

# **Отчёт по лабораторной работе №10**

**Основы работы с модулями ядра операционной системы**

Наурузова Айшат Магометовна

# **Содержание**

<b>1 Цель работы</b>	<b>5</b>
<b>2 Ход выполнения</b>	<b>6</b>
2.1 Управление модулями ядра из командной строки . . . . .	6
2.2 Загрузка модулей ядра с параметрами . . . . .	9
2.3 Обновление ядра системы . . . . .	10
<b>3 Контрольные вопросы</b>	<b>12</b>
<b>4 Заключение</b>	<b>14</b>

# Список иллюстраций

2.1	Вывод команды lspci -k . . . . .	6
2.2	Список загруженных модулей ядра . . . . .	7
2.3	Загрузка и проверка модуля ext4 . . . . .	8
2.4	Выгрузка модулей ext4 и xfs . . . . .	8
2.5	Загрузка модуля bluetooth . . . . .	9
2.6	Информация о модуле bluetooth и его выгрузка . . . . .	10
2.7	Проверка версии и доступных пакетов ядра . . . . .	10
2.8	Обновление ядра и системы . . . . .	11
2.9	Проверка версий ядра и системных сведений . . . . .	11

# **Список таблиц**

# **1 Цель работы**

Получить навыки работы с утилитами управления модулями ядра операционной системы.

## 2 Ход выполнения

### 2.1 Управление модулями ядра из командной строки

После входа в систему были получены права администратора.

Выполнен просмотр подключённых устройств и связанных с ними модулей ядра с помощью команды `lspci -k`.

Команда вывела список всех устройств, включая сетевые, графические, USB и звуковые адаптеры. Для каждого устройства указаны загруженные драйверы ядра и модули, обеспечивающие их работу.

Например, для сетевого контроллера Intel PRO/1000 MT Desktop Adapter используется модуль `e1000`, а для видеокарты — `vmwgfx`.

```
amnauruzova@amnauruzova:~$ su
Password:
root@amnauruzova:/home/amnauruzova# lspci -k
00:00.0 Host bridge: Intel Corporation 440FX - 82441FX PMC [Natoma] (rev 02)
00:01.0 ISA bridge: Intel Corporation 82371SB PIIX3 ISA [Natoma/Triton II]
00:01.1 IDE interface: Intel Corporation 82371AB/EB/MB PIIX4 IDE (rev 01)
    Kernel driver in use: ata_piix
    Kernel modules: ata_piix, ata_generic
00:02.0 VGA compatible controller: VMware SVGA II Adapter
    Subsystem: VMware SVGA II Adapter
    Kernel driver in use: vmwgfx
    Kernel modules: vmwgfx
00:03.0 Ethernet controller: Intel Corporation 82540EM Gigabit Ethernet Controller (rev 02)
    Subsystem: Intel Corporation PRO/1000 MT Desktop Adapter
    Kernel driver in use: e1000
    Kernel modules: e1000
00:04.0 System peripheral: InnoTek Systemberatung GmbH VirtualBox Guest Service
    Kernel driver in use: vboxguest
    Kernel modules: vboxguest
00:05.0 Multimedia audio controller: Intel Corporation 82801AA AC'97 Audio Controller (rev 01)
    Subsystem: Dell Device 0177
    Kernel driver in use: snd_intel8x0
    Kernel modules: snd_intel8x0
00:06.0 USB controller: Apple Inc. KeyLargo/Intrepid USB
    Kernel driver in use: ohci-pci
00:07.0 Bridge: Intel Corporation 82371AB/EB/MB PIIX4 ACPI (rev 08)
    Kernel driver in use: piix4_smbus
    Kernel modules: i2c_piix4
00:0b.0 USB controller: Intel Corporation 82801FB/FBM/FR/FW/FRW (ICH6 Family) USB2 EHCI Controller
    Kernel driver in use: ehci-pci
00:0d.0 SATA controller: Intel Corporation 82801HM/HEM (ICH8M/ICH8M-E) SATA Controller [AHCI mode] (rev 02)
    Kernel driver in use: ahci
    Kernel modules: ahci
root@amnauruzova:/home/amnauruzova#
```

Рис. 2.1: Вывод команды `lspci -k`

Для отображения всех загруженных модулей ядра была выполнена команда `lsmod | sort`.

