

# Основы работы с модулями ядра операционной системы

## Часть 1

---

Славинский В.В.

8 ноября 2025

Российский университет дружбы народов, Москва, Россия Россия

## Информация

---

..... {.columns align=center} ::: {.column width="70%"}

- Славинский Владислав Вадимович
- Студент
- Российский университет дружбы народов
- [1132246169@pfur.ru]

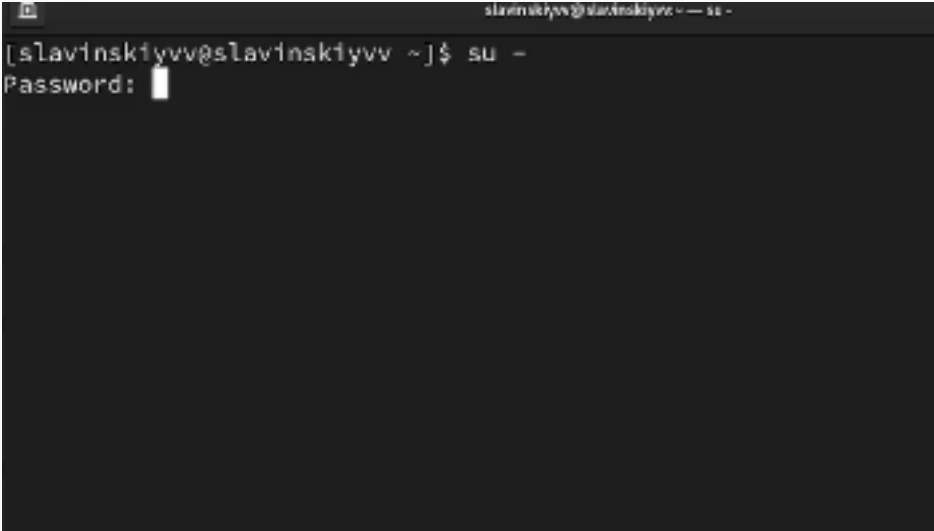
::: ::: {.column width="30%"}

## Вводная часть

---

## Получение root прав

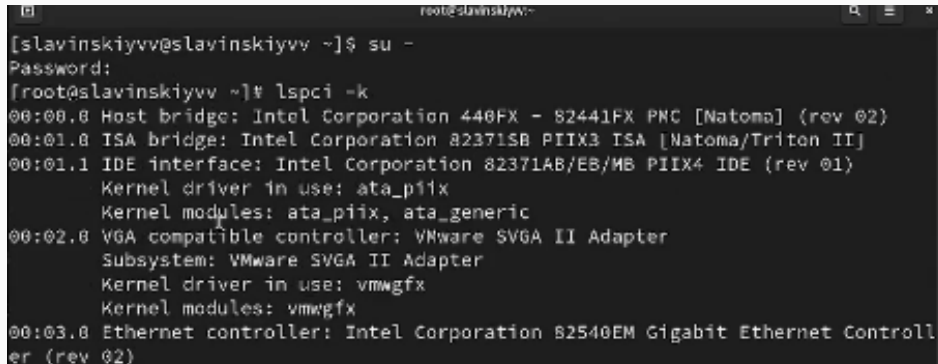
Запустим терминал и получим полномочия администратора: su -.

A terminal window with a dark background. The title bar at the top shows a window icon and the text 'slavinskiyvv@slavinskiyvv ~ - ss -'. The terminal content shows the prompt '[slavinskiyvv@slavinskiyvv ~]\$' followed by the command 'su -'. Below the command, the text 'Password:' is displayed, followed by a white cursor block indicating where the password is being entered.

```
slavinskiyvv@slavinskiyvv ~ - ss -  
[slavinskiyvv@slavinskiyvv ~]$ su -  
Password: 
```

## Устройства в системе

Посмотрим, какие устройства имеются в нашей системе и какие модули ядра с ними связаны: `lspci -k`. Эта команда выводит нам список устройств, которые подключены через шину PCI, а также модули ядра, которые используются для работы этих устройств. Например VGA compatible controller - это виртуальный видеоконтроллер VMware. А драйвер `vmwgfx` - это драйвер графического адаптера VMware для гостевых систем.

