

Отчет по выполнению лабораторной работы

Лабораторная работа №10

Колонтырский Илья Русланович

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
3	Ответы на контрольные вопросы	15
4	Выводы	16

Список иллюстраций

2.1	Переход в режим суперпользователя и просмотр устройств и модулей ядра, с ними связанных	6
2.2	Просмотр загруженных модулей ядра. Часть 1	7
2.3	Просмотр загруженных модулей ядра. Часть 2	8
2.4	Проверка и установка модуля ядра ext4	9
2.5	Просмотр информации о модуле ядра ext4	9
2.6	Выгрузка модуля	10
2.7	Попытка выгрузить модуль	11
2.8	Проверка и загрузка модуля bluetooth	11
2.9	Просмотр информации о модуле bluetooth	12
2.10	Выгрузка модуля bluetooth	12
2.11	Просмотр версии ядра	13
2.12	Вывод на экран список пакетов, относящихся к ядру	13
2.13	Обновление системы	13
2.14	Обновление ядра	13
2.15	Обновление системы	14
2.16	Просмотр версии ядра	14
2.17	Команда hostnamectl	14

List of Tables

1 Цель работы

Получить навыки работы с утилитами управления модулями ядра операционной системы.

2 Выполнение лабораторной работы

Задание 1. Получите полномочия администратора. Посмотрите, какие устройства имеются в вашей системе и какие модули ядра с ними связаны.

Я перешел в режим работы суперпользователя, использовав команду `su -`. А затем использовала команду `lspci -k` (рис. 2.1).

```

Password:
[root@pvbarabash ~]# 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
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
```

Рис. 2.1: Переход в режим суперпользователя и просмотр устройств и модулей ядра, с ними связанных

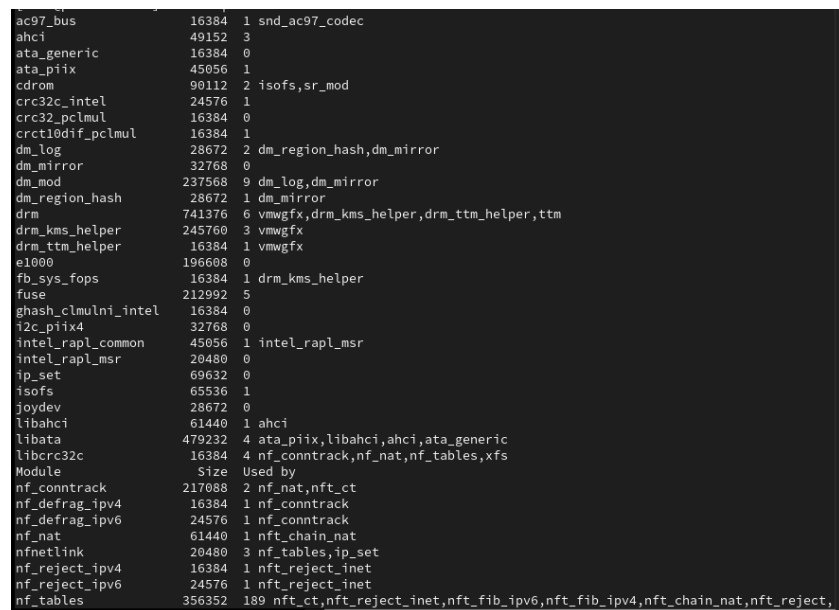
Вывод команды включает следующие данные:

1. Идентификатор устройства: Уникальный код, который идентифицирует устройство.

2. Имя устройства: Человекочитаемое название устройства.
3. Драйвер: Информация о том, какой драйвер используется для управления устройством. Если драйвер не загружен, это также будет указано.

Задание 2. Посмотрите, какие модули ядра загружены.

Я ввел команду `lsmod | sort` (рис. 2.2) и (рис. 2.3).



ac97_bus	16384	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	16384	0
crct10dif_pclmul	16384	1
dm_log	28672	2 dm_region_hash,dm_mirror
dm_mirror	32768	0
dm_mod	237568	9 dm_log,dm_mirror
dm_region_hash	28672	1 dm_mirror
drm	741376	6 vmwgfx,drm_kms_helper,drm_ttm_helper,ttm
drm_kms_helper	245760	3 vmwgfx
drm_ttm_helper	16384	1 vmwgfx
e1000	196608	0
fb_sys_fops	16384	1 drm_kms_helper
fuse	212992	5
ghash_clmulni_intel	16384	0
i2c_piix4	32768	0
intel_rapl_common	45056	1 intel_rapl_msr
intel_rapl_msr	20480	0
ip_set	69632	0
isofs	65536	1
joydev	28672	0
libahci	61440	1 ahci
libata	479232	4 ata_piix,libahci,ahci,ata_generic
libcrc32c	16384	4 nf_contrack,nf_nat,nf_tables,xfs
Module	Size	Used by
nf_contrack	217088	2 nf_nat,nft_ct
nf_defrag_ipv4	16384	1 nf_contrack
nf_defrag_ipv6	24576	1 nf_contrack
nf_nat	61440	1 nft_chain_nat
nfnetlink	20480	3 nf_tables,ip_set
nf_reject_ipv4	16384	1 nft_reject_inet
nf_reject_ipv6	24576	1 nft_reject_inet
nf_tables	356352	189 nft_ct,nft_reject_inet,nft_fib_ipv6,nft_fib_ipv4,nft_chain_nat,nft_reject,

Рис. 2.2: Просмотр загруженных модулей ядра. Часть 1

nf_tables	356352	189	nft_ct,nft_reject_inet,nft_fib_ipv6,nft_fib_ipv4,nft_chain_nat,nft_reject,
nft_fib,nft_fib_inet			
nft_chain_nat	16384	3	
nft_ct	24576	7	
nft_fib	16384	3	nft_fib_ipv6,nft_fib_ipv4,nft_fib_inet
nft_fib_inet	16384	1	
nft_fib_ipv4	16384	1	nft_fib_inet
nft_fib_ipv6	16384	1	nft_fib_inet
nft_reject	16384	1	nft_reject_inet
nft_reject_inet	16384	6	
nls_utf8	16384	1	
pcspkr	16384	0	
qrtr	57344	4	
rfkill	40960	3	
sd_mod	90112	3	
serio_raw	20480	0	
sg	53248	0	
snd	147456	12	snd_seq,snd_seq_device,snd_intel8x0,snd_timer,snd_ac97_codec,snd_pcm
snd_ac97_codec	200704	1	snd_intel8x0
snd_hrtimer