

# **Лабораторная работа №10**

**Отчет**

Устинова Виктория Вадимовна

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Задание</b>	<b>6</b>
<b>3</b>	<b>Выполнение лабораторной работы</b>	<b>7</b>
<b>4</b>	<b>Выводы</b>	<b>12</b>
<b>5</b>	<b>Ответы на контрольные вопросы</b>	<b>13</b>

# Список иллюстраций

3.1	разнообразные виртуальные устройства (мосты, IDE, VGA, Ethernet, аудио, USB, SATA-контроллеры) и соответствующие им драйверы ядра, например: ata_piix, vmwgfx, e1000, snd_intel8x0, ohci-pci, ehci-pci, ahci. . . . .	7
3.2	разнообразные виртуальные устройства (мосты, IDE, VGA, Ethernet, аудио, USB, SATA-контроллеры) и соответствующие им драйверы ядра, например: ata_piix, vmwgfx, e1000, snd_intel8x0, ohci-pci, ehci-pci, ahci. . . . .	8
3.3	смотри загруженные модули . . . . .	8
3.4	Загружаем модуль и убеждаемся в этом . . . . .	8
3.5	Конфигурация файловой системы ext4 происходит при её создании (mkfs.ext4) или монтировании (mount -o), а не через параметры самого модуля ядра, который лишь предоставляет базовую логику. . . . .	9
3.6	Не удалось выгрузить модуль, потому что он активно используется системой, что подтверждается ошибкой “is in use”. . . . .	9
3.7	Загружаем модуль блютуза и смотрим список модулей ядра отвечающих за работу с блютузом . . . . .	9
3.8	disable_esco: Отключает расширенные аудио-соединения (eSCO). disable_ertm: Отключает улучшенный режим повторной передачи данных (ERTM).enable_ecred: Включает улучшенный контроль потока данных (ECRED) для Bluetooth Low Energy. . . . .	10
3.9	Версия ядра и список пакетов . . . . .	10
3.10	Все уже обновлено . . . . .	10
3.11	После перезагрузки смотрим версию . . . . .	11

## **Список таблиц**

# 1 Цель работы

Получить навыки работы с утилитами управления модулями ядра операционной системы.

## 2 Задание

1. Продемонстрируйте навыки работы по управлению модулями ядра (см. раздел 10.4.1).
2. Продемонстрируйте навыки работы по загрузке модулей ядра с параметрами (см. раздел 10.4.2).

### 3 Выполнение лабораторной работы

Посмотрите, какие устройства имеются в вашей системе и какие модули ядра с ними связаны(рис. 3.1).

```
[root@vuvstinova ~]# 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
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)
```

Рис. 3.1: разнообразные виртуальные устройства (мосты, IDE, VGA, Ethernet, аудио, USB, SATA-контроллеры) и соответствующие им драйверы ядра, например: ata\_piix, vmwgfx, e1000, snd\_intel8x0, ohci-pci, ehci-pci, ahci.

Посмотрите, какие устройства имеются в вашей системе и какие модули ядра с ними связаны(рис. 3.2).

```
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
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/FR/FW/FW (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
```

Рис. 3.2: разнообразные виртуальные устройства (мосты, IDE, VGA, Ethernet, аудио, USB, SATA-контроллеры) и соответствующие им драйверы ядра, например: ata\_piix, vmwgfx, e1000, snd\_intel8x0, ohci-pci, ehci-pci, ahci.

Посмотрите, какие модули ядра загружены(рис. 3.3).

```
[root@vvustinova ~]# lsmod | sort
ac97_bus          12288  1 snd_ac97_codec
ahci              49152  3
ata_generic       16384  0
ata_piix          45056  1
cdrom             90112  2 isofs,sr_mod
crc32c_intel      24576  1
crc32_pclmul      12288  0
crt10dif_pclmul   12288  1
dm_log            24576  2 dm_region_hash,dm_mirror
dm_mirror         28672  0
dm_mod            245760  9 dm_log,dm_mirror
dm_region_hash    28672  1 dm_mirror
drm               811008  6 vmwgfx,drm_kms_helper,drm_ttm_helper,ttm
drm_kms_helper    266240  2 vmwgfx,drm_ttm_helper
drm_ttm_helper    16384  2 vmwgfx
e1000             196608  0
fuse              212992  5
ghash_clmulni_intel 16384  0
i2c_piix4         28672  0
```

Рис. 3.3: смотри загруженные модули

Посмотрите, загружен ли модуль ext4, загрузите модуль ядра ext4, убедитесь, что модуль загружен, посмотрев список загруженных модулей(рис. 3.4).

```
[root@vvustinova ~]# lsmod | grep ext4
[root@vvustinova ~]# modprobe ext4
[root@vvustinova ~]# lsmod | grep ext4
ext4              1191936  0
mbcache           16384  1 ext4
jbd2              221184  1 ext4
```

