                 __x32_compat_sys_pwritev64
536     x32     rt_tgsigqueueinfo       __x32_compat_sys_rt_tgsigqueueinfo
537     x32     recvmmsg                __x32_compat_sys_recvmmsg
538     x32     sendmmsg                __x32_compat_sys_sendmmsg
539     x32     process_vm_readv        __x32_compat_sys_process_vm_readv
540     x32     process_vm_writev       __x32_compat_sys_process_vm_writev
541     x32     setsockopt              __x32_compat_sys_setsockopt
542     x32     getsockopt              __x32_compat_sys_getsockopt
543     x32     io_setup                __x32_compat_sys_io_setup
544     x32     io_submit               __x32_compat_sys_io_submit
545     x32     execveat                __x32_compat_sys_execveat/ptregs
546     x32     preadv2                 __x32_compat_sys_preadv64v2
547     x32     pwritev2                __x32_compat_sys_pwritev64v2
548     64      hello                   jeethello
549     64      q2                      __x64_sys_jeetprint
```

STEP-12: gedit userspaceq2.c



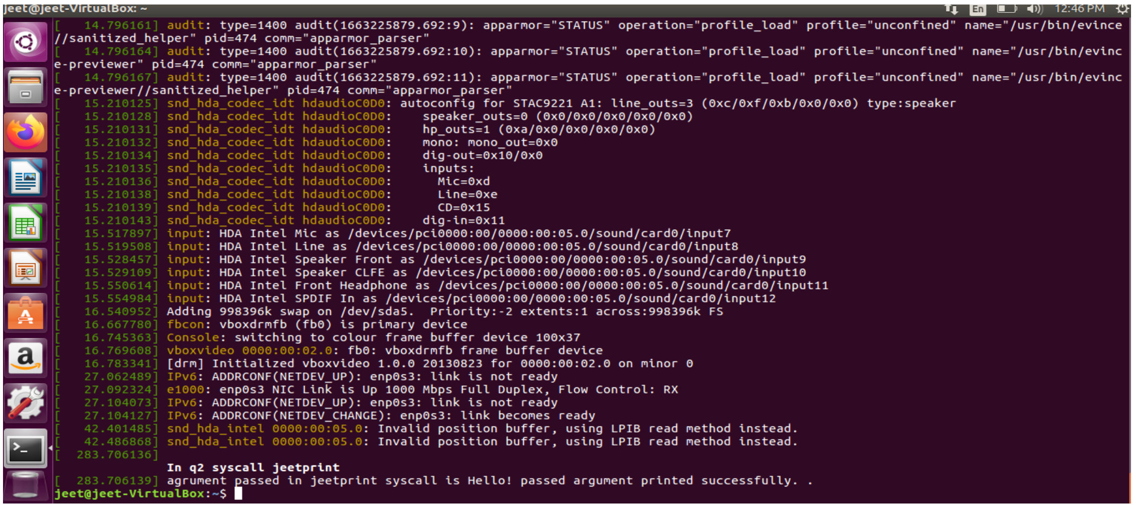
```
#include <stdio.h>
#include <linux/kernel.h>
#include <sys/syscall.h>
#include <unistd.h>
int main()
{
        long int amma = syscall(549,"Hello! passed argument printed successfully.");
        printf("System call jeetprint returned %ld\n", amma);
        return 0;
}
```

STEP-13: gcc userspaceq2.c
./a.out



```
jeet@jeet-VirtualBox:~$ gedit userspaceq2.c
jeet@jeet-VirtualBox:~$ gcc userspaceq2.c
jeet@jeet-VirtualBox:~$ ./a.out
System call jeetprint returned 0
jeet@jeet-VirtualBox:~$ 
```

STEP-14: dmesg