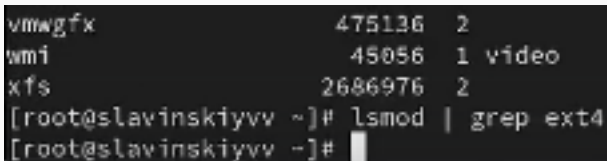


```
root@slavinskiyvv:~  
[slavinskiyvv@slavinskiyvv ~]$ su -  
Password:  
[root@slavinskiyvv ~]# lspci -k  
00:00.0 Host bridge: Intel Corporation 440FX - 82441FX PMC [Natoma] (rev 02)  
00:01.0 ISA bridge: Intel Corporation 82371SB PIIX3 ISA [Natoma/Triton II]  
00:01.1 IDE interface: Intel Corporation 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 B2540EM Gigabit Ethernet Controller (rev 02)
```

Посмотрим, какие модули ядра загружены: `lsmod | sort`.

```
Kernel driver in use: ahci
Kernel modules: ahci
[root@slavinskiyvv ~]# lsmod | sort
ac97_bus          12288  1 snd_ac97_codec
ahci              49152  3
ata_generic      16384  0
ata_piix         45056  1
cdrom            90112  2 iso9660fs,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  7 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
```

Посмотрим, загружен ли модуль ext4: `lsmod | grep ext4`. Модуль не загружен, поэтому нам нужно его загрузить.



```
vmwgfx          475136    2
wmi             45056    1 video
xfs            2686976    2
[root@slavinskiyv ~]# lsmod | grep ext4
[root@slavinskiyv ~]#
```

Рис. 4: sc4



Загрузим модуль ядра ext4: modprobe ext4. Проверяем, загрузился ли он.

```
xfs                2686976    2
[root@slavinskiyvv ~]# lsmod | grep ext4
[root@slavinskiyvv ~]# modprobe ext4
[root@slavinskiyvv ~]# lsmod | grep ext4
ext4                1191936    0
mbcache              16384    1 ext4
jbd2                 221184    1 ext4
[root@slavinskiyvv ~]#
```

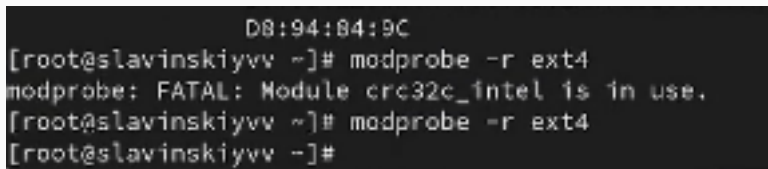
Рис. 5: sc5

## Просмотр информации о модуле ядра ext4

Далее посмотрим информацию о модуле ядра ext4: `modinfo ext4`. Команда выводит полную информацию о модуле ядра Linux. Модуль ядра ext4 обеспечивает поддержку одноименной файловой системы в Linux.

```
author:      Remy Card, Stephen Tweedie, Andrew Morton, Andreas Dilger, Theodor
             ro Ts'o and others
alias:       fs-ext4
alias:       ext3
alias:       fs-ext3
alias:       ext2
alias:       fs-ext2
rhelversion: 9.6
srcversion:  30C9EEDC227E3D26EE09D87
depends:      jbd2,mbcache
retpoline:   Y
intree:      Y
name:        ext4
vermagic:    5.14.0-570.39.1.el9_6.x86_64 SMP preempt mod_unload modversions
sig_id:      PKCS#7
signer:      Rocky kernel signing key
sig_key:     63:AD:10:02:33:B5:DE:70:E4:AB:58:C1:A4:DB:58:4B:31:65:42:9E
sig_hashalg: sha256
```

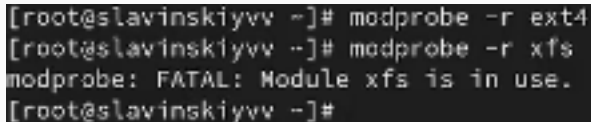
Выгрузим модуль ядра ext4: `modprobe -r ext4`. В первый раз нам не удалось выгрузить, потому что ext4 использует другой модуль ядра, а именно `crc32c_intel`. Во второй раз получилось, потому что модуль `crc20c_intel` больше не использовался напрямую системой, либо часть функционала ext4 была временно неактивна.



```
D8:94:84:9C
[root@slavinskiyvv ~]# modprobe -r ext4
modprobe: FATAL: Module crc32c_intel is in use.
[root@slavinskiyvv ~]# modprobe -r ext4
[root@slavinskiyvv ~]#
```

Рис. 7: sc7

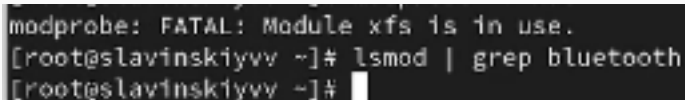
Потом попробуем выгрузить модуль ядра xfs: `modprobe -r xfs`. Здесь же мы не можем уже выгрузить модуль ядра xfs.

