



Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Московский государственный технический университет  
имени Н.Э. Баумана  
(национальный исследовательский университет)»  
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

## ОТЧЕТ

*к лабораторной работе №3*

*По курсу: «Операционные системы»*

*На тему: «Загружаемые модули ядра»*

Студентка ИУ7-65Б  
Оберган Т.М

Преподаватель  
Рязанова Н.Ю.

2020 г.

## Оглавление

Часть 1 .....	3
Листинг .....	3
Результат работы программы.....	4
Часть 2 .....	5
Задание.....	5
Листинг .....	5
Результат работы программы.....	7

# Часть 1

## Листинг

### Содержимое makefile:

```
ifneq ($(KERNELRELEASE),)
    obj-m := md.o
else
    CURRENT = $(shell uname -r)
    KDIR = /lib/modules/$(CURRENT)/build
    PWD = $(shell pwd)

default:
    $(MAKE) -C $(KDIR) M=$(PWD) modules
    make clean

clean:
    @rm -f *.o *.cmd *.flags *.mod.c *.order
    @rm -f *.*.cmd ~~ *.~~ TODO.*
    @rm -fR .tmp*
    @rm -rf .tmp_versions

disclean: clean
    @rm *.ko *.symvers
endif
```

### Содержимое md.c:

```
#include <linux/module.h>
#include <linux/kernel.h>
#include <linux/init.h>
#include <linux/sched.h>
#include <linux/init_task.h>

MODULE_LICENSE("Dual BSD/GPL");
MODULE_AUTHOR("Obergan T.M");

static int __init md_init(void)
{
    printk("LAB3: Hello world.\n");
    struct task_struct *task = &init_task;
    do
    {
        printk("LAB3:---%s-%d, parent %s-%d", task->comm,
            task->pid, task->parent->comm, task->parent->pid);
    } while ((task = next_task(task)) != &init_task);
    printk("LAB3:---%s-%d, parent %s-%d", current->comm,
        current->pid, current->parent->comm, current->parent->pid);
    return 0;
}

static void __exit md_exit(void)
{
    printk("LAB3: Goodbye.\n");
}

module_init(md_init);
module_exit(md_exit);
```

## Результат работы программы

Компиляция модуля при помощи makefile:

```
[winterpuma@winterpuma ~]$ make
make -C /lib/modules/5.4.28-rt19-MANJARO/build M=/home/winterpuma modules
make[1]: Entering directory '/usr/lib/modules/5.4.28-rt19-MANJARO/build'
  CC [M]  /home/winterpuma/md.o
  Building modules, stage 2.
  MODPOST 1 modules
  CC [M]  /home/winterpuma/md.mod.o
  LD [M]  /home/winterpuma/md.ko
make[1]: Leaving directory '/usr/lib/modules/5.4.28-rt19-MANJARO/build'
make clean
make[1]: Entering directory '/home/winterpuma'
make[1]: Leaving directory '/home/winterpuma'
```