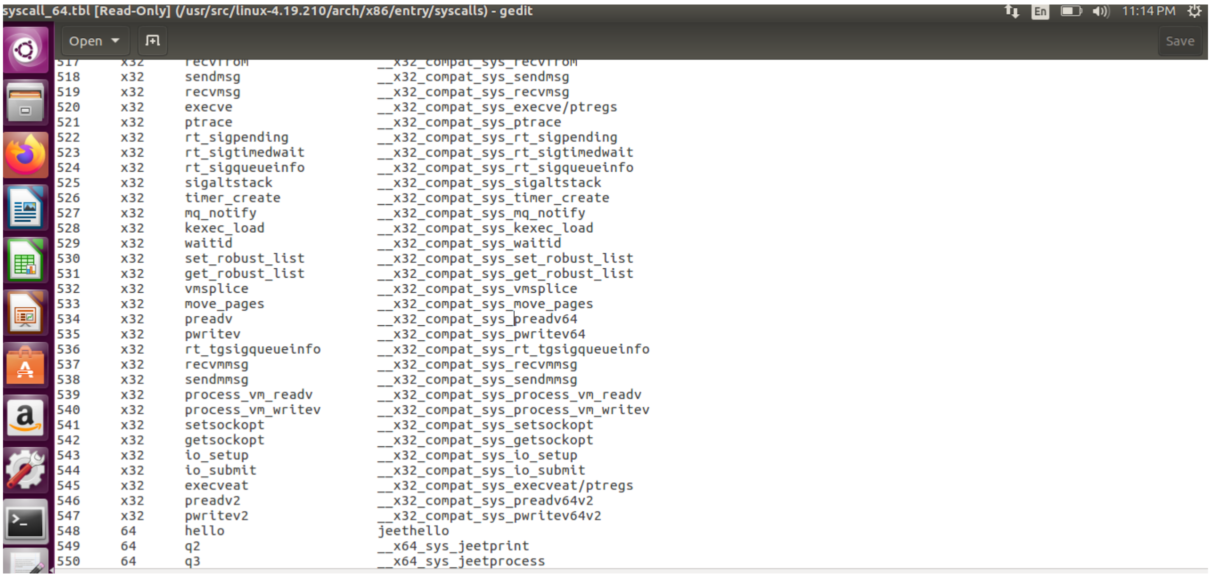


```
[   14.796161] audit: type=1400 audit(1663225879.692:9): apparmor="STATUS" operation="profile_load" profile="unconfined" name="/usr/bin/evince
//sanitized_helper" pid=474 comm="apparmor_parser"
[   14.796164] audit: type=1400 audit(1663225879.692:10): apparmor="STATUS" operation="profile_load" profile="unconfined" name="/usr/bin/evinc
e-previewer" pid=474 comm="apparmor_parser"
[   14.796167] audit: type=1400 audit(1663225879.692:11): apparmor="STATUS" operation="profile_load" profile="unconfined" name="/usr/bin/evinc
e-previewer//sanitized_helper" pid=474 comm="apparmor_parser"
[   15.210125] snd_hda_codec_idt hdaudioC0D0: autoconfig for STAC9221 A1: line_outs=3 (0xc/0xf/0xb/0x0/0x0) type:speaker
[   15.210128] snd_hda_codec_idt hdaudioC0D0:    speaker_outs=0 (0x0/0x0/0x0/0x0/0x0)
[   15.210131] snd_hda_codec_idt hdaudioC0D0:    hp_outs=1 (0xa/0x0/0x0/0x0/0x0)
[   15.210132] snd_hda_codec_idt hdaudioC0D0:    mono: mono_out=0x0
[   15.210134] snd_hda_codec_idt hdaudioC0D0:    dig-out=0x10/0x0
[   15.210135] snd_hda_codec_idt hdaudioC0D0:    inputs:
[   15.210136] snd_hda_codec_idt hdaudioC0D0:      Mic=0xd
[   15.210138] snd_hda_codec_idt hdaudioC0D0:      Line=0xe
[   15.210139] snd_hda_codec_idt hdaudioC0D0:      CD=0x15
[   15.210143] snd_hda_codec_idt hdaudioC0D0:    dig-in=0x11
[   15.517897] input: HDA Intel Mic as /devices/pci0000:00/0000:00:05.0/sound/card0/input7
[   15.519508] input: HDA Intel Line as /devices/pci0000:00/0000:00:05.0/sound/card0/input8
[   15.528457] input: HDA Intel Speaker Front as /devices/pci0000:00/0000:00:05.0/sound/card0/input9