A terminal window with a black background and white text. The prompt is [root@slavinskiyv ~]#. The first command is modprobe -r ext4. The second command is modprobe -r xfs. The output is modprobe: FATAL: Module xfs is in use. The prompt returns to [root@slavinskiyv ~]#.

```
[root@slavinskiyv ~]# modprobe -r ext4
[root@slavinskiyv ~]# modprobe -r xfs
modprobe: FATAL: Module xfs is in use.
[root@slavinskiyv ~]#
```

Рис. 8: sc8

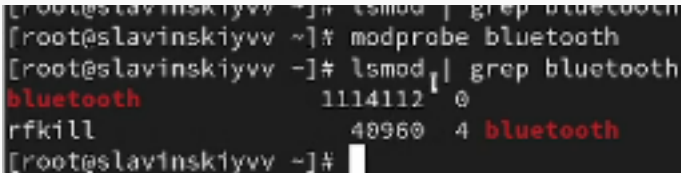
Дальше посмотрим, загружен ли модуль bluetooth: `lsmod | grep bluetooth`. Видим, что у нас не загружен.



```
modprobe: FATAL: Module xfs is in use.  
[root@slavinskiyvy ~]# lsmod | grep bluetooth  
[root@slavinskiyvy ~]#
```

Рис. 9: sc9

Загрузим модуль bluetooth: `modprobe bluetooth`. И дальше проверим список модулей ядра:  
`lsmod | grep bluetooth`



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

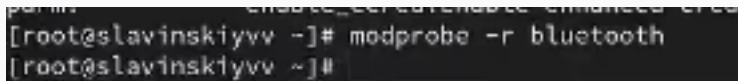
Рис. 10: sc10

## Информация о модуле bluetooth

Посмотрим информацию о модуле `bluetooth:modinfo bluetooth`. Команда вывела информацию о модуле ядра `bluetooth`. У нас есть такие параметры как: `disable eSCO support`, `disable retransmission mode`, `enable enhanced credit flow control mode`. Первый параметр - это отключение типа аудиоканала `bluetooth`. Второй параметр - это отключение расширенного режима повторной передачи, и третий параметр включает улучшенный режим управления потоком данных.

```
9D:85:EA:72:3D:A8:EC:41:1E:0C:0A:33:70:7A:70:3A:71:CB:E0:47:
70:78:12:FF:73:17:64:DA:F5:F1:68:EB:EB:E1:29:F8:78:C3:7F:C8:
33:B1:65:E3:85:2A:FF:1B:27:4F:C8:4F:62:39:D6:22:C9:FA:6F:34:
DE:87:5D:47:FF:D3:86:C7:F6:2D:A3:8F:EF:33:0C:10:5E:29:8F:88:
DD:53:EB:97:B8:1F:77:74:DF:1E:A6:FA:86:27:01:FC:56:CB:D9:E5:
C7:6C:C8:DC:B3:0F:E9:6E:61:6C:5C:F8:F7:C8:C8:2D:E7:AB:05:A1:
F5:C0:BB:CB:35:CE:23:11:B8:25:4C:2E:E8:A3:AB:DA:B2:D4:B0:E7:
63:F5:2D:AC:10:42:1C:86:AB:15:E7:51:4E:0E:11:3A:E0:CD:9C:9C:
43:16:8D:F3:BC:AE:3A:61:F6:D2:15:50:3C:C0:13:66:68:01:32:90:
5F:CA:DF:E9
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)
```

Выгрузим модуль bluetooth.

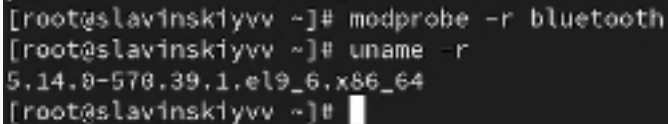
A terminal window with a black background and white text. The prompt is [root@slavinskiyv ~]#. The command modprobe -r bluetooth has been entered and executed. The output is not visible, but the prompt has moved to a new line.

```
par m: enable_...enable_...enable_...  
[root@slavinskiyv ~]# modprobe -r bluetooth  
[root@slavinskiyv ~]#
```