Результат показал множество модулей, загруженных в систему, включая e1000, vmwgfx, snd\_intel8x0, ata\_piix и другие. Это свидетельствует о корректной работе драйверов для устройств, установленных в системе.

```
root@amnauruzova:/home/amnauruzova# lsmod | sort
ac97_bus           12288  1 snd_ac97_codec
ahci              57344  3
ata_generic        16384  0
ata_piix          45056  1
cdrom             90112  2 isofs,sr_mod
crc32c_intel      16384  0
crc32_pclmul     12288  0
crct10dif_pclmul 12288  1
dm_log            24576  2 dm_region_hash,dm_mirror
dm_mirror         28672  0
dm_mod            245760 9 dm_multipath,dm_log,dm_mirror
dm_multipath      53248  0
dm_region_hash    28672  1 dm_mirror
drm_ttm_helper    16384  2 vmwgfx
e1000             200704  0
fuse              253952  5
ghash_clmulni_intel 16384  0
i2c_piix4         36864  0
i2c_smbus         20480  1 i2c_piix4
intel_pmc_core    126976  0
intel_rapl_common 53248  1 intel_rapl_msrm
intel_rapl_msrm   20480  0
intel_uncore_frequency_common 16384  0
intel_vsec         20480  1 intel_pmc_core
ip_set             69632  0
isofs              69632  1
joydev             28672  0
libahci            69632  1 ahci
libata             512000 4 ata_piix,libahci,ahci,ata_generic
loop               45056  0
Module             Size Used by
nf_conntrack       204800 2 nf_nat,nft_ct
nf_defrag_ipv4     12288  1 nf_conntrack
nf_defrag_ipv6     24576  1 nf_conntrack
```

Рис. 2.2: Список загруженных модулей ядра

Для проверки, загружен ли модуль файловой системы ext4, использовалась команда `lsmod | grep ext4`.

Модуль не был найден, поэтому был выполнен его запуск с помощью `modprobe ext4`.

После повторной проверки модуль успешно появился в списке.

```

root@amnauruzova:/home/amnauruzova# lsmod | grep ext4
root@amnauruzova:/home/amnauruzova# modprobe ext4
root@amnauruzova:/home/amnauruzova# lsmod | grep ext4
ext4           1187840  0
mbcache          16384  1 ext4
jbd2            217088  1 ext4
root@amnauruzova:/home/amnauruzova# modinfo ext4
filename:       /lib/modules/6.12.0-55.12.1.el10_0.x86_64/kernel/fs/ext4/ext4.ko.xz
softdep:        pre: crc32c
license:        GPL
description:   Fourth Extended Filesystem
author:         Remy Card, Stephen Tweedie, Andrew Morton, Andreas Dilger, Theodore Ts'o and others
alias:          fs-ext4
alias:          ext3
alias:          fs-ext3
alias:          ext2
alias:          fs-ext2
rhelversion:    10.0
srcversion:     CBA9BD0FC931061AEC0A8F4
depends:        jbd2,mbcache
intree:         Y
name:          ext4
retpoline:      Y
vermagic:      6.12.0-55.12.1.el10_0.x86_64 SMP preempt mod_unload modversions
sig_id:         PKCS#7
signer:         Rocky kernel signing key
sig_key:        57:BC:A1:34:94:0D:57:12:99:67:28:D8:C0:8B:1C:4B:8C:8A:13:06
sig_hashalgo:   sha256
signature:      05:D7:3D:A0:55:EA:4A:16:70:78:23:1F:C7:9A:17:C6:A9:9C:B5:71:
                6B:EB:97:2C:94:5A:49:CB:07:A0:85:2E:27:8D:C4:01:27:43:2A:C8:
                B8:37:9C:AB:EF:B7:D9:AC:70:98:F6:2B:1B:A6:90:7D:62:0E:8C:2E:
                4E:D6:0E:3F:B9:33:E7:D9:8E:05:F0:70:09:81:08:5F:1E:26:99:E4:
                C2:33:CD:4A:01:R7:F3:17:11:06:R1:R2:F8:06:23:85:76:76:43:9R:

```

Рис. 2.3: Загрузка и проверка модуля ext4

Информация о модуле была получена с помощью `modinfo ext4`.