[   15.529109] input: HDA Intel Speaker CLFE as /devices/pci0000:00/0000:00:05.0/sound/card0/input10
[   15.550614] input: HDA Intel Front Headphone as /devices/pci0000:00/0000:00:05.0/sound/card0/input11
[   15.554984] input: HDA Intel SPDIF In as /devices/pci0000:00/0000:00:05.0/sound/card0/input12
[   16.540952] Adding 998396k swap on /dev/sda5.  Priority:-2 extents:1 across:998396k FS
[   16.667780] fbcon: vboxdrmfb (fb0) is primary device
[   16.745363] Console: switching to colour frame buffer device 100x37
[   16.769608] vboxvideo 0000:00:02.0: fb0: vboxdrmfb frame buffer device
[   16.783341] [drm] Initialized vboxvideo 1.0.0 20130823 for 0000:00:02.0 on minor 0
[   27.062489] IPv6: ADDRCONF(NETDEV_UP): enp0s3: link is not ready
[   27.092324] e1000: enp0s3 NIC Link is Up 1000 Mbps Full Duplex, Flow Control: RX
[   27.104073] IPv6: ADDRCONF(NETDEV_UP): enp0s3: link is not ready
[   27.104127] IPv6: ADDRCONF(NETDEV_CHANGE): enp0s3: link becomes ready
[   42.401485] snd_hda_intel 0000:00:05.0: Invalid position buffer, using LPIB read method instead.
[   42.486868] snd_hda_intel 0000:00:05.0: Invalid position buffer, using LPIB read method instead.
[  283.706136]
               In q2 syscall jeetprint
[  283.706139] agrument passed in jeetprint syscall is Hello! passed argument printed successfully. .
jeet@jeet-VirtualBox:~$
```

Q3 All the steps of Q1 are the same, just that step - 7 is to be skipped here.

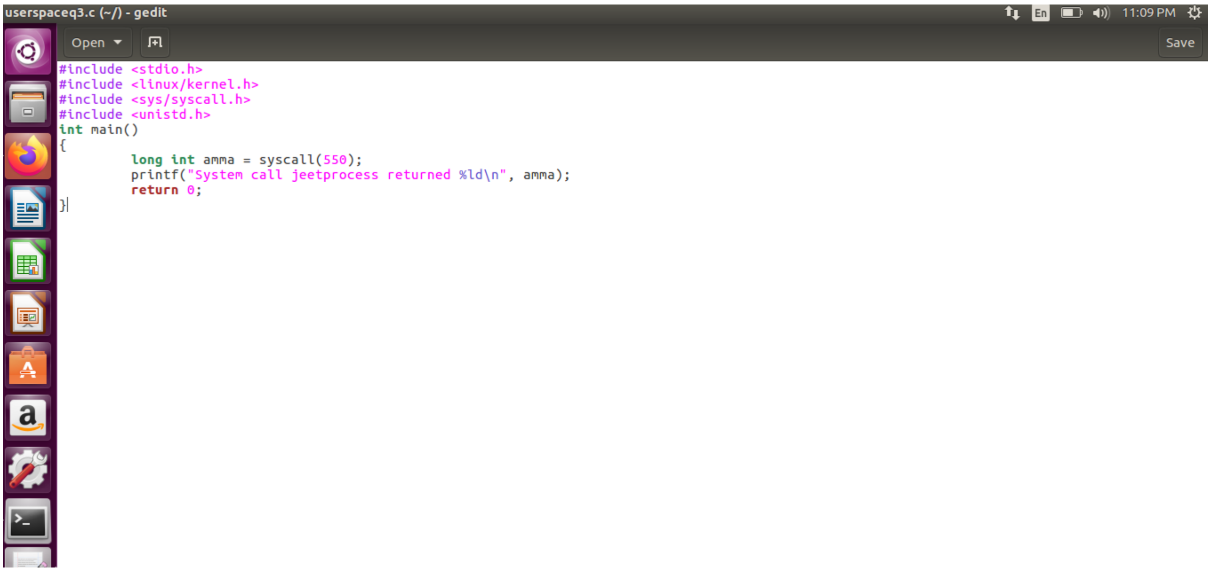
Ques: Are both process ids different or same? Why? Justify.