Рис. 3.4: Загружаем модуль и убеждаемся в этом

Посмотрите информацию о модуле ядра ext4, обратите внимание, что у этого



модуля нет параметров(рис. 3.5).

```
sig_hashalgo: sha256
signature: 61:BC:53:9F:C0:C1:2A:2D:B2:34:DF:10:67:4E:50:C6:10:A5:13:AB:
          2E:2B:4D:A7:EE:AD:A3:55:E7:39:D4:25:67:53:8B:1E:0B:8F:0F:05:
          90:48:2A:CA:64:CD:DF:62:5D:18:2B:9A:B2:D2:D1:6A:9B:18:DF:60:
          8B:12:FF:F0:DC:FB:1B:10:EE:F8:3C:FB:3E:BC:58:4C:5F:12:47:C3:
          4B:85:BD:A0:90:F2:36:E6:D7:A2:A2:60:59:79:DF:B5:EA:CF:16:7A:
          AF:D6:56:0D:5E:E4:34:16:0C:98:F2:46:5F:47:62:69:C8:0E:97:48:
          F8:03:00:DA:B9:CC:ED:40:C9:60:40:6D:24:17:AD:32:DE:9A:A6:DC:
          94:A6:A2:D4:7F:6D:0C:7C:B9:DC:8D:26:80:B1:BF:C0:37:15:F8:C5:
          46:53:8A:81:40:D4:02:A2:0F:DD:76:C9:F4:DE:70:16:6D:1B:E0:C9:
          4D:C5:3D:4F:EE:B6:15:41:F7:EC:00:59:8E:9A:A4:38:DA:F7:52:1C:
          B2:1A:F8:D9:74:D3:32:78:80:BE:B6:F7:E0:04:46:E4:00:25:74:AF:
          3C:F7:1D:7E:F1:9D:4B:27:B0:35:98:76:C8:66:4D:C2:BC:9E:54:32:
          5D:3C:1B:67:80:E8:1D:6D:14:E2:E0:31:FD:5A:1B:A8:88:6C:3F:92:
          B4:4B:B8:E5:D3:D0:7D:BD:22:76:57:20:28:62:55:B2:8A:0C:92:F3:
          57:E0:05:BF:95:92:23:79:36:C2:AA:11:EE:21:8F:23:9C:7E:7E:23:
          92:0C:B4:3C:DD:D5:85:6B:B7:97:64:E8:59:92:61:88:7D:93:C7:3F:
          4F:4B:BD:4C:35:F6:05:A4:B1:4E:61:1C:36:38:57:03:40:74:57:80:
          A4:6E:74:BA:F6:10:E7:29:73:6E:AE:53:2A:C8:00:E5:01:DD:98:B4:
          49:7A:0E:1F:B9:39:0C:03:24:5B:97:EE:E3:11:25:98:1D:41:37:5B:
          1C:9D:0D:F7
```

Рис. 3.5: Конфигурация файловой системы ext4 происходит при её создании (mkfs.ext4) или монтировании (mount -o), а не через параметры самого модуля ядра, который лишь предоставляет базовую логику.

Попробуйте выгрузить модуль ядра ext4, попробуйте выгрузить модуль ядра xfs(рис. 3.6).

```
1C:9D:0D:F7
[root@vvustinova ~]# modprobe -r ext4
modprobe: FATAL: Module crc32c_intel is in use.
[root@vvustinova ~]# modprobe -r ext4
[root@vvustinova ~]# modprobe -r xfs
modprobe: FATAL: Module xfs is in use.
```

Рис. 3.6: Не удалось выгрузить модуль, потому что он активно используется системой, что подтверждается ошибкой “is in use”.

Посмотрите, загружен ли модуль bluetooth, загрузите модуль ядра bluetooth, посмотрите список модулей ядра, отвечающих за работу с Bluetooth(рис. 3.7).

```
[root@vvustinova ~]# lsmod | grep bluetooth
[root@vvustinova ~]# modprobe bluetooth
[root@vvustinova ~]# lsmod | grep bluetooth
bluetooth          1114112  0
rfkill             40960  4 bluetooth
```

Рис. 3.7: Загружаем модуль блютуза и смотрим список модулей ядра отвечающих за работу с блютузом

Посмотрите информацию о модуле bluetooth:(рис. 3.8).

```
42:04:B9:4E:1D:9D:6C:3B:4B:80:C3:43:3F:5A:58:AE:DB:BB:E5:82:
D7:76:76:34:6C:F2:0F:69:38:03:DF:A4:89:A7:30:61:8D:D5:07:72:
2C:3E:9E:A3:35:EE:9B:8D:64:AD:07:07:DA:73:75:20:FE:44:28:BC:
7C:A3:26:F7:82:B8:DA:B1:0A:8B:87:F2:69:2D:F8:F5:7F:0C:2B:C2:
B3:40:73:8F:1D:48:EE:62:F7:08:43:33:A1:18:52:7E:F3:91:E1:49:
A7:61:FF:DA:FA:6C:F5:DD:B3:AB:B2:9D:AA:DB:58:41:CF:F8:38:0B:
EE:C1:85:7E:14:6C:AF:33:F8:40:B4:1B:BD:0B:8A:53:5D:6C:EE:84:
47:1E:DA:9F
parm: disable_esco:Disable eSCO connection creation (bool)
parm: disable_ertm:Disable enhanced retransmission mode (bool)
parm: enable_ecred:Enable enhanced credit flow control mode (bool)
[root@vvustinova ~]# modprobe -r bluetooth
```

Рис. 3.8: disable\_esco: Отключает расширенные аудио-соединения (eSCO). disable\_ertm: Отключает улучшенный режим повторной передачи данных (ERTM).enable\_ecred: Включает улучшенный контроль потока данных (ECRED) для Bluetooth Low Energy.