Были выведены сведения о местоположении модуля, лицензии GPL, авторах (Remy Card, Stephen Tweedie и др.), а также зависимостях `jbd2` и `mbcache`. Отмечено, что модуль не имеет параметров настройки.

Для удаления модуля `ext4` была выполнена команда `modprobe -r ext4`.

Модуль выгружен без ошибок. Однако при попытке выгрузить модуль `xfs` система выдала сообщение об ошибке «`FATAL: Module xfs is in use`», что означает, что модуль используется файловой системой и не может быть удалён.

```

root@amnauruzova:/home/amnauruzova# modprobe -r ext4
root@amnauruzova:/home/amnauruzova# modprobe -r ext4
root@amnauruzova:/home/amnauruzova# modprobe -r xfs
modprobe: FATAL: Module xfs is in use.
root@amnauruzova:/home/amnauruzova#

```

Рис. 2.4: Выгрузка модулей ext4 и xfs

## 2.2 Загрузка модулей ядра с параметрами

Для проверки наличия модуля Bluetooth использовалась команда `lsmod | grep bluetooth`.

Модуль отсутствовал, после чего он был загружен при помощи `modprobe bluetooth`.

Повторная проверка показала, что модуль bluetooth успешно активирован.

```
root@annauruzova:/home/annauruzova# lsmod | grep bluetooth
root@annauruzova:/home/annauruzova# modprobe bluetooth
root@annauruzova:/home/annauruzova# lsmod | grep bluetooth
bluetooth           1114112  0
rfkill                  40960  4 bluetooth
filename:          /lib/modules/6.12.0-55.12.1.el10_0.x86_64/kernel/net/bluetooth/bluetooth.ko.xz
alias:             net-pf-31
license:            GPL
version:           2.22
description:       Bluetooth Core ver 2.22
author:            Marcel Holtmann <marcel@holtmann.org>
rhelversion:        10.0
srcversion:         5F9AA895ADC3BA3840211D1
depends:            rfkill
intree:             Y
name:              bluetooth
retpoline:          Y
vermagic:          6.12.0-55.12.1.el10_0.x86_64 SMP preempt mod_unload modversions
sig_id:             PKCS#7
signer:             Rocky kernel signing key
sig_key:            57:BC:A1:34:94:00:57:12:99:67:28:D8:C0:8B:1C:4B:8C:8A:13:06
sig_hashalgo:       sha256
signature:          2E:24:0B:EE:9E:E6:A3:58:91:2A:D3:87:25:A3:1B:46:06:89:06:BC:
                   4A:99:10:80:7A:0F:1E:FF:92:59:FD:59:74:DB:7E:42:40:14:03:C9:
                   43:D0:A3:4F:F3:10:71:07:4D:FF:34:52:96:1A:8B:90:9C:DF:D8:FB:
                   07:E9:07:16:2E:8D:4B:23:0C:92:93:0B:88:E9:82:6B:06:1B:BD:25:
                   69:7D:37:70:0C:C2:A5:0B:85:C9:34:D9:69:1C:28:BB:B1:B0:8A:C7:
                   9F:4D:5D:46:4R:R1:C4:F3:RR:1A:12:5D:F2:52:00:42:00:FD:
```

Рис. 2.5: Загрузка модуля bluetooth

Информация о модуле получена с помощью `modinfo bluetooth`.

Выведено описание — Bluetooth Core ver 2.22, лицензия GPL, автор Marcel Holtmann.

Зависимость — модуль rfkill.