Sol: Yes, both are different. Until the terminal is open the process id is same as kernel is already having the PCB stored. But for child processes every time a new child process is created and new child process id is generated.

STEP-6: Make entry inside syscall_64.tbl


syscall_64.tbl [Read-Only] (/usr/src/linux-4.19.210/arch/x86/entry/syscalls) - gedit
En
11:14 PM
Open
Save

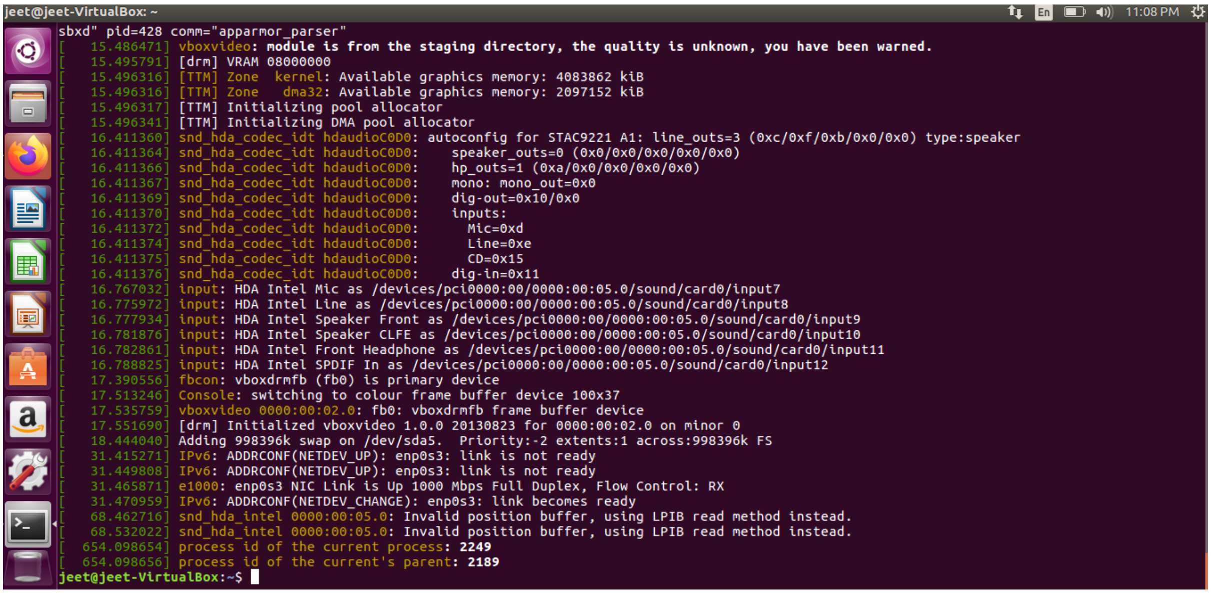
```
517     x32     recvfrom                __x32_compat_sys_recvfrom
518     x32     sendmsg                 __x32_compat_sys_sendmsg
519     x32     recvmsg                 __x32_compat_sys_recvmsg
520     x32     execve                  __x32_compat_sys_execve/ptregs
521     x32     ptrace                  __x32_compat_sys_ptrace
522     x32     rt_sigpending           __x32_compat_sys_rt_sigpending
523     x32     rt_sigtimedwait         __x32_compat_sys_rt_sigtimedwait
524     x32     rt_sigqueueinfo         __x32_compat_sys_rt_sigqueueinfo
525     x32     sigaltstack             __x32_compat_sys_sigaltstack
526     x32     timer_create            __x32_compat_sys_timer_create
527     x32     mq_notify               __x32_compat_sys_mq_notify
528     x32     kexec_load              __x32_compat_sys_kexec_load
529     x32     waitid                  __x32_compat_sys_waitid
530     x32     set_robust_list         __x32_compat_sys_set_robust_list
531     x32     get_robust_list         __x32_compat_sys_get_robust_list
532     x32     vmsplice                __x32_compat_sys_vmsplice
533     x32     move_pages              __x32_compat_sys_move_pages