Посмотрите версию ядра, используемую в операционной системе, Выведите на экран список пакетов, относящихся к ядру операционной системы(рис. 3.9).

```
[root@vvustinova ~]# su -
[root@vvustinova ~]# uname -r
5.14.0-570.17.1.el9_6.x86_64
[root@vvustinova ~]# dnf list kernel
Rocky Linux 9 - BaseOS                8.7 kB/s | 4.1 kB    00:00
Rocky Linux 9 - BaseOS                2.2 MB/s | 2.5 MB    00:01
Rocky Linux 9 - AppStream             11 kB/s | 4.5 kB    00:00
Rocky Linux 9 - AppStream             1.1 MB/s | 9.5 MB    00:08
Rocky Linux 9 - Extras                6.9 kB/s | 2.9 kB    00:00
Установленные пакеты
kernel.x86_64                        5.14.0-570.17.1.el9_6    @anaconda
Имеющиеся пакеты
kernel.x86_64                        5.14.0-570.58.1.el9_6    baseos
[root@vvustinova ~]#
```

Рис. 3.9: Версия ядра и список пакетов

Обновите систему, чтобы убедиться, что все существующие пакеты обновлены, Обновите ядро операционной системы, а затем саму операционную систему(рис. 3.10).

```
[root@vvustinova ~]# dnf update kernel
Последняя проверка окончания срока действия метаданных: 0:20:30 назад, Пт 07 ноя 2025 11:23:10.
Зависимости разрешены.
Отсутствуют действия для выполнения.
Выполнено!
[root@vvustinova ~]# dnf update
Последняя проверка окончания срока действия метаданных: 0:20:43 назад, Пт 07 ноя 2025 11:23:10.
Зависимости разрешены.
Отсутствуют действия для выполнения.
Выполнено!
[root@vvustinova ~]# dnf upgrade --refresh
Rocky Linux 9 - BaseOS                7.9 kB/s | 4.1 kB    00:00
Rocky Linux 9 - AppStream             9.0 kB/s | 4.5 kB    00:00
Rocky Linux 9 - Extras                3.6 kB/s | 2.9 kB    00:00
Зависимости разрешены.
Отсутствуют действия для выполнения.
Выполнено!
[root@vvustinova ~]#
```

Рис. 3.10: Все уже обновлено

Посмотрите версию ядра, используемую в операционной системы(рис. 3.11).

```
[vvustinova@vvustinova ~]$ uname -r
5.14.0-570.58.1.el9_6.x86_64
[vvustinova@vvustinova ~]$ hostnamectl
  Static hostname: vvustinova.localdomain
        Icon name: computer-vm
        Chassis: vm
        Machine ID: 580d4f7e5f63408eb20cbd7b314c132a
        Boot ID: e8d6583cc779475b936d8e36de6360eb
  Virtualization: oracle
Operating System: Rocky Linux 9.6 (Blue Onyx)
        CPE OS Name: cpe:/o:rocky:rocky:9::baseos
        Kernel: Linux 5.14.0-570.58.1.el9_6.x86_64
  Architecture: x86-64
  Hardware Vendor: innotek GmbH
  Hardware Model: VirtualBox
Firmware Version: VirtualBox
[vvustinova@vvustinova ~]$
```

Рис. 3.11: После перезагрузки смотрим версию

## 4 Выводы

Мы успешно получили навыки работы с утилитами управления модулями ядра операционной системы.

## 5 Ответы на контрольные вопросы

1. Какая команда показывает текущую версию ядра, которая используется на вашей системе? `uname -r`
2. Как можно посмотреть более подробную информацию о текущей версии ядра операционной системы? `uname -a`
3. Какая команда показывает список загруженных модулей ядра? `lsmod`
4. Какая команда позволяет вам определять параметры модуля ядра? `modinfo` (параметры показываются в секции `parm:`)
5. Как выгрузить модуль ядра? `modprobe -r`
6. Что вы можете сделать, если получите сообщение об ошибке при попытке выгрузить модуль ядра?
  - Проверить, не используется ли модуль активно (например, файловая система).
  - Попытаться остановить сервисы или отмонтировать устройства, которые используют этот модуль.
  - Убедиться, что нет других модулей, зависящих от него.
7. Как определить, какие параметры модуля ядра поддерживаются? `modinfo` (информация о параметрах будет в строках, начинающихся с `parm:`)
8. Как установить новую версию ядра? Обычно это делается через менеджер пакетов вашей операционной системы (например, `sudo apt update && sudo`

apt upgrade для Debian/Ubuntu или sudo dnf update для Fedora/RHEL). Новые ядра поставляются как пакеты, и система автоматически их устанавливает и настраивает загрузчик.