Модуль поддерживает следующие параметры:

- disable\_esco — отключение eSCO-соединений
- disable\_ertm — отключение режима расширенной передачи
- enable\_ecred — включение улучшенного потока управления

После проверки модуль был выгружен командой `modprobe -r bluetooth`.

```
54:25:78:36:FC:A5:17:C0:5A:62:C5:4C:41:F6:FF:87:5F:0C:58:45:  
AF:12:D3:12:5D:11:49:20:BD:B1:E1:A4:DB:BA:D3:18:B6:EE:D0:57:  
EC:75:3A:A6:0D:E2:36:37:78:C6:BB:DD:9B:1C:67:EB:D2:A8:9F:16:  
2B:8A:CF:8F:23:38:B4:7A:A9:92:49:86:D7:9F:08:7D:8A:92:7D:DD:  
41:D6:4B:31:F6:45:31:D4:7E:6E:E1:D6:94:BA:D0:6A:12:CC:76:A2:  
12:55:EF:6B:91:CB:45:F0:F5:75:08:DD:7D:6B:D7:E1:B0:13:51:40:  
29:D2:1D:3B:DD:E2:BF:AC:4B:77:67:BA:C8:95:16:41:00:9A:8F:96:  
21:DC:9A:C2:B2:20:31:5C:EC:90:EA:B0:24:C3:5D:B9:72:31:B8:98:  
DB:A9:4D:FE  
parm: disable_esco:Disable eSCO connection creation (bool)  
parm: disable_erm:Disable enhanced retransmission mode (bool)  
parm: enable_ecred:Enable enhanced credit flow control mode (bool)  
root@amnauruzova:/home/amnauruzova#  
root@amnauruzova:/home/amnauruzova# modprobe -r bluetooth  
root@amnauruzova:/home/amnauruzova#
```

Рис. 2.6: Информация о модуле bluetooth и его выгрузка

## 2.3 Обновление ядра системы

Для проверки текущей версии ядра использовалась команда `uname -r`.

В системе установлено ядро версии `6.12.0-55.12.1.el10_0.x86_64`.

Для просмотра доступных пакетов ядра применена команда `dnf list kernel`.

В списке пакетов видно, что доступна более новая версия ядра — `6.12.0-55.37.1.el10_0.x86_64`.

```
root@amnauruzova:/home/amnauruzova#  
root@amnauruzova:/home/amnauruzova# uname -r  
6.12.0-55.12.1.el10_0.x86_64  
root@amnauruzova:/home/amnauruzova# dnf list kernel  
Rocky Linux 10 - BaseOS 5.9 kB/s | 4.3 kB 00:00  
Rocky Linux 10 - AppStream 3.3 kB/s | 4.3 kB 00:01  
Rocky Linux 10 - Extras 8.1 kB/s | 3.1 kB 00:00  
Installed Packages  
kernel.x86_64 6.12.0-55.12.1.el10_0 @anaconda  
Available Packages  
kernel.x86_64 6.12.0-55.37.1.el10_0 baseos  
root@amnauruzova:/home/amnauruzova#
```

Рис. 2.7: Проверка версии и доступных пакетов ядра

Для обновления были выполнены команды `dnf update kernel`, `dnf update`, `dnf upgrade --refresh`.

Система подтвердила отсутствие зависимостей и корректное завершение обновления.

```

zlib-ng-compat-devel-2.2.3-1.el10.rocky.0.1.x86_64
Installed:
kernel-6.12.0-55.37.1.el10_0.x86_64
kernel-devel-6.12.0-55.37.1.el10_0.x86_64
kernel-modules-core-6.12.0-55.37.1.el10_0.x86_64
libatomic-14.2.1-7.el10.x86_64
libdrex-0.8.1-1.el10.x86_64

Complete!
root@amnauruzova:/home/amnauruzova# dnf update kernel
Last metadata expiration check: 0:02:32 ago on Wed 15 Oct 2025 01:40:50 PM MSK.
Dependencies resolved.
Nothing to do.
Complete!
root@amnauruzova:/home/amnauruzova# dnf update
Last metadata expiration check: 0:02:36 ago on Wed 15 Oct 2025 01:40:50 PM MSK.
Dependencies resolved.
Nothing to do.
Complete!
root@amnauruzova:/home/amnauruzova# dnf upgrade --refresh
Rocky Linux 10 - BaseOS
Rocky Linux 10 - AppStream
Rocky Linux 10 - Extras
Dependencies resolved.
Nothing to do.
Complete!
root@amnauruzova:/home/amnauruzova# 

```

Рис. 2.8: Обновление ядра и системы

После перезагрузки и выбора нового ядра команда `uname -r` показала, что активна версия ядра `6.12.0-55.37.1.el10_0.x86_64`.

