

# **Отчёт по лабораторной работе №10**

**Основы работы с модулями ядра операционной системы**

Шаханеоядж Хаоладар

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Отчёт по выполнению работы</b>	<b>6</b>
2.1	Управление модулями ядра из командной строки . . . . .	6
2.2	Загрузка модулей ядра с параметрами . . . . .	10
2.3	Обновление ядра системы . . . . .	12
<b>3</b>	<b>Контрольные вопросы</b>	<b>15</b>
<b>4</b>	<b>Заключение</b>	<b>17</b>

## Список иллюстраций

2.1	Вывод команды <code>lspci -k</code> . . . . .	7
2.2	Просмотр загруженных модулей ядра . . . . .	8
2.3	Загрузка модуля <code>ext4</code> . . . . .	9
2.4	Попытка выгрузки модулей <code>ext4</code> и <code>xfs</code> . . . . .	10
2.5	Обновление ядра и системы . . . . .	12
2.6	Обновление ядра и системы . . . . .	13
2.7	Проверка версии ядра и сведений о системе . . . . .	14

## **Список таблиц**

# 1 Цель работы

Получить навыки работы с утилитами управления модулями ядра операционной системы.

## 2 Отчёт по выполнению работы

### 2.1 Управление модулями ядра из командной строки

1. В терминале были получены полномочия администратора с помощью команды `su -`.
2. Командой `lspci -k` был выведен список устройств и связанных с ними модулей ядра.

Среди них: контроллеры **VGA (VMware SVGA II Adapter)**, **Ethernet (Intel PRO/1000 MT)**, **аудио (Intel AC'97)**, **USB**, **SATA** и др.

Для каждого устройства указаны активные драйверы (`kernel driver in use`) и доступные модули (`kernel modules`).

```

haoladar@haoladar:~$ su
Password:
root@haoladar:/home/haoladar# 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@haoladar:/home/haoladar# █

```

Рис. 2.1: Вывод команды `lspci -k`

3. Список загруженных модулей ядра был просмотрен с помощью команды `lsmod | sort`.

Среди активных модулей присутствовали **ahci**, **ata\_piix**, **vmwgfx**, **snd\_intel8x0**, **e1000**, **isofs**, **loop** и другие.

```

root@haoladar:/home/haoladar#
root@haoladar:/home/haoladar# 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_msr
intel_rapl_msr          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
nf_nat                  69632  1 nft_chain_nat

```

Рис. 2.2: Просмотр загруженных модулей ядра

4. Проверка загрузки модуля **ext4** (`lsmod | grep ext4`) показала его отсутствие в памяти.

После выполнения команды `modprobe ext4` модуль успешно загрузился, что подтвердилось повторным вызовом `lsmod | grep ext4`.



```

root@haoladar:/home/haoladar#
root@haoladar:/home/haoladar# lsmod | grep ext4
root@haoladar:/home/haoladar# modprobe ext4
root@haoladar:/home/haoladar# lsmod | grep ext4
ext4                1187840    0
mbcache              16384      1 ext4
jbd2                 217088      1 ext4
root@haoladar:/home/haoladar#
root@haoladar:/home/haoladar# modprobe ext4
root@haoladar:/home/haoladar# 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:

```

Рис. 2.3: Загрузка модуля ext4

5. Команда `modinfo ext4` вывела подробную информацию о модуле: имя, версию, лицензию GPL, авторов, зависимости (jbd2, mbcache), а также путь к бинарному файлу `/lib/modules/.../fs/ext4/ext4.ko.xz`.

У модуля отсутствуют настраиваемые параметры, что указано в выводе.

6. При попытке выгрузить модуль **ext4** с помощью `modprobe -r ext4` система выдала сообщение об ошибке, так как модуль был задействован.