534     x32     preadv                  __x32_compat_sys_preadv64
535     x32     pwritev                 __x32_compat_sys_pwritev64
536     x32     rt_tgsigqueueinfo       __x32_compat_sys_rt_tgsigqueueinfo
537     x32     recvmmsg                __x32_compat_sys_recvmmsg
538     x32     sendmmsg                __x32_compat_sys_sendmmsg
539     x32     process_vm_readv        __x32_compat_sys_process_vm_readv
540     x32     process_vm_writev       __x32_compat_sys_process_vm_writev
541     x32     setsockopt              __x32_compat_sys_setsockopt
542     x32     getsockopt              __x32_compat_sys_getsockopt
543     x32     io_setup                __x32_compat_sys_io_setup
544     x32     io_submit               __x32_compat_sys_io_submit
545     x32     execveat                __x32_compat_sys_execveat/ptregs
546     x32     preadv2                 __x32_compat_sys_preadv64v2
547     x32     pwritev2                __x32_compat_sys_pwritev64v2
548     64      hello                   jeethello
549     64      q2                      __x64_sys_jeetprint
550     64      q3                      __x64_sys_jeetprocess
```

STEP-12: gedit userspaceq3.c



```
#include <stdio.h>
#include <linux/kernel.h>
#include <sys/syscall.h>
#include <unistd.h>
int main()
{
        long int amma = syscall(550);
        printf("System call jeetprocess returned %ld\n", amma);
        return 0;
}
```

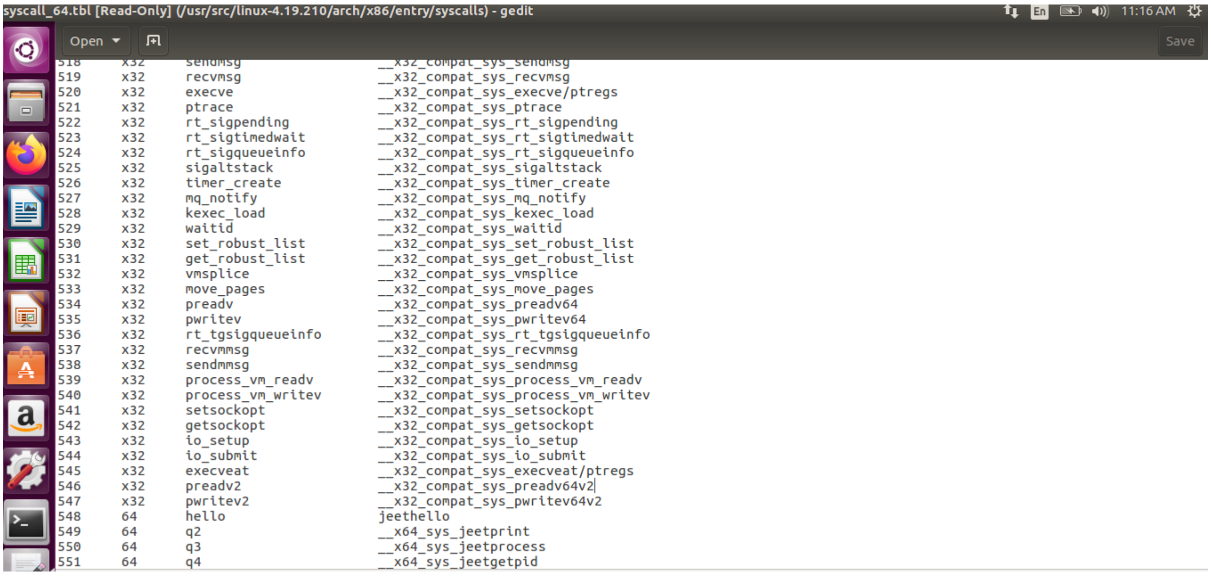
STEP-14: dmesg



```
sbxd" pid=428 comm="apparmor_parser"
[   15.486471] vboxvideo: module is from the staging directory, the quality is unknown, you have been warned.
