МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ
И МАССОВЫХ КОММУНИКАЦИЙ
Ордена Трудового Красного Знамени
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский технический университет связи и информатики»

Кафедра "Системное программирование"

Отчет по практической работе №2

на тему:

«Ядро Linux»

Выполнила:
студентка группы БВТ2102
Никифорова Олеся Ильинична

Проверила: Королькова Т. В.

Москва 2023

Цель работы

Изучить архитектуру и основные компоненты ядра Linux, включая механизм системных вызовов. Изучить возможности утилиты **strace** для отладки и анализа работы программ на уровне системных вызовов. Приобрести практический навык создания и загрузки модуля в ядро Linux.

Выполнение

Задание 1. Анализ системных вызовов с помощью утилиты strace

1. Убедитесь, что **strace** установлена, запустив ее с параметром -V: ***strace -V***. Если утилита отсутствует, установите ее: ***sudo apt install strace***.

```
nikiforova@debian:~$ strace -V
strace -- version 6.1
Copyright (c) 1991-2022 The strace developers <https://strace.io>.
This is free software; see the source for copying conditions.  There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

Optional features enabled: stack-trace=libunwind stack-demangle m32-mpers mx32
-mpers
```

2. Ознакомьтесь со справкой об использовании утилиты **strace**: ***man strace***.

```
STRACE(1)                  General Commands Manual                  STRACE(1)

NAME
       strace - trace system calls and signals

SYNOPSIS
       strace [-ACdffhikqqrtttTvVwxxyyzZ] [-I n] [-b execve] [-e expr]...
              [-O overhead] [-S sortby] [-U columns] [-a column] [-o file]
              [-s strsize] [-X format] [-P path]... [-p pid]...
              [--seccomp-bpf] { -p pid | [-DDD] [-E var[=val]]...
              [-u username] command [args] }

       strace -c [-dfwzZ] [-I n] [-b execve] [-e expr]... [-O overhead]
              [-S sortby] [-U columns] [-P path]... [-p pid]...
              [--seccomp-bpf] { -p pid | [-DDD] [-E var[=val]]...
              [-u username] command [args] }

DESCRIPTION
       In the simplest case strace runs the specified command until  it  ex-
       its.   It intercepts and records the system calls which are called by
       a process and the signals which are received by a process.  The  name
 Manual page strace(1) line 1 (press h for help or q to quit)
```

3. Запустите strace для команды из таблицы 1. Обратите внимание, что в некоторых случаях потребуется задать аргументы. На основании полученных результатов заполните таблицу 2 для 5-7 различных системных вызовов.

```
nikiforova@debian:~$ strace ls
execve("/usr/bin/ls", ["ls"], 0x7fff07a4a240 /* 39 vars */) = 0
brk(NULL)                               = 0x562fae9ba000
mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7
fc2fe492000
access("/etc/ld.so.preload", R_OK)      = -1 ENOENT (Нет такого файла или ката
лога)
openat(AT_FDCWD, "/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3
newfstatat(3, "", {st_mode=S_IFREG|0644, st_size=59914, ...}, AT_EMPTY_PATH) =
 0
mmap(NULL, 59914, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7fc2fe483000
close(3)                                = 0
openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libselinux.so.1", O_RDONLY|O_CLOEXEC)
= 3
read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\0\0\0\0\0\0\0\0"..., 8
32) = 832
newfstatat(3, "", {st_mode=S_IFREG|0644, st_size=174312, ...}, AT_EMPTY_PATH)
= 0
mmap(NULL, 186064, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7fc2fe45500
0
newfstatat(4, "", {st_mode=S_IFREG|0644, st_size=27028, ...}, AT_EMPTY_PATH) =
 0
mmap(NULL, 27028, PROT_READ, MAP_SHARED, 4, 0) = 0x7fc2fe48b000
close(4)                                = 0
futex(0x7fc2fe447a4c, FUTEX_WAKE_PRIVATE, 2147483647) = 0
getdents64(3, 0x562fae9c0ce0 /* 0 entries */, 32768) = 0
close(3)                                = 0
newfstatat(1, "", {st_mode=S_IFCHR|0620, st_rdev=makedev(0x88, 0), ...}, AT_EM
PTY_PATH) = 0
write(1, " \320\222\320\270\320\264\320\265\320\276\t     \320\227\320\260\320
\263\321\200\321\203\320\267\320\272\320"..., 78 Видео       Загрузки      Муз
ыка       'Рабочий стол'
) = 78
write(1, " \320\224\320\276\320\272\321\203\320\274\320\265\320\275\321\202\32
1\213   \320\230\320\267\320\276\320\261\321\200"..., 91 Документы   Изображен
ия   Общедоступные   Шаблоны
) = 91
close(1)                                = 0
close(2)                                = 0
exit_group(0)                           = ?
+++ exited with 0 +++
nikiforova@debian:~$
```

Таблица 2. Результаты анализа

№	Системный вызов	Описание вызова	Входные параметры	Время выполнения, мкс	Возвращаемое значение
1	getdents64()	Читает несколько структур linuxdiret из каталога, на который указывает открытый файловый дескриптор fd, в буфер, указанный в dirp. В аргументе count задаётся размер этого буфера.	ssizet getdents64(int fd, void dirp[.count], sizet count);	39	Записи каталога
2	write()	Записывает количество байт из буфера, начиная с bufк файлу, на который ссылается файловый дескриптор fd. Для искомого файла запись происходит в файле смещение, и смещение файла увеличивается на количество байт на самом деле написано.	ssizet write(int fd, const void buf[.count], sizet count);	24	В случае успеха возвращается количество записанных байтов. При ошибке, возвращается -1, и errno настроен на указание на ошибку.

3	openat()	Если путь, указанный в пути, является относительным, то он интерпретируется относительно каталога, на который ссылается дескриптор файла dirfd.	int openat(int dirfd, const char *pathname, int flags);
			
			int openat(int dirfd, const char *pathname, int flags, modet mode);
				16	В случае успеха возвращает новый файл дескриптор (неотрицательное целое число). При ошибке возвращается -1 и errno настроен на указание на ошибку.

4	close()	Закрывает дескриптор файла, так что он	int close(int	13	Возвращает ноль в случае

		больше не ссылается на любой файл и может быть повторно использован.	fd);		успеха. При ошибке возвращается -1, и errno настроен на указание на ошибку.
5	ioctl()	Манипулирует базовым устройством параметры специальных файлов. В частности, многие действующие характеристики специальных файлов символов (например, терминалов) могут контролироваться с помощью запросов ioctl().	int ioctl(int fd, unsigned long request, ...);	6	при успехе возвращается ноль. Несколько запросов ioctl() используют возвращаемое значение в качестве выходного параметра и возвращаемое неотрицательное ценность успеха. При ошибке возвращается -1, а errno установлен на укажите ошибку.

4. Перенаправьте вывод **strace** в файл log в вашей домашней директории.

```
nikiforova@debian:~$ strace -r -o out2.log ls
 out2.log   Документы   Изображения   Общедоступные   Шаблоны
 Видео      Загрузки    Музыка        'Рабочий стол'
nikiforova@debian:~$ cat out2.log
     0.000000 execve("/usr/bin/ls", ["ls"], 0x7fff2971f5f8 /* 39 vars */) = 0
     0.000606 brk(NULL)                      = 0x55c2c7b50000
     0.001127 mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS
, -1, 0) = 0x7f2c71287000
     0.000114 access("/etc/ld.so.preload", R_OK) = -1 ENOENT (Нет такого файла
 или каталога)
     0.000344 openat(AT_FDCWD, "/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3
     0.000041 newfstatat(3, "", {st_mode=S_IFREG|0644, st_size=59914, ...}, AT
_EMPTY_PATH) = 0
     0.000078 mmap(NULL, 59914, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7f2c71278000
     0.000030 close(3)                       = 0
     0.000095 openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libselinux.so.1", O_RDON
LY|O_CLOEXEC) = 3
     0.000052 read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\0\0\0\0\
0\0\0\0"..., 832) = 832
     0.000108 newfstatat(3, "", {st_mode=S_IFREG|0644, st_size=174312, ...}, A
T_EMPTY_PATH) = 0
```

5. Получите статистику выполненных системных вызовов.

```
nikiforova@debian:~$ strace -c ls
 Видео       Загрузки      Музыка         'Рабочий стол'
 Документы   Изображения   Общедоступные   Шаблоны
% time     seconds  usecs/call     calls    errors syscall
------ ----------- ----------- --------- --------- ----------------
 30,00    0,000039          19         2           getdents64
 18,46    0,000024          12         2           write
 12,31    0,000016           2         8           openat
 10,77    0,000014           0        19           mmap
 10,00    0,000013           1        10           close
  8,46    0,000011           1         9           newfstatat
  4,62    0,000006           3         2           ioctl
  3,85    0,000005           2         2         2 access
  1,54    0,000002           2         1           futex
  0,00    0,000000           0         5           read
  0,00    0,000000           0         5           mprotect
  0,00    0,000000           0         1           munmap
  0,00    0,000000           0         3           brk
  0,00    0,000000           0         2           pread64
  0,00    0,000000           0         1           execve
  0,00    0,000000           0         2         2 statfs
  0,00    0,000000           0         1           arch_prctl
  0,00    0,000000           0         1           set_tid_address
  0,00    0,000000           0         1           set_robust_list
  0,00    0,000000           0         1           prlimit64
  0,00    0,000000           0         1           getrandom
  0,00    0,000000           0         1           rseq
------ ----------- ----------- --------- --------- ----------------
100,00    0,000130           1        80         4 total
nikiforova@debian:~$
```

6. Выполните трассировку системных вызовов для произвольного работающего процесса, подключившись к нему по PID.

Запускаем страницу с man документацией, смотрим pid через ps -aux из другого терминала, подключаемся к процессу и завершаем

```
nikiforova@debian:~$ strace -p 2927
strace: Process 2927 attached
write(1, "order to figure out what support"..., 4096) = 4096
read(0, "", 4096)                       = 0
write(1, "rror.  It is ad\342\200\220\n       visabl"..., 3771) = 3771
exit_group(0)                           = ?
+++ exited with 0 +++
nikiforova@debian:~$
```

Задание 2. Сборка и загрузка модуля в ядро Linux

1. Установите необходимые пакеты:

apt-get install gcc make linux-headers-$(uname -r)

2. Создайте файл модуля:

mkdir kmod-helloworld

cd kmod-helloworld/

touch ./mhello.c

с содержимым:

```
#define MODULE
#include <linux/module.h>
#include <linux/init.h>
#include <linux/kernel.h>
MODULE_LICENSE("GPLv3");
int init_module(void){
printk("<1> Hello,World\n");
return 0;
}
void cleanup_module(void){
printk("<1> Goodbye.\n");
}
```

```
nikiforova@debian:~$ mkdir kmod-helloworld
nikiforova@debian:~$ cd kmod-helloworld
nikiforova@debian:~/kmod-helloworld$ touch ./mhello.c
nikiforova@debian:~/kmod-helloworld$
```

```
  GNU nano 7.2                          mhello.c
#define MODULE
#include <linux/module.h>
#include <linux/init.h>
#include <linux/kernel.h>
MODULE_LICENSE("GPLv3");
int init_module(void) {
        printk("<1> Hello, World\n");
        return 0;
}
void cleanup_module(void) {
        printk("<1> Goodbye.\n");
}
```

3. Создайте Makefile:
touch ./Makefile
с содержимым:
obj-m += mhello.o3
hello-objs := mhello.c
all:
make -C /lib/modules/$(shell uname -r)/build/ M=$(PWD) modules
clean:
make -C /lib/modules/$(shell uname -r)/build/ M=$(PWD) clean
Перед командой "make" необходимо использовать табуляцию для создания отступа, а не пробелы.

```
  GNU nano 7.2                          Makefile
obj-m += mhello.o3
hello-objs := mhello.c
all:
        make -C /lib/modules/6.1.0-12-amd64/build/ M=$(pwd) modules
clean:
        make -C /lib/modules/6.1.0-12-amd64/build/ M=$(pwd) clean
```

Из-за данной проблемы дальнейшая часть работы была выполнена на другом компьютере

```
root@debian:/home/nikiforova/kmod-helloworld# make all
make -C /lib/modules/6.1.0-12-amd64/build/ M=/home/nikiforova/kmod-helloworld
modules
make[1]: вход в каталог «/usr/src/linux-headers-6.1.0-12-amd64»
make[2]: *** Нет правила для сборки цели «/home/nikiforova/kmod-helloworld/mhe
llo.o3», требуемой для «/home/nikiforova/kmod-helloworld/modules.order».  Оста
нов.
make[1]: *** [/usr/src/linux-headers-6.1.0-12-common/Makefile:2037: /home/niki
forova/kmod-helloworld] Ошибка 2
make[1]: выход из каталога «/usr/src/linux-headers-6.1.0-12-amd64»
make: *** [Makefile:4: all] Ошибка 2
root@debian:/home/nikiforova/kmod-helloworld#
```

4. Соберите модуль
make all
и установите его с помощью insmod:

```
make[1]: Entering directory '/usr/src/linux-headers-6.2.0-32-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 12.3.0-1ubuntu1~23.04
) 12.3.0
  You are using:           gcc-12 (Ubuntu 12.3.0-1ubuntu1~23.04) 12.3.0
make[1]: Leaving directory '/usr/src/linux-headers-6.2.0-32-generic'
```

sudo insmod yhello .ko

После этого модуль появится в списке установленных модулей:

lsmod

При помощи команды dmesg выведите буфер сообщений от ядра:

sudo dmesg

В конце вывода должно отобразиться сообщение от установленного модуля «Hello, World”.

В отчете по работе приведите снимки экрана установки модуля, результатов выполнения **dmesg** и **lsmod**.

```
[   37.699449] [drm] Initialized vmwgfx 2.20.0 20211206 for 0000:00:02.0 on minor 0
[   37.700264] fbcon: vmwgfxdrmfb (fb0) is primary device
[   37.701816] Console: switching to colour frame buffer device 160x50
[   37.706454] vmwgfx 0000:00:02.0: [drm] fb0: vmwgfxdrmfb frame buffer device
[   38.347979] intel_rapl_common: Found RAPL domain package
[   38.347986] intel_rapl_common: Found RAPL domain core
[   38.537692] snd_intel8x0 0000:00:05.0: allow list rate for 1028:0177 is 48000
[   41.957994] kauditd_printk_skb: 39 callbacks suppressed
[   41.958000] audit: type=1400 audit(1696588780.952:51): apparmor="STATUS" operation="profile_replace" info="same as c
urrent profile, skipping" profile="unconfined" name="rsyslogd" pid=665 comm="apparmor_parser"
[   48.511169] e1000: enp0s3 NIC Link is Up 1000 Mbps Full Duplex, Flow Control: RX
[   48.511470] IPv6: ADDRCONF(NETDEV_CHANGE): enp0s3: link becomes ready
[   50.148767] audit: type=1400 audit(1696588789.144:52): apparmor="DENIED" operation="capable" class="cap" profile="/u
sr/sbin/cupsd" pid=752 comm="cupsd" capability=12  capname="net_admin"
[   51.473526] loop12: detected capacity change from 0 to 8
[   56.680153] audit: type=1400 audit(1696588795.676:53): apparmor="DENIED" operation="capable" class="cap" profile="/s
nap/snapd/20092/usr/lib/snapd/snap-confine" pid=895 comm="snap-confine" capability=12  capname="net_admin"
[   56.680161] audit: type=1400 audit(1696588795.676:54): apparmor="DENIED" operation="capable" class="cap" profile="/s
nap/snapd/20092/usr/lib/snapd/snap-confine" pid=895 comm="snap-confine" capability=38  capname="perfmon"
[   72.157254] rfkill: input handler disabled
[  102.728510] systemd-journald[222]: File /var/log/journal/6f5883b2448646e493af7230dc8246fc/user-1000.journal corrupte
d or uncleanly shut down, renaming and replacing.
[  102.931607] audit: type=1400 audit(1696588844.275:55): apparmor="DENIED" operation="capable" class="cap" profile="/s
nap/snapd/20092/usr/lib/snapd/snap-confine" pid=1362 comm="snap-confine" capability=12  capname="net_admin"
[  102.932425] audit: type=1400 audit(1696588844.275:56): apparmor="DENIED" operation="capable" class="cap" profile="/s
nap/snapd/20092/usr/lib/snapd/snap-confine" pid=1362 comm="snap-confine" capability=38  capname="perfmon"
[  103.124656] rfkill: input handler enabled
[  111.157145] rfkill: input handler disabled
[  650.049563] mhello: loading out-of-tree module taints kernel.
[  650.049570] mhello: module license 'GPLv3' taints kernel.
[  650.049571] Disabling lock debugging due to kernel taint
[  650.049614] mhello: module verification failed: signature and/or required key missing - tainting kernel
[  650.050032] <1> Hello,World
```

5. Выгрузите модуль с помощью команды **rmmod** и включите снимок экрана вывода в отчет. Убедитесь, что модуль выгружен с помощью **dmesg** и **lsmod.**

```
[   37.700264] fbcon: vmwgfxdrmfb (fb0) is primary device
[   37.701816] Console: switching to colour frame buffer device 160x50
[   37.706454] vmwgfx 0000:00:02.0: [drm] fb0: vmwgfxdrmfb frame buffer device
[   38.347979] intel_rapl_common: Found RAPL domain package
[   38.347986] intel_rapl_common: Found RAPL domain core
[   38.537692] snd_intel8x0 0000:00:05.0: allow list rate for 1028:0177 is 48000
[   41.957994] kauditd_printk_skb: 39 callbacks suppressed
[   41.958000] audit: type=1400 audit(1696588780.952:51): apparmor="STATUS" operation="profile_replace" info="same as c
urrent profile, skipping" profile="unconfined" name="rsyslogd" pid=665 comm="apparmor_parser"
[   48.511169] e1000: enp0s3 NIC Link is Up 1000 Mbps Full Duplex, Flow Control: RX
[   48.511470] IPv6: ADDRCONF(NETDEV_CHANGE): enp0s3: link becomes ready
[   50.148767] audit: type=1400 audit(1696588789.144:52): apparmor="DENIED" operation="capable" class="cap" profile="/u
sr/sbin/cupsd" pid=752 comm="cupsd" capability=12  capname="net_admin"
[   51.473526] loop12: detected capacity change from 0 to 8
[   56.680153] audit: type=1400 audit(1696588795.676:53): apparmor="DENIED" operation="capable" class="cap" profile="/s
nap/snapd/20092/usr/lib/snapd/snap-confine" pid=895 comm="snap-confine" capability=12  capname="net_admin"
[   56.680161] audit: type=1400 audit(1696588795.676:54): apparmor="DENIED" operation="capable" class="cap" profile="/s
nap/snapd/20092/usr/lib/snapd/snap-confine" pid=895 comm="snap-confine" capability=38  capname="perfmon"
[   72.157254] rfkill: input handler disabled
[  102.728510] systemd-journald[222]: File /var/log/journal/6f5883b2448646e493af7230dc8246fc/user-1000.journal corrupte
d or uncleanly shut down, renaming and replacing.
[  102.931607] audit: type=1400 audit(1696588844.275:55): apparmor="DENIED" operation="capable" class="cap" profile="/s
nap/snapd/20092/usr/lib/snapd/snap-confine" pid=1362 comm="snap-confine" capability=12  capname="net_admin"
[  102.932425] audit: type=1400 audit(1696588844.275:56): apparmor="DENIED" operation="capable" class="cap" profile="/s
nap/snapd/20092/usr/lib/snapd/snap-confine" pid=1362 comm="snap-confine" capability=38  capname="perfmon"
[  103.124656] rfkill: input handler enabled
[  111.157145] rfkill: input handler disabled
[  650.049563] mhello: loading out-of-tree module taints kernel.
[  650.049570] mhello: module license 'GPLv3' taints kernel.
[  650.049571] Disabling lock debugging due to kernel taint
[  650.049614] mhello: module verification failed: signature and/or required key missing - tainting kernel
[  650.050032] <1> Hello,World
[ 1164.215618] <1> Goodbye.
```

Вывод

В результате выполненной работы были получены представления о составе дистрибутивов Linux, приобретены практические навыки установки и запуска дистрибутива Linux в виртуальной машине Oracle VM VirtualBox, получения справочной информации о системе и установленных приложениях с помощью интерфейса командной строки.