Аналогичная ситуация возникла при попытке выгрузить модуль **xfs** — команда завершилась ошибкой `FATAL: Module xfs is in use`.

```

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:
68: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:A6:9D: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:B2:F3:17:11:06:81:B2:F8:C6:23:85:76:76:43:9B:
00:78:01:A7:51:FA:AC:9F:25:39:88:1C:FF:27:C2:AF:0B:30:64:BA:
D6:FB:2C:81:43:AA:61:09:3E:6B:C8:5C:43:D2:B1:BB:8F:E4:99:95:
45:FE:EF:EC:B8:E0:3F:A1:50:E8:31:E2:8C:F2:5D:EF:15:87:2D:54:
98:24:93:D0:B7:A8:17:6B:E7:5A:23:6B:DB:9D:1C:CC:42:F1:5B:2B:
D5:48:EE:F5:A4:EA:FC:0E:D4:AD:8B:FB:A9:0B:53:DD:33:AD:DA:F4:
73:F5:28:0B:2B:19:84:75:BB:B6:D5:2E:34:3E:89:53:FB:FD:78:C6:
FB:83:F0:1D:75:8B:8C:96:8D:87:62:27:B0:58:99:A0:A0:23:E2:A7:
2B:E7:E7:25:2C:AB:D2:68:A2:C9:44:30:39:FF:46:82:CC:80:DF:CB:
3A:10:6D:7F:43:6F:35:78:A0:96:EA:BC:3C:31:E4:5F:15:DB:8E:C4:
01:8F:DB:E7:F9:0D:AF:80:8F:C2:59:13:7D:01:5E:78:18:AE:ED:F8:
FA:7A:C2:3E:9E:63:B4:57:50:92:28:E8:95:2D:D3:44:C5:8B:F5:C8:
0A:BE:F5:96:B5:D8:DE:8F:91:DA:85:E5:BF:A2:7D:9B:2A:4C:AD:4E:
07:54:52:A9:F2:62:B6:D7:58:9F:2E:8D:6A:C2:F4:8E:98:32:EB:14:
E6:BE:EE:3B:C5:98:36:95:53:D8:61:BD:41:8A:FF:4B:AC:49:70:B2:
19:94:AF:2A
root@haoladar:/home/haoladar#
root@haoladar:/home/haoladar# modprobe -r ext4
root@haoladar:/home/haoladar# modprobe -r ext4
root@haoladar:/home/haoladar# modprobe -r xfs
modprobe: FATAL: Module xfs is in use.
root@haoladar:/home/haoladar# modprobe -r xfs
modprobe: FATAL: Module xfs is in use.
root@haoladar:/home/haoladar# █

```

Рис. 2.4: Попытка выгрузки модулей ext4 и xfs

## 2.2 Загрузка модулей ядра с параметрами

1. Проверка наличия модуля **bluetooth** (`lsmod | grep bluetooth`) показала, что он не был загружен.

После выполнения команды `modprobe bluetooth` модуль появился в списке активных.

2. Информация о модуле, полученная через `modinfo bluetooth`, содержит сведения о версии (2.22), лицензии (GPL), авторе (**Marcel Holtmann**), зависимостях (`rfkill`), а также список параметров:

- `disable_esco` — отключение eSCO;
- `disable_ertm` — отключение улучшенного режима передачи;

- `enable_ecred` — включение улучшенного управления потоком.