[   15.495791] [drm] VRAM 08000000
[   15.496316] [TTM] Zone  kernel: Available graphics memory: 4083862 kiB
[   15.496316] [TTM] Zone   dma32: Available graphics memory: 2097152 kiB
[   15.496317] [TTM] Initializing pool allocator
[   15.496341] [TTM] Initializing DMA pool allocator
[   16.411360] snd_hda_codec_idt hdaudioC0D0: autoconfig for STAC9221 A1: line_outs=3 (0xc/0xf/0xb/0x0/0x0) type:speaker
[   16.411364] snd_hda_codec_idt hdaudioC0D0:    speaker_outs=0 (0x0/0x0/0x0/0x0/0x0)
[   16.411366] snd_hda_codec_idt hdaudioC0D0:    hp_outs=1 (0xa/0x0/0x0/0x0/0x0)
[   16.411367] snd_hda_codec_idt hdaudioC0D0:    mono: mono_out=0x0
[   16.411369] snd_hda_codec_idt hdaudioC0D0:    dig-out=0x10/0x0
[   16.411370] snd_hda_codec_idt hdaudioC0D0:    inputs:
[   16.411372] snd_hda_codec_idt hdaudioC0D0:      Mic=0xd
[   16.411374] snd_hda_codec_idt hdaudioC0D0:      Line=0xe
[   16.411375] snd_hda_codec_idt hdaudioC0D0:      CD=0x15
[   16.411376] snd_hda_codec_idt hdaudioC0D0:    dig-in=0x11
[   16.767032] input: HDA Intel Mic as /devices/pci0000:00/0000:00:05.0/sound/card0/input7
[   16.775972] input: HDA Intel Line as /devices/pci0000:00/0000:00:05.0/sound/card0/input8
[   16.777934] input: HDA Intel Speaker Front as /devices/pci0000:00/0000:00:05.0/sound/card0/input9
[   16.781876] input: HDA Intel Speaker CLFE as /devices/pci0000:00/0000:00:05.0/sound/card0/input10
[   16.782861] input: HDA Intel Front Headphone as /devices/pci0000:00/0000:00:05.0/sound/card0/input11
[   16.788825] input: HDA Intel SPDIF In as /devices/pci0000:00/0000:00:05.0/sound/card0/input12
[   17.390556] fbcon: vboxdrmfb (fb0) is primary device
[   17.513246] Console: switching to colour frame buffer device 100x37
[   17.535759] vboxvideo 0000:00:02.0: fb0: vboxdrmfb frame buffer device
[   17.551690] [drm] Initialized vboxvideo 1.0.0 20130823 for 0000:00:02.0 on minor 0
[   18.444040] Adding 998396k swap on /dev/sda5.  Priority:-2 extents:1 across:998396k FS
[   31.415271] IPv6: ADDRCONF(NETDEV_UP): enp0s3: link is not ready
[   31.449808] IPv6: ADDRCONF(NETDEV_UP): enp0s3: link is not ready
[   31.465871] e1000: enp0s3 NIC Link is Up 1000 Mbps Full Duplex, Flow Control: RX
[   31.470959] IPv6: ADDRCONF(NETDEV_CHANGE): enp0s3: link becomes ready
[   68.462716] snd_hda_intel 0000:00:05.0: Invalid position buffer, using LPIB read method instead.
[   68.532022] snd_hda_intel 0000:00:05.0: Invalid position buffer, using LPIB read method instead.