Загрузка модуля при помощи команды insmod:

```
[winterpuma@winterpuma ~]$ sudo insmod md.ko
[sudo] password for winterpuma:
[winterpuma@winterpuma ~]$ lsmod | grep md
md                16384  0
crypto_simd       16384  1 aesni_intel
cryptd            24576  4 crypto_simd,ghash_clmulni_intel
```

```
[winterpuma@winterpuma ~]$ sudo dmesg | grep LAB3
[ 630.214562] LAB3: Hello world.
[ 630.214562] LAB3:---swapper/0-0, parent swapper/0-0
[ 630.214582] LAB3:---systemd-1, parent swapper/0-0
[ 630.214582] LAB3:---kthreadd-2, parent swapper/0-0
[ 630.214583] LAB3:---rcu_gp-3, parent kthreadd-2
[ 630.214584] LAB3:---rcu_par_gp-4, parent kthreadd-2
[ 630.214584] LAB3:---kworker/0:0H-6, parent kthreadd-2
[ 630.214585] LAB3:---kworker/u16:0-7, parent kthreadd-2
[ 630.214585] LAB3:---mm_percpu_wq-8, parent kthreadd-2
[ 630.214586] LAB3:---ksoftirqd/0-9, parent kthreadd-2
[ 630.214586] LAB3:---rcuc/0-10, parent kthreadd-2
[ 630.214587] LAB3:---rcu_preempt-11, parent kthreadd-2
[ 630.214587] LAB3:---rcub/0-12, parent kthreadd-2
[ 630.214588] LAB3:---posixcpu_tmr/0-13, parent kthreadd-2
```

```
...
[ 630.214789] LAB3:---thumbnail.so-2198, parent kdeinit5-1239
[ 630.214790] LAB3:---kworker/u16:1-2238, parent kthreadd-2
[ 630.214790] LAB3:---konsole-2243, parent plasmashell-1308
[ 630.214791] LAB3:---bash-2251, parent konsole-2243
[ 630.214792] LAB3:---tags.so-2645, parent kdeinit5-1239
[ 630.214792] LAB3:---sudo-2661, parent bash-2251
[ 630.214792] LAB3:---insmod-2662, parent sudo-2661
[ 630.214793] LAB3:---insmod-2662, parent sudo-2661
```

Выгрузка модуля при помощи команды rmmod:

```
[winterpuma@winterpuma ~]$ sudo insmod md.ko
[winterpuma@winterpuma ~]$ sudo rmmod md
[winterpuma@winterpuma ~]$ sudo dmesg | grep LAB3 | tail -1
[ 747.582877] LAB3: Goodbye.
[winterpuma@winterpuma ~]$
```

## Часть 2

### Задание

Реализовать три загружаемых модуля ядра:

- Вызываемый модуль md1
- Вызывающий модуль md2
- «Отладочный» модуль md3

Каждый загружаемый модуль должен содержать:

- Указание лицензии GPL
- Указание автора

Загружаемые модули должны собираться при помощи Make-файла (сборка командой make). **Вызов каждой функции модуля должен сопровождаться записью в системный журнал информации, какая функция какого модуля была вызвана.**

### Листинг

md.h:

```
//#ifndef MY_MD
//#define MY_MD 1

extern char* md1_data;
extern char* md1_proc( void );
extern char* md1_noexport( void );

//#endif
```

md1.c

```
#include <linux/init.h>
#include <linux/module.h>
#include "md.h"

MODULE_LICENSE("Dual BSD/GPL");
MODULE_AUTHOR("Obergan T.M");

char* md1_data = "Hello world!";

extern char* md1_proc(void)
{
    return md1_data;
}

static char* md1_local(void)
{
    return md1_data;
}
```

```

extern char* md1_noexport(void)
{
    return md1_data;
}

EXPORT_SYMBOL(md1_data);
EXPORT_SYMBOL(md1_proc);

static int __init md_init(void)
{
    printk("LAB3: module md1 start!\n");
    return 0;
}

static void __exit md_exit(void)
{
    printk("LAB3: module md1 unloaded!\n");
}

module_init(md_init);
module_exit(md_exit);

```

## md2.c

```

#include <linux/init.h>
#include <linux/module.h>
#include "md.h"

MODULE_LICENSE("Dual BSD/GPL");
MODULE_AUTHOR("Obergan T.M");

static int __init md_init(void)
{
    printk("LAB3: module md2 start!\n");
    printk("LAB3: data string exported from md1 : %s\n", md1_data);
    printk("LAB3: string returned md1_proc() is : %s\n", md1_proc());
    return 0;
}

static void __exit md_exit(void)
{
    printk("LAB3: module md2 unloaded!\n");
}

module_init(md_init);
module_exit(md_exit);

```

## md3.c

```

#include <linux/init.h>
#include <linux/module.h>
#include "md.h"

MODULE_LICENSE("Dual BSD/GPL");
MODULE_AUTHOR("Obergan T.M");

static int __init md_init(void)
{
    printk("LAB3: module md3 start!\n");
    printk("LAB3: data string exported from md1 : %s\n", md1_data);
    printk("LAB3: string returned md1_proc() is : %s\n", md1_proc());
}

```

```

    return -1;
}

static void __exit md_exit(void)
{
    printk("LAB3: module md3 unloaded!\n");
}

module_init(md_init);
module_exit(md_exit);

```

## Результат работы программы

Так как md2 и md3 используют данные из md1, то при попытке загрузки этих модулей до загрузки md1 возникнет ошибка.

```

[winterpuma@winterpuma ~]$ sudo insmod md2.ko
insmod: ERROR: could not insert module md2.ko: Unknown symbol in module
[winterpuma@winterpuma ~]$ sudo dmesg | grep md2
[ 1215.024415] md2: Unknown symbol md1_data (err -2)
[ 1215.024439] md2: Unknown symbol md1_proc (err -2)
[winterpuma@winterpuma ~]$
[winterpuma@winterpuma ~]$
[winterpuma@winterpuma ~]$
[winterpuma@winterpuma ~]$ sudo insmod md3.ko
insmod: ERROR: could not insert module md3.ko: Unknown symbol in module
[winterpuma@winterpuma ~]$ sudo dmesg | grep md3
[ 1233.426307] md3: Unknown symbol md1_data (err -2)
[ 1233.426331] md3: Unknown symbol md1_proc (err -2)
[winterpuma@winterpuma ~]$

```

При загрузке модулей в правильном порядке md1 и md2 будут успешно загружены, а md3 не будет загружен т.к. функция инициализации md3 вернула ненулевое значение (но код инициализирующей функции будет выполнен, что можно наблюдать в буфере сообщений ядра).

```

[winterpuma@winterpuma ~]$ sudo insmod md1.ko
[winterpuma@winterpuma ~]$ sudo dmesg | grep LAB3
[ 1301.964092] LAB3: module md1 start!
[winterpuma@winterpuma ~]$ sudo insmod md2.ko
[winterpuma@winterpuma ~]$ sudo insmod md3.ko
insmod: ERROR: could not insert module md3.ko: Operation not permitted
[winterpuma@winterpuma ~]$ sudo dmesg | grep LAB3
[ 1301.964092] LAB3: module md1 start!
[ 1341.690637] LAB3: module md2 start!
[ 1341.690638] LAB3: data string exported from md1 : Hello world!
[ 1341.690638] LAB3: string returned md1_proc() is : Hello world!
[ 1348.356556] LAB3: module md3 start!
[ 1348.356557] LAB3: data string exported from md1 : Hello world!
[ 1348.356557] LAB3: string returned md1_proc() is : Hello world!

```

Модуль md1 используется модулем md2:

```
[winterpuma@winterpuma ~]$ sudo lsmod | grep md
md2                16384  0
md1                16384  1 md2
crypto_simd        16384  1 aesni_intel
cryptd             24576  4 crypto_simd,ghash_clmulni_intel
```

При попытке выгрузки md2 до md1 возникнет следующая ошибка:

```
[winterpuma@winterpuma ~]$ sudo rmmod md1
rmmod: ERROR: Module md1 is in use by: md2
```

Для успешной выгрузки, модули нужно выгружать в обратном порядке (сначала md2, а потом md1).

```
[winterpuma@winterpuma ~]$ sudo rmmod md2
[winterpuma@winterpuma ~]$ sudo rmmod md1
[winterpuma@winterpuma ~]$ sudo dmesg | grep LAB3 | tail -2
[ 1484.372401] LAB3: module md2 unloaded!
[ 1487.478188] LAB3: module md1 unloaded!
[winterpuma@winterpuma ~]$
```