



UNIVERSIDADE DE SÃO PAULO - USP
BACHARELADO EM SISTEMAS DE INFORMAÇÃO
Sistemas Operacionais - Lista prática 1

Confecção de uma chamada de sistema

GIOVANNA PEDRINO BELASCO - 12543287
LAURA FERNANDES CAMARGOS - 13692334

São Carlos - SP
18/10/23

Primeiramente, para a implementação da chamada ao sistema, é necessário preparar o ambiente e garantir que temos todos os pacotes necessários. Para isso, foram executados os seguintes comandos:

Garantir que o sistema operacional está completamente atualizado.

```
sudo apt update && sudo apt upgrade -y
```

Instalação dos pacotes essenciais para execução dos kernels.

```
sudo apt install build-essential libncurses-dev libssl-dev  
libelf-dev bison flex -y
```

Remoção de pacotes já existentes:

```
sudo apt clean && sudo apt autoremove -y
```

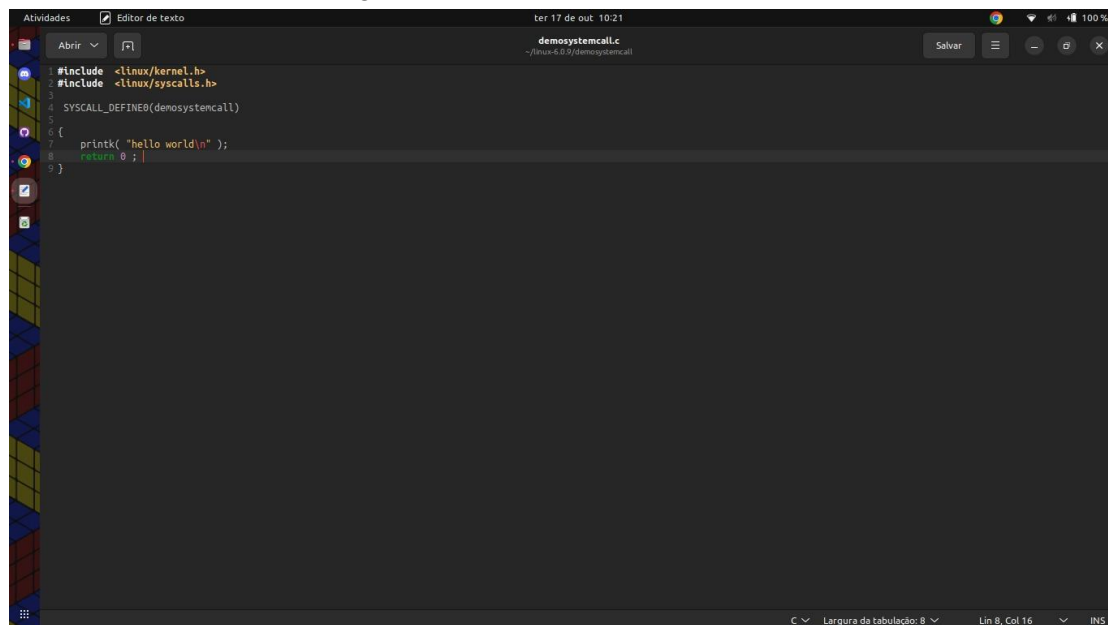
Instalação da versão mais atual de Kernel do Linux. (essa versão era a mais atual na época do tutorial que serviu como orientação).

```
wget -P ~/
```

<https://cdn.kernel.org/pub/linux/kernel/v6.x/linux-6.0.9.tar.gz>

Após a instalação, foi feito o *unpack* e reboot do computador.

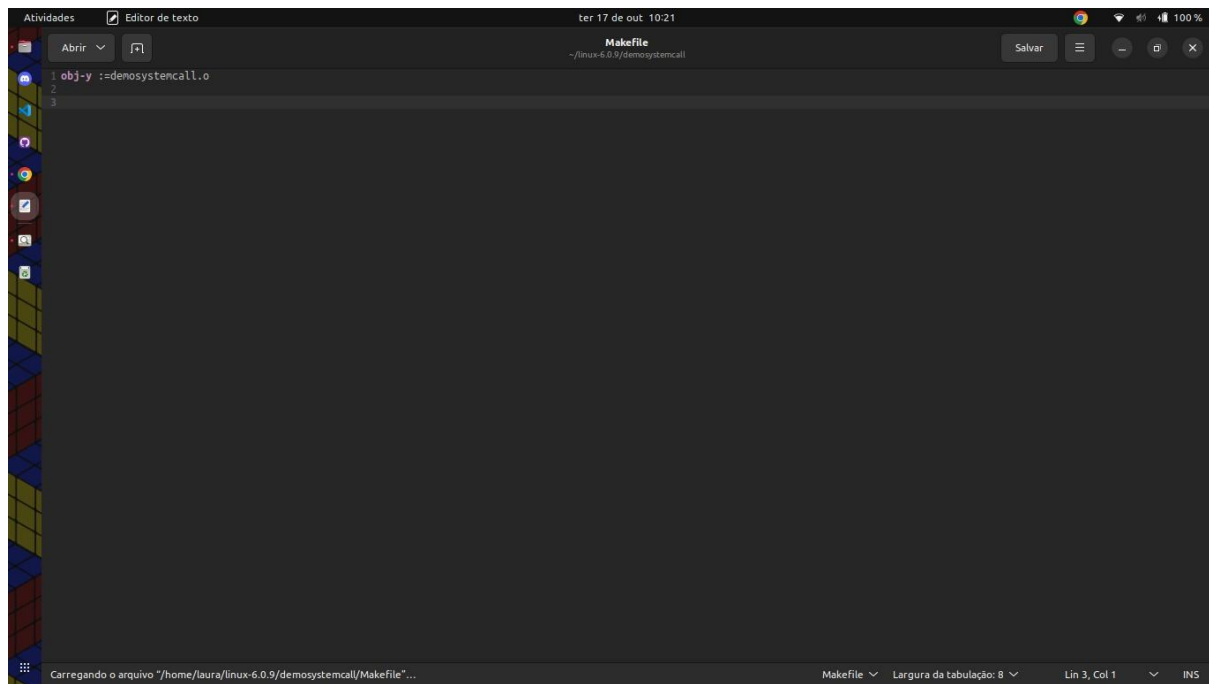
Nessa segunda etapa, começamos criando um diretório e um código em C na pasta do kernel do Linux. Abaixo, segue print do arquivo em C criado:

A screenshot of a Linux desktop environment showing a terminal window. The terminal title bar reads "Atividades Editor de texto" and "ter 17 de out 10:21". The terminal content shows the following C code:

```
#include <linux/kernel.h>
#include <linux/syscalls.h>
SYSCALL_DEFINE(demosystemcall)
{
    printk( "hello world\n" );
    return 0 ;
}
```

The terminal window has a dark background with light-colored text. The code is written in a standard C syntax. The terminal also shows the file path "~/linux-6.0.9/demosystemcall" and a "Salvar" button in the top right corner.

No mesmo diretório, também foi criado um Makefile, como é possível observar no print abaixo.



Explicação do Makefile: essa linha informa ao sistema de compilação do kernel que o arquivo "demosystemcall.o" faz parte do módulo do kernel do Linux e deve ser incluído na compilação. Portanto, ao compilar o módulo, o utilitário make se encarrega de compilar "demosystemcall.c" e criar "demosystemcall.o". O resultado disso tudo é um arquivo de módulo que será possível carregar no kernel para adicionar a syscall.

Após isso, abrimos o Makefile em algum editor de texto (*como exemplo, gedit Makefile*) e alteramos a linha 1104 (no nosso caso):

```
1104 kernel/ certs/ mm/ fs/ ipc/ security/ crypto/ block/
```

Para:

```
1104 kernel/ certs/ mm/ fs/ ipc/ security/ crypto/ block/ identity/
```

Mudamos o diretório:

```
include/linux/syscalls.h
```

Acessamos o arquivo header de chamadas ao sistema e adicionamos um header correspondente a nossa função:

```
asmlinkage long sys_demosystemcall(void);
```

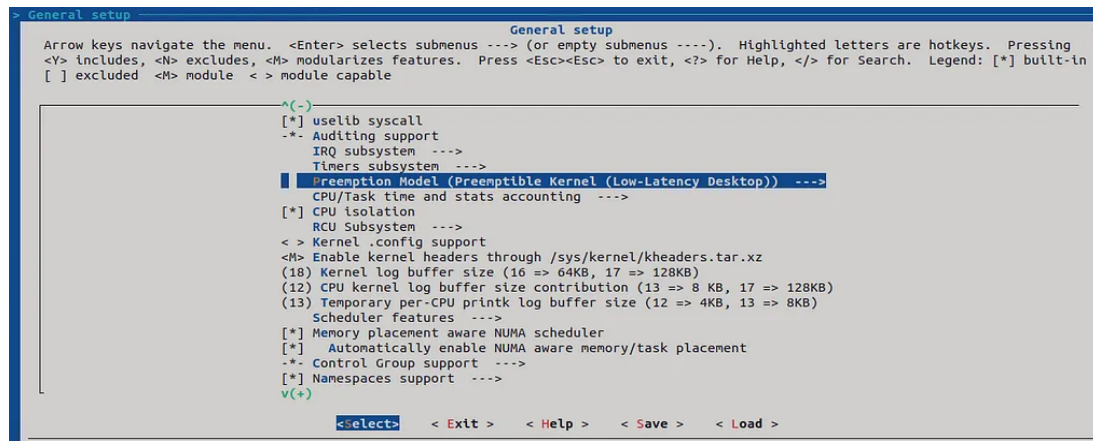
Também adicionamos a nova system call na tabela do kernel:

```
gedit arch/x86/entry/syscalls/syscall_64.tbl - abre a tabela
```

```
335 common demosystemcall sys_demosystemcall - linha adicionada
```

Para a configuração do Kernel, mudamos a configuração da imagem abaixo:

`make menuconfig`



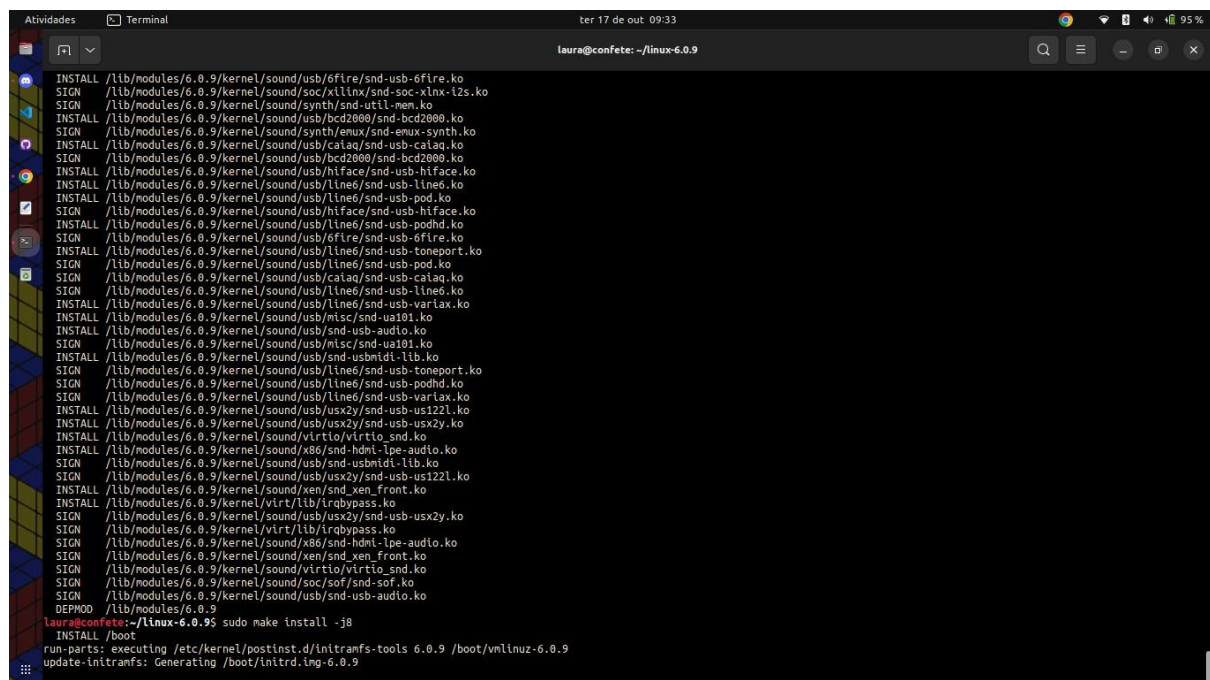
Nas imagens abaixo, segue a compilação do novo kernel com os comandos:

`make -j8` (no nosso caso o computador que utilizado tem 8 processadores, por isso o j8)