[  654.098654] process id of the current process: 2249
[  654.098656] process id of the current's parent: 2189
jeet@jeet-VirtualBox:~$ 
```

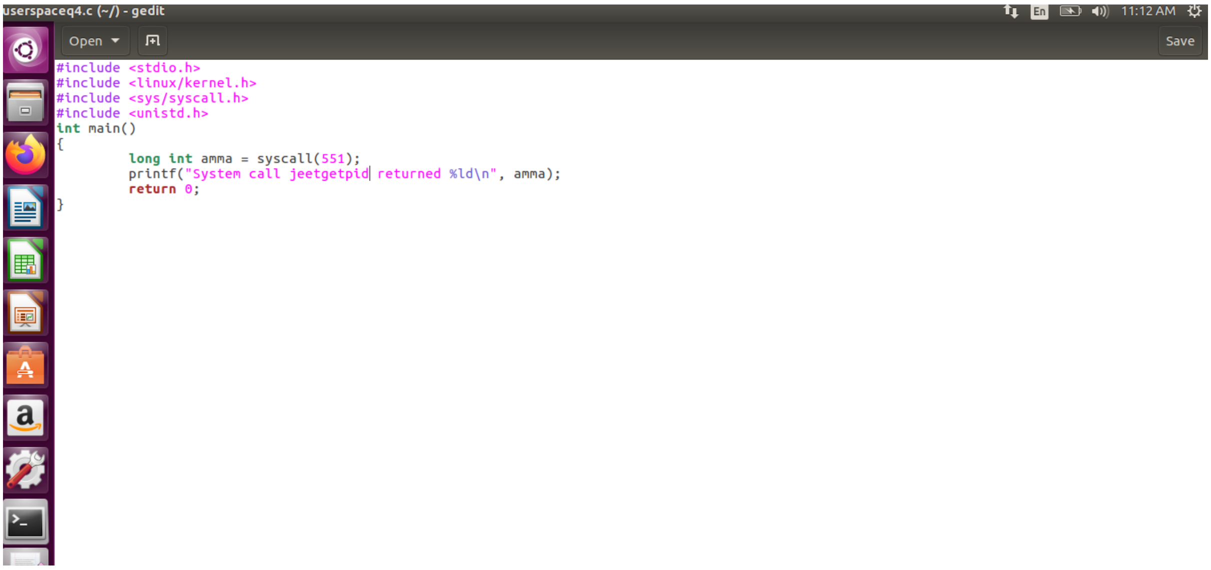
Q4 All the steps of Q1 are the same, just that step - 7 is to be skipped here.

STEP-6: Make entry inside syscall_64.tbl


syscall_64.tbl [Read-Only] (/usr/src/linux-4.19.210/arch/x86/entry/syscalls) - gedit
En
11:16 AM
Open
Save

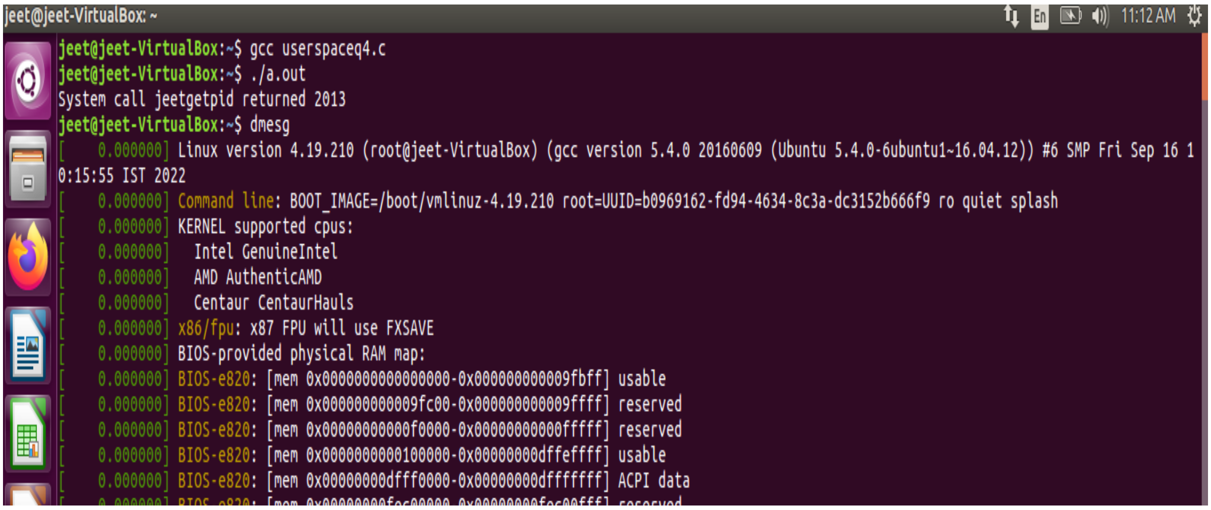
```
518     x32     sendmsg                 __x32_compat_sys_sendmsg
519     x32     recvmsg                 __x32_compat_sys_recvmsg
520     x32     execve                  __x32_compat_sys_execve/ptregs
521     x32     ptrace                  __x32_compat_sys_ptrace
522     x32     rt_sigpending           __x32_compat_sys_rt_sigpending
523     x32     rt_sigtimedwait         __x32_compat_sys_rt_sigtimedwait
524     x32     rt_sigqueueinfo         __x32_compat_sys_rt_sigqueueinfo
525     x32     sigaltstack             __x32_compat_sys_sigaltstack
526     x32     timer_create            __x32_compat_sys_timer_create
527     x32     mq_notify               __x32_compat_sys_mq_notify
528     x32     kexec_load              __x32_compat_sys_kexec_load
529     x32     waitid                  __x32_compat_sys_waitid
530     x32     set_robust_list         __x32_compat_sys_set_robust_list
531     x32     get_robust_list         __x32_compat_sys_get_robust_list
532     x32     vmsplice                __x32_compat_sys_vmsplice
533     x32     move_pages              __x32_compat_sys_move_pages
534     x32     preadv                  __x32_compat_sys_preadv64
535     x32     pwritev                 __x32_compat_sys_pwritev64
536     x32     rt_tgsigqueueinfo       __x32_compat_sys_rt_tgsigqueueinfo
537     x32     recvmmsg                __x32_compat_sys_recvmmsg
538     x32     sendmmsg                __x32_compat_sys_sendmmsg
539     x32     process_vm_readv        __x32_compat_sys_process_vm_readv
540     x32     process_vm_writev       __x32_compat_sys_process_vm_writev
541     x32     setsockopt              __x32_compat_sys_setsockopt