### 3. После тестирования модуль был успешно выгружен командой `modprobe -r bluetooth`.

```
root@haoladar:/home/haoladar#
root@haoladar:/home/haoladar# lsmod | grep bluetooth
root@haoladar:/home/haoladar# modprobe bluetooth
root@haoladar:/home/haoladar# lsmod | grep bluetooth
bluetooth                1114112  0
rfkill                    40960  4 bluetooth
root@haoladar:/home/haoladar# modinfo 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:A1:34:94:0D: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:99:10:8D: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:CC:C2:A5:08:85:C9:34:D9:69:1C:28:BB:81:B0:8A:C7:
9F:AD:5D:46:AB:C1:B2:B1:C4:E3:8B:1A:12:5D:E2:52:D0:42:DD:FD:
2B:D0:5A:51:5C:6E:AD:0D:FD:90:EB:62:89:91:D2:C8:4A:25:EA:9D:
CA:B9:BE:27:C7:BB:85:D6:A5:A1:7B:8B:DE:E2:CC:26:F3:D6:D1:D2:
8C:FF:82:18:76:56:0C:DE:85:BB:B2:BD:8C:56:99:B6:21:B5:3A:ED:
15:E8:CC:3D:5F:C9:84:65:1C:94:DD:98:ED:03:D3:6D:DE:F3:B7:31:
19:34:07:1F:19:4A:82:C6:F5:82:F3:3C:87:38:F3:A7:69:66:0E:3B:
54:25:78:36:FC:A5:17:C0:5A:62:C5:4C:41:F6:FF:87:5F:0C:58:45:
```

```

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:    2E:24:0B:EE:9E:E6:A3:58:91:2A:D3:87:25:A3:1B:46:06:89:06:BC:
              4A:99:10:8D: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:CC:C2:A5:08:85:C9:34:D9:69:1C:28:BB:81:B0:8A:C7:
              9F:AD:5D:46:AB:C1:B2:B1:C4:E3:8B:1A:12:5D:E2:52:D0:42:DD:FD:
              2B:D0:5A:51:5C:6E:AD:0D:FD:90:EB:62:89:91:D2:C8:4A:25:EA:9D:
              CA:B9:BE:27:C7:BB:85:D6:A5:A1:7B:8B:DE:E2:CC:26:F3:D6:D1:D2:
              8C:FF:82:18:76:56:0C:DE:85:BB:B2:BD:8C:56:99:B6:21:B5:3A:ED:
              15:E8:CC:3D:5F:C9:84:65:1C:94:DD:98:ED:03:D3:6D:DE:F3:B7:31:
              19:34:07:1F:19:4A:82:C6:F5:82:F3:3C:87:38:F3:A7:69:66:0E:3B:
              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:F5:75:0B: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_ertm:Disable enhanced retransmission mode (bool)
parm:         enable_ecred:Enable enhanced credit flow control mode (bool)
root@haoladar:/home/haoladar#
root@haoladar:/home/haoladar#
root@haoladar:/home/haoladar# modprobe -r bluetooth
root@haoladar:/home/haoladar# modprobe -r bluetooth
root@haoladar:/home/haoladar# █

```

## 2.3 Обновление ядра системы

1. Проверка версии ядра командой `uname -r` показала использование версии **6.12.0-55.12.1.el10\_0.x86\_64**.
2. Список доступных пакетов ядра был выведен командой `dnf list kernel`. Обнаружено обновление до версии **6.12.0-55.37.1.el10\_0.x86\_64**.

```

root@haoladar:/home/haoladar# uname -r
6.12.0-55.12.1.el10_0.x86_64
root@haoladar:/home/haoladar# dnf list kernel
Rocky Linux 10 - BaseOS                               10 kB/s | 4.3 kB  00:00
Rocky Linux 10 - AppStream                             17 kB/s | 4.3 kB  00:00
Rocky Linux 10 - Extras                               5.0 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@haoladar:/home/haoladar# dnf upgrade --refresh

```

Рис. 2.5: Обновление ядра и системы

3. Для обновления системы были последовательно выполнены команды:

- `dnf upgrade --refresh`
- `dnf update kernel`
- `dnf update`

После установки пакетов обновлённое ядро было добавлено в систему.

```

sudo-1.9.15-8.p5.el10_0.2.x86_64
tiwilink-firmware-20250812-15.7.el10_0.noarch
tuned-ppd-2.25.1-2.el10_0.noarch
udisks2-tscsi-2.10.90-5.el10_0.1.x86_64
valgrind-1:3.24.0-5.el10_0.x86_64
which-2.21-44.el10_0.x86_64
xdg-user-dirs-0.18-6.el10_0.1.x86_64
yum-4.20.0-14.el10_0.rocky.0.1.noarch
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
sudo-python-plugin-1.9.15-8.p5.el10_0.2.x86_64
tuned-2.25.1-2.el10_0.noarch
udisks2-2.10.90-5.el10_0.1.x86_64
udisks2-lvm2-2.10.90-5.el10_0.1.x86_64
valgrind-devel-1:3.24.0-5.el10_0.x86_64
xdg-desktop-portal-1.20.0-1.el10_0.x86_64
xorg-x11-server-Xwayland-24.1.5-4.el10_0.x86_64
zlib-ng-compat-2.2.3-1.el10.rocky.0.1.x86_64
Complete!
root@haoladar:/home/haoladar# dnf update kernel
Last metadata expiration check: 0:01:17 ago on Fri 17 Oct 2025 02:25:51 PM MSK.
Dependencies resolved.
Nothing to do.
Complete!
root@haoladar:/home/haoladar# dnf update
Last metadata expiration check: 0:01:20 ago on Fri 17 Oct 2025 02:25:51 PM MSK.
Dependencies resolved.
Nothing to do.
Complete!
root@haoladar:/home/haoladar# dnf upgrade --refresh
Rocky Linux 10 - BaseOS                               12 kB/s | 4.3 kB  00:00
Rocky Linux 10 - AppStream                             14 kB/s | 4.3 kB  00:00
Rocky Linux 10 - Extras                               12 kB/s | 3.1 kB  00:00
Dependencies resolved.
Nothing to do.
Complete!
root@haoladar:/home/haoladar#

```

Рис. 2.6: Обновление ядра и системы

- После перезагрузки команда `uname -r` подтвердила использование нового ядра **6.12.0-55.37.1.el10\_0.x86\_64**.

Команда `hostnamectl` показала, что система работает под управлением **Rocky Linux 10.0 (Red Quartz)** с поддержкой до 2035 года.

```
haoladar@haoladar:~$ uname -r
6.12.0-55.37.1.el10_0.x86_64
haoladar@haoladar:~$ hostnamectl
  Static hostname: haoladar.localdomain
    Icon name: computer-vm
    Chassis: vm 🖥
  Machine ID: 680b0151ac144c679386de82018881d0
    Boot ID: ca47b22136e3420b817960b0c9a014f3
  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 1w 5d
    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 2d
haoladar@haoladar:~$
```

Рис. 2.7: Проверка версии ядра и сведений о системе

### 3 Контрольные вопросы

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

**uname -r** — отображает версию загруженного ядра Linux.

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

**hostnamectl** — выводит сведения о системе, включая версию ядра, дистрибутив и архитектуру.

3. Какая команда показывает список загруженных модулей ядра?

**lsmod** — выводит таблицу всех модулей, загруженных в память ядра.

4. Какая команда позволяет вам определять параметры модуля ядра?

**modinfo** — показывает информацию о модуле, включая параметры, которые можно задать при его загрузке.

5. Как выгрузить модуль ядра?

**modprobe -r** — удаляет модуль из памяти, если он не используется другими процессами или зависимостями.

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

Необходимо убедиться, что модуль не используется, проверить зависимости командой **lsmod**, а при необходимости завершить процессы, использующие данный модуль, или выгрузить зависимые модули.

7. Как определить, какие параметры модуля ядра поддерживаются?

**modinfo** — в выводе содержится раздел *parm*, где перечислены доступные параметры и их описание.

8. Как установить новую версию ядра?

Последовательно выполнить команды:

- **dnf upgrade --refresh** — обновить репозитории и существующие пакеты.
- **dnf update kernel** — установить обновлённую версию ядра.
- После установки перезагрузить систему и убедиться, что загружено новое ядро с помощью **uname -r**.



## 4 Заключение

Освоены основные приёмы управления модулями ядра и обновления версии ядра в операционной системе **Rocky Linux** с использованием командной строки.