`sudo make modules_install -j8`

`sudo make install -j8`

`sudo update-grub`



```
Atividades Terminal ter 17 de out 09:40
Laura@confete: ~/linux-6.0.9

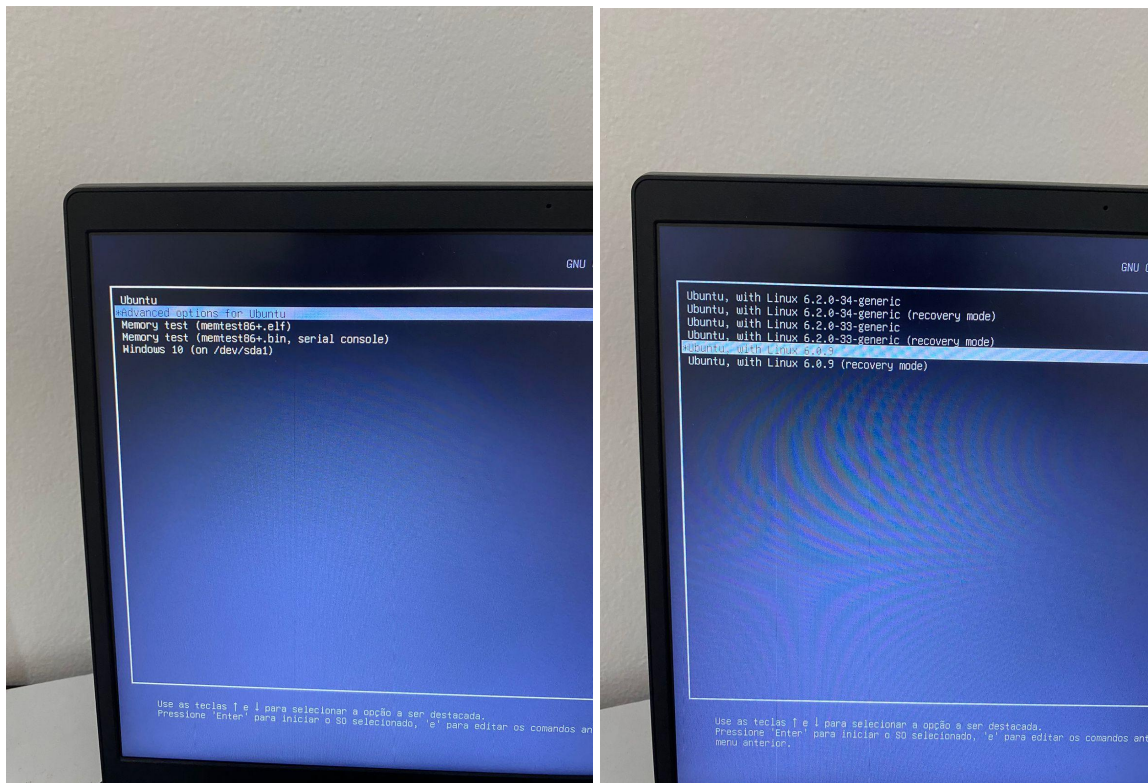
SIGN /lib/modules/6.0.9/kernel/sound/soc/sof/snd-sof.ko
SIGN /lib/modules/6.0.9/kernel/sound/usb/snd-usb-audio.ko
DEPMOD /lib/modules/6.0.9
Laura@confete:~/linux-6.0.9$ sudo make install -j8
INSTALL /boot
run-parts: executing /etc/kernel/postinst.d/initramfs-tools 6.0.9 /boot/vmlinuz-6.0.9
update-initramfs: Generating /boot/initrd.img-6.0.9
run-parts: executing /etc/kernel/postinst.d/unattended-upgrades 6.0.9 /boot/vmlinuz-6.0.9
run-parts: executing /etc/kernel/postinst.d/update-notifier 6.0.9 /boot/vmlinuz-6.0.9
run-parts: executing /etc/kernel/postinst.d/xx-update-initrd-links 6.0.9 /boot/vmlinuz-6.0.9
I: /boot/initrd.img.old is now a symlink to initrd.img-6.2.0-34-generic
I: /boot/initrd.img is now a symlink to initrd.img-6.0.9
run-parts: executing /etc/kernel/postinst.d/zz-shim 6.0.9 /boot/vmlinuz-6.0.9
run-parts: executing /etc/kernel/postinst.d/zz-update-grub 6.0.9 /boot/vmlinuz-6.0.9
Sourcing file /etc/default/grub
Sourcing file /etc/default/grub.d/init-select.cfg
Generating grub configuration file ...
Found linux image: /boot/vmlinuz-6.2.0-34-generic
Found initrd image: /boot/initrd.img-6.2.0-34-generic
Found linux image: /boot/vmlinuz-6.2.0-33-generic
Found initrd image: /boot/initrd.img-6.2.0-33-generic
Found linux image: /boot/vmlinuz-6.0.9
Found initrd image: /boot/initrd.img-6.0.9
Found memtest86+ image: /boot/memtest86+.elf
Found memtest86+ image: /boot/memtest86+.bin
Warning: os-prober will be executed to detect other bootable partitions.
Its output will be used to detect bootable binaries on them and create new boot entries.
Found Windows 10 on /dev/sda1
done
Laura@confete:~/linux-6.0.9$ sudo update-grub
Sourcing file /etc/default/grub
Sourcing file /etc/default/grub.d/init-select.cfg
Generating grub configuration file ...
Found linux image: /boot/vmlinuz-6.2.0-34-generic
Found initrd image: /boot/initrd.img-6.2.0-34-generic
Found linux image: /boot/vmlinuz-6.2.0-33-generic
Found initrd image: /boot/initrd.img-6.2.0-33-generic
Found linux image: /boot/vmlinuz-6.0.9
Found initrd image: /boot/initrd.img-6.0.9
Found memtest86+ image: /boot/memtest86+.elf
Found memtest86+ image: /boot/memtest86+.bin
Warning: os-prober will be executed to detect other bootable partitions.
Its output will be used to detect bootable binaries on them and create new boot entries.
Found Windows 10 on /dev/sda1
done
Laura@confete:~/linux-6.0.9$
```