Рис. 12: sc12



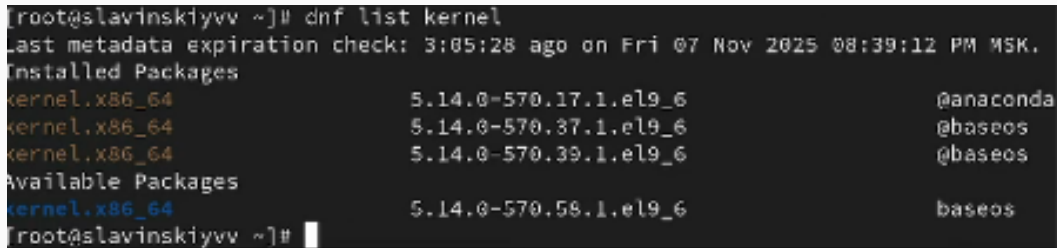
Дальше посмотрим версию ядра Linux, используемую в операционной системе: `uname -r`.



```
[root@slavinskiyvv ~]# modprobe -r bluetooth
[root@slavinskiyvv ~]# uname -r
5.14.0-570.39.1.el9_6.x86_64
[root@slavinskiyvv ~]#
```

Рис. 13: sc13

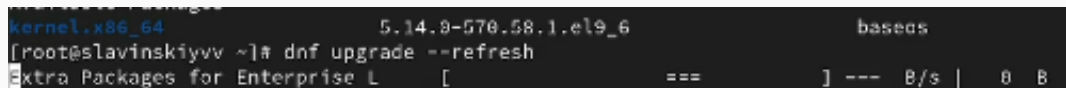
Выведем на экран список пакетов, относящихся к ядру операционной системы: `dnf list kernel`.



```
[root@slavinskiyvv ~]# dnf list kernel
Last metadata expiration check: 3:05:28 ago on Fri 07 Nov 2025 08:39:12 PM MSK.
Installed Packages
kernel.x86_64                    5.14.0-570.17.1.el9_6      @anaconda
kernel.x86_64                    5.14.0-570.37.1.el9_6      @baseos
kernel.x86_64                    5.14.0-570.39.1.el9_6      @baseos
Available Packages
kernel.x86_64                    5.14.0-570.58.1.el9_6      baseos
[root@slavinskiyvv ~]#
```

Рис. 14: sc14

Обновим систему, чтобы убедиться, что все существующие пакеты обновлены, чтобы избежать конфликтов: `dnf upgrade --refresh`.



```
kernel.x86_64 5.14.0-570.58.1.el9_6 baseos
[root@slavinskiyvv ~]# dnf upgrade --refresh
Extra Packages for Enterprise L [ === ] --- 0/s | 0 B
```

Рис. 15: sc15

Обновим ядро операционной системы, а затем саму операционную систему: `dnf update kernel`, `dnf update`, `dnf upgrade --refresh`.

```
Complete!
[root@slavinskiyvv ~]# dnf update kernel
Last metadata expiration check: 0:11:11 ago on Fri 07 Nov 2025 11:45:19 PM MSK.
Dependencies resolved.
Nothing to do.
Complete!
[root@slavinskiyvv ~]# dnf update
Last metadata expiration check: 0:11:18 ago on Fri 07 Nov 2025 11:45:19 PM MSK.
Dependencies resolved.
Nothing to do.
Complete!
[root@slavinskiyvv ~]# dnf upgrade --refresh
Extra Packages for Enterprise Linux 9 - x86_64                48 kB/s | 38 kB      00:00
Extra Packages for Enterprise Linux 9 openh264 (From Cisco) - x86_64 6.3 kB/s | 993 B     00:00
Rocky Linux 9 - BaseOS                                         17 kB/s | 4.1 kB     00:00
Rocky Linux 9 - AppStream                                       13 kB/s | 4.5 kB     00:00
Rocky Linux 9 - Extras                                         14 kB/s | 2.9 kB     00:00
Dependencies resolved.
Nothing to do.
Complete!
```

## Выбор нового ядра

Перезагружаем систему, выбираем новое ядро и проверяем изменения: `uname -r`, `hostnamectl`. И как видим, версия ядра изменилась.

```
[slavinskiyv@slavinskiyv ~]$ uname -r
5.14.0-570.58.1.el9_6.x86_64
[slavinskiyv@slavinskiyv ~]$ hostnamectl
  Static hostname: slavinskiyv
        Icon name: computer-vm
        Chassis: vm RT
        Machine ID: bd69652ae92748ae5b335d52b03b91df
        Boot ID: 85e6ba4786bc4f2493a82765b9ebbd15
  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
[slavinskiyv@slavinskiyv ~]$
```