542     x32     getsockopt              __x32_compat_sys_getsockopt
543     x32     io_setup                __x32_compat_sys_io_setup
544     x32     io_submit               __x32_compat_sys_io_submit
545     x32     execveat                __x32_compat_sys_execveat/ptregs
546     x32     preadv2                 __x32_compat_sys_preadv64v2
547     x32     pwritev2                __x32_compat_sys_pwritev64v2
548     64      hello                   jeethello
549     64      q2                      __x64_sys_jeetprint
550     64      q3                      __x64_sys_jeetprocess
551     64      q4                      __x64_sys_jeetgetpid
```

STEP-12: gedit userspaceq4.c



```c
#include <stdio.h>
#include <linux/kernel.h>
#include <sys/syscall.h>
#include <unistd.h>
int main()
{
        long int amma = syscall(551);
        printf("System call jeetgetpid returned %ld\n", amma);
        return 0;
}
```

STEP-13: gcc userspaceq4.c And then ./a.out



```
jeet@jeet-VirtualBox:~$ gcc userspaceq4.c
jeet@jeet-VirtualBox:~$ ./a.out
System call jeetgetpid returned 2013
jeet@jeet-VirtualBox:~$ dmesg
[    0.000000] Linux version 4.19.210 (root@jeet-VirtualBox) (gcc version 5.4.0 20160609 (Ubuntu 5.4.0-6ubuntu1~16.04.12)) #6 SMP Fri Sep 16 1
0:15:55 IST 2022
[    0.000000] Command line: BOOT_IMAGE=/boot/vmlinuz-4.19.210 root=UUID=b0969162-fd94-4634-8c3a-dc3152b666f9 ro quiet splash
[    0.000000] KERNEL supported cpus:
[    0.000000]   Intel GenuineIntel
[    0.000000]   AMD AuthenticAMD
[    0.000000]   Centaur CentaurHauls
[    0.000000] x86/fpu: x87 FPU will use FXSAVE
[    0.000000] BIOS-provided physical RAM map:
[    0.000000] BIOS-e820: [mem 0x0000000000000000-0x000000000009fbff] usable
[    0.000000] BIOS-e820: [mem 0x000000000009fc00-0x000000000009ffff] reserved
[    0.000000] BIOS-e820: [mem 0x00000000000f0000-0x00000000000fffff] reserved
[    0.000000] BIOS-e820: [mem 0x0000000000100000-0x00000000dffeffff] usable
[    0.000000] BIOS-e820: [mem 0x00000000dfff0000-0x00000000dfffffff] ACPI data
[    0.000000] BIOS-e820: [mem 0x00000000fec00000-0x00000000fec00fff] reserved
```