Команда `hostnamectl` отобразила информацию о системе: дистрибутив Rocky Linux 10.0, виртуализация Oracle (VirtualBox), архитектура `x86_64`.

```

amnauruzova@amnauruzova:~$ uname -r
6.12.0-55.37.1.el10_0.x86_64
amnauruzova@amnauruzova:~$ hostnamectl
  Static hostname: amnauruzova.localdomain
    Icon name: computer-vm
      Chassis: VM 🖥
    Machine ID: 1c80a20b5c184a759c9cd8fa84bc080c
      Boot ID: 1f78f442531842b8ba4735762e7ebceb
  Virtualization: oracle
Operating System: Rocky Linux 10.0 (Red Quartz)
  CPE OS Name: cpe:/o:rocky:rocky:10::baseos
    OS Support End: Thu 2035-05-31
  OS Support Remaining: 9y 7month 2w
        Kernel: Linux 6.12.0-55.37.1.el10_0.x86_64
      Architecture: x86-64
    Hardware Vendor: innotek GmbH
      Hardware Model: VirtualBox
    Firmware Version: VirtualBox
      Firmware Date: Fri 2006-12-01
      Firmware Age: 18y 10month 2w
amnauruzova@amnauruzova:~$ 

```

Рис. 2.9: Проверка версии ядра и системных сведений

## **3 Контрольные вопросы**

**1. Какая команда показывает текущую версию ядра, которая используется на вашей системе?**

Для этого используется команда `uname -r`.

Она выводит номер версии ядра, загруженного в данный момент.

**2. Как можно посмотреть более подробную информацию о текущей версии ядра операционной системы?**

Для получения детальных сведений используется команда `hostnamectl`.

Она показывает версию ядра, архитектуру системы, производителя оборудования и виртуализацию.

**3. Какая команда показывает список загруженных модулей ядра?**

Для отображения всех загруженных модулей применяется команда `lsmod`.

Она выводит список модулей с указанием их размера и количества использований.

**4. Какая команда позволяет вам определять параметры модуля ядра?**

Чтобы просмотреть параметры модуля, используется команда `modinfo <имя_модуля>`.

В выводе отображаются доступные параметры, их типы и описание.

**5. Как выгрузить модуль ядра?**

Для выгрузки модуля используется команда `modprobe -r <имя_модуля>`.

Она удаляет модуль из памяти, если он не используется другими процессами.

**6. Что вы можете сделать, если получите сообщение об ошибке при попытке выгрузить модуль ядра?**

Если система сообщает, что модуль используется, необходимо сначала остановить процесс или службу, задействующую этот модуль, либо размонтировать файловую систему, если она использует модуль.

После этого можно повторно выполнить команду выгрузки.

### **7. Как определить, какие параметры модуля ядра поддерживаются?**

Для этого также используется команда `modinfo <имя_модуля>`.

В разделе *param* указываются поддерживаемые параметры и их описание.

### **8. Как установить новую версию ядра?**

Для установки новой версии ядра применяется пакетный менеджер `dnf`.

Сначала выполняется обновление списка пакетов:

```
dnf upgrade --refresh
```

Затем обновление ядра:

```
dnf update kernel
```

После установки необходимо перезагрузить систему и выбрать новое ядро при старте.

## **4 Заключение**

В ходе лабораторной работы были изучены основные принципы управления модулями ядра в операционной системе Linux.

Были рассмотрены команды для просмотра, загрузки, выгрузки и анализа модулей, такие как `lsmod`, `modprobe` и `modinfo`.

На практике были загружены и исследованы модули файловых систем и сетевых устройств, а также выполнено обновление ядра системы до новой версии.