Após atualizar o Grub, demos um reboot para reiniciar o computador.

```
Atividades Terminal ter 17 de out 09:40
Laura@confete: ~/linux-6.0.9

update-initramfs: Generating /boot/initrd.img-6.0.9
run-parts: executing /etc/kernel/postinst.d/unattended-upgrades 6.0.9 /boot/vmlinuz-6.0.9
run-parts: executing /etc/kernel/postinst.d/update-notifier 6.0.9 /boot/vmlinuz-6.0.9
run-parts: executing /etc/kernel/postinst.d/xx-update-initrd-links 6.0.9 /boot/vmlinuz-6.0.9
I: /boot/initrd.img.old is now a symlink to initrd.img-6.2.0-34-generic
I: /boot/initrd.img is now a symlink to initrd.img-6.0.9
run-parts: executing /etc/kernel/postinst.d/zz-shim 6.0.9 /boot/vmlinuz-6.0.9
run-parts: executing /etc/kernel/postinst.d/zz-update-grub 6.0.9 /boot/vmlinuz-6.0.9
Sourcing file /etc/default/grub
Sourcing file /etc/default/grub.d/init-select.cfg
Generating grub configuration file ...
Found linux image: /boot/vmlinuz-6.2.0-34-generic
Found initrd image: /boot/initrd.img-6.2.0-34-generic
Found linux image: /boot/vmlinuz-6.2.0-33-generic
Found initrd image: /boot/initrd.img-6.2.0-33-generic
Found linux image: /boot/vmlinuz-6.0.9
Found initrd image: /boot/initrd.img-6.0.9
Found memtest86+ image: /boot/memtest86+.elf
Found memtest86+ image: /boot/memtest86+.bin
Warning: os-prober will be executed to detect other bootable partitions.
Its output will be used to detect bootable binaries on them and create new boot entries.
Found Windows 10 on /dev/sda1
done
Laura@confete:~/linux-6.0.9$ sudo update-grub
Sourcing file /etc/default/grub
Sourcing file /etc/default/grub.d/init-select.cfg
Generating grub configuration file ...
Found linux image: /boot/vmlinuz-6.2.0-34-generic
Found initrd image: /boot/initrd.img-6.2.0-34-generic
Found linux image: /boot/vmlinuz-6.2.0-33-generic
Found initrd image: /boot/initrd.img-6.2.0-33-generic
Found linux image: /boot/vmlinuz-6.0.9
Found initrd image: /boot/initrd.img-6.0.9
Found memtest86+ image: /boot/memtest86+.elf
Found memtest86+ image: /boot/memtest86+.bin
Warning: os-prober will be executed to detect other bootable partitions.
Its output will be used to detect bootable binaries on them and create new boot entries.
Found Windows 10 on /dev/sda1
done
Laura@confete:~/linux-6.0.9$ Reboot
Comando 'Reboot' não encontrado, você quis dizer:
  comando 'reboot' do deb systemd-sysv (249.11-0ubuntu3.10)
  comando 'reboot' do deb finit-sysv (4.2-1)
  comando 'reboot' do deb molly-guard (0.7.2)
Experimente: sudo apt install <deb name>
Laura@confete:~/linux-6.0.9$ reboot
```

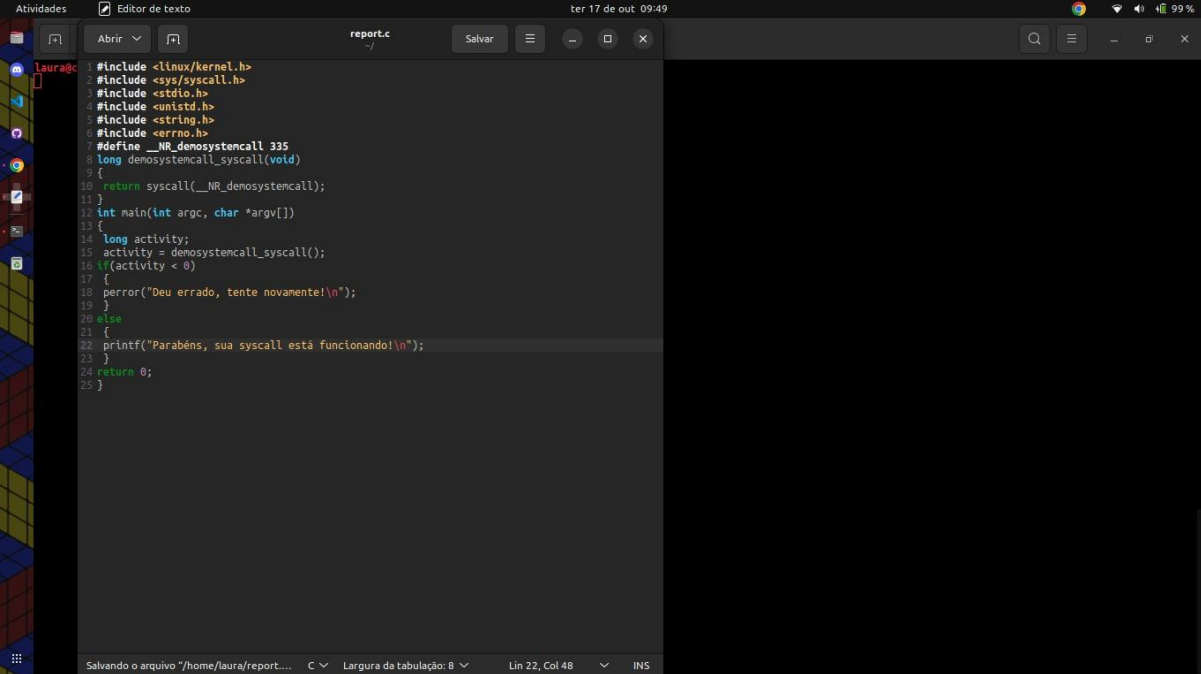

Quando o computador ligou novamente, foi preciso bootar a versão correta do Linux, indo em *opções avançadas* e *selecionando a versão do kernel 6.0.9* que compilamos.



Na versão 6.0.9 do kernel e com o SO inicializado, executamos o comando `ls -lrt`, listando os arquivos e diretórios no diretório atual em formato de lista, na ordem inversa, ou seja, do mais recente para o mais antigo.

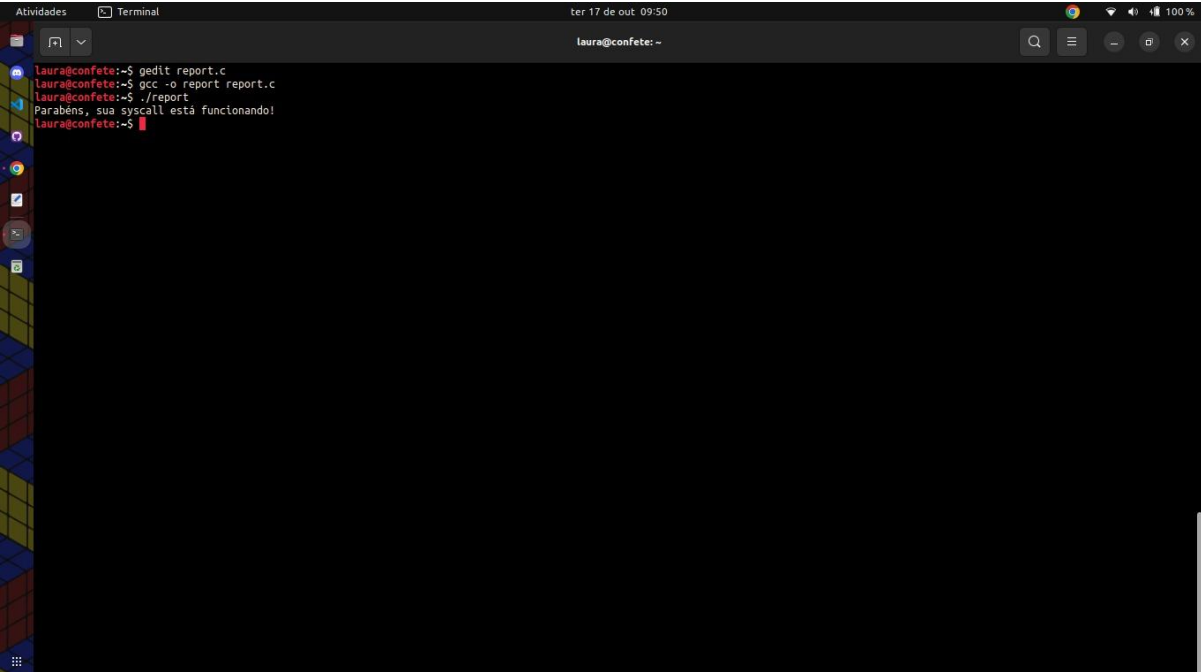
```
Atividades Terminal ter 17 de out: 09:47
laura@confete: /boot
laura@confete:/boot$ ls -lrt
total 1122340
-rw-r--r-- 1 root root 104980 fev 6 2022 memtest86+_multiboot.bin
-rw-r--r-- 1 root root 104476 fev 6 2022 memtest86+_elf
-rw-r--r-- 1 root root 102800 fev 6 2022 memtest86+_bin
-rw-r--r-- 1 root root 7969198 set 7 04:11 System.map-6.2.0-33-generic
-rw-r--r-- 1 root root 275587 set 7 04:11 config-6.2.0-33-generic
-rw-r--r-- 1 root root 13788872 set 7 05:18 vmlinuz-6.2.0-33-generic
-rw-r--r-- 1 root root 7971367 set 7 09:17 System.map-6.2.0-34-generic
-rw-r--r-- 1 root root 275605 set 7 09:17 config-6.2.0-34-generic
-rw-r--r-- 1 root root 13795400 set 7 09:18 vmlinuz-6.2.0-34-generic
lrwxrwxrwx 1 root root 24 out 5 15:12 vmlinuz.old -> vmlinuz-6.2.0-34-generic
-rw-r--r-- 1 root root 137380167 out 13 09:32 initrd.img-6.2.0-34-generic
-rw-r--r-- 1 root root 137280985 out 13 09:33 initrd.img-6.2.0-33-generic
-rw-r--r-- 1 root root 10562624 out 17 09:33 vmlinuz-6.0.9
lrwxrwxrwx 1 root root 13 out 17 09:33 vmlinuz -> vmlinuz-6.0.9
-rw-r--r-- 1 root root 6257794 out 17 09:33 System.map-6.0.9
-rw-r--r-- 1 root root 270451 out 17 09:33 config-6.0.9
-rw-r--r-- 1 root root 812830928 out 17 09:38 initrd.img-6.0.9
lrwxrwxrwx 1 root root 27 out 17 09:38 initrd.img.old -> initrd.img-6.2.0-34-generic
lrwxrwxrwx 1 root root 16 out 17 09:38 initrd.img -> initrd.img-6.0.9
drwxr-xr-x 5 root root 4096 out 17 09:40 grub
laura@confete:/boot$
```

Para testar nossa syscall e sua execução de fato, criamos um arquivo com *gedit* *report.c* e colocamos um código que chama a syscall personalizada com o número 335 (definido anteriormente na tabela das syscalls) e imprime uma mensagem de sucesso ou erro com base no resultado da syscall.



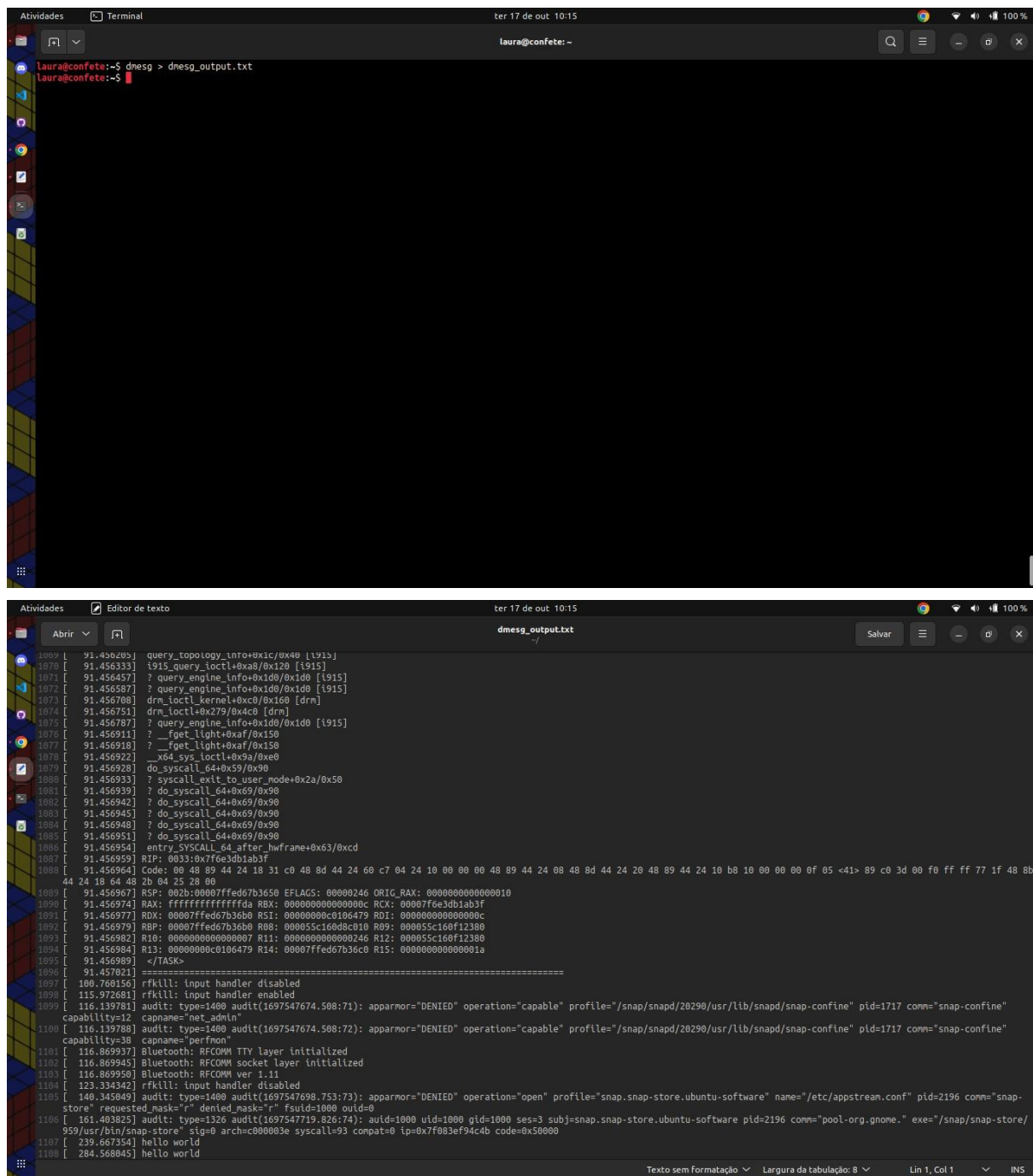
```
1 #include <linux/kernel.h>
2 #include <sys/syscall.h>
3 #include <stdio.h>
4 #include <unistd.h>
5 #include <string.h>
6 #include <errno.h>
7 #define __NR_demosystemcall 335
8 long demosystemcall_syscall(void)
9 {
10     return syscall(__NR_demosystemcall);
11 }
12 int main(int argc, char *argv[])
13 {
14     long activity;
15     activity = demosystemcall_syscall();
16     if (activity < 0)
17     {
18         perror("Deu errado, tente novamente!\n");
19     }
20     else
21     {
22         printf("Parabéns, sua syscall está funcionando!\n");
23     }
24     return 0;
25 }
```

Compilando e executando ... SUCESSO!



```
laura@confete:~$ gedit report.c
laura@confete:~$ gcc -o report report.c
laura@confete:~$ ./report
Parabéns, sua syscall está funcionando!
laura@confete:~$
```

Executamos também o comando *dmesg*, que mostra uma lista de mensagens do kernel, geralmente começando com as mensagens mais recentes no topo, a fim de verificar se a syscall implementada estava funcionando conforme o esperado, ou seja, printando “hello world”:



The first screenshot shows a terminal window with the command `dmesg > dmesg_output.txt` being executed. The second screenshot shows the contents of the file `dmesg_output.txt` in a text editor. The file contains a list of kernel messages, including system calls, hardware events, and a "hello world" message.

```
1069 [ 91.456205] query_topology_info+0x1c/0x48 [1915]
1070 [ 91.456333] i915_query_ioctl+0xa8/0x120 [1915]
1071 [ 91.456457] ? query_engine_info+0x1d0/0x1d0 [1915]
1072 [ 91.456587] ? query_engine_info+0x1d0/0x1d0 [1915]
1073 [ 91.456708] drm_ioctl_kernel+0xc0/0x160 [drm]
1074 [ 91.456751] drm_ioctl+0x279/0x4c0 [drm]
1075 [ 91.456787] ? query_engine_info+0x1d0/0x1d0 [1915]
1076 [ 91.456911] ? __fget_light+0xaf/0x150
1077 [ 91.456918] ? __fget_light+0xaf/0x150
1078 [ 91.456922] ? x64_sys_ioctl+0x9a/0xe0
1079 [ 91.456928] do_syscall_64+0x59/0x90
1080 [ 91.456933] ? syscall_exit_to_user_mode+0x2a/0x50
1081 [ 91.456939] ? do_syscall_64+0x69/0x90
1082 [ 91.456942] ? do_syscall_64+0x69/0x90
1083 [ 91.456945] ? do_syscall_64+0x69/0x90
1084 [ 91.456948] ? do_syscall_64+0x69/0x90
1085 [ 91.456951] ? do_syscall_64+0x69/0x90
1086 [ 91.456954] entry_SYSCALL_64_after_hwframe+0x63/0xcd
1087 [ 91.456959] RIP: 0033:0x7f63db1ab3f
1088 [ 91.456964] Code: 00 48 b9 44 24 18 31 c0 48 8d 44 24 60 c7 04 24 10 00 00 48 89 44 24 08 8d 44 24 20 89 44 24 10 b8 10 00 00 0f 05 <41> 89 c0 3d 00 f0 ff 77 1f 48 8b
44 24 18 64 48 2b 04 25 28 00
1089 [ 91.456967] RSP: 002b:00007ffed67b3650 EFLAGS: 00000246 ORIG_RAX: 0000000000000010
1090 [ 91.456974] RAX: ffffffff7ffed67b3650 RBX: 000000000000000c RCX: 00007f63db1ab3f
1091 [ 91.456977] RDX: 00007ffed67b3650 RSI: 0000000000000047 RDI: 000000000000000c
1092 [ 91.456979] RBP: 00007ffed67b3650 R08: 000055c160d8c018 R09: 000055c160f12380
1093 [ 91.456982] R10: 0000000000000007 R11: 0000000000000246 R12: 000055c160f12380
1094 [ 91.456984] R13: 00000000c0106479 R14: 00007ffed67b36c0 R15: 000000000000001a
1095 [ 91.456989] </TASK>
1096 [ 91.457021] =====
1097 [ 100.760156] rkill: input handler disabled
1098 [ 115.972681] rkill: input handler enabled
1099 [ 116.139781] audit: type=1400 audit(1697547674.508:71): apparmor="DENIED" operation="capable" profile="/snap/snapd/20290/usr/lib/snapd/snap-confine" pid=1717 comm="snap-confine"
capability=12 capname="net_admin"
1100 [ 116.139788] audit: type=1400 audit(1697547674.508:72): apparmor="DENIED" operation="capable" profile="/snap/snapd/20290/usr/lib/snapd/snap-confine" pid=1717 comm="snap-confine"
capability=38 capname="perfmon"
1101 [ 116.869937] Bluetooth: RFCOMM TTY layer initialized
1102 [ 116.869945] Bluetooth: RFCOMM socket layer initialized
1103 [ 116.869950] Bluetooth: RFCOMM ver 1.11
1104 [ 123.334342] rkill: input handler disabled
1105 [ 140.345049] audit: type=1400 audit(1697547698.753:73): apparmor="DENIED" operation='open' profile='snap.snap-store.ubuntu-software' name='/etc/appstream.conf' pid=2196 comm='snap-
store' requested_mask='r' denied_mask='r' fsuid=1000 ouid=0
1106 [ 161.408325] audit: type=1326 audit(1697547719.826:74): audit=1000 uid=1000 gid=1000 ses=3 subj=snap.snap-store.ubuntu-software pid=2196 comm="pool-org.gnome." exe="/snap/snap-store/
959/usr/bin/snap-store" sig=0 arch=c000003e syscall=93 compat=0 ip=0x7f083ef94c4b code=0x50000
1107 [ 239.667354] hello world
1108 [ 284.568045] hello world
```

Tutoriais que serviram como guia:

- <https://medium.com/@rajiv.cse/add-a-system-call-to-the-linux-kernel-6-0-9-in-ubuntu-22-04-acd7f7afc933>
- <https://dev.to/jasper/adding-a-system-call-to-the-linux-kernel-5-8-1-in-ubuntu-20-04-lts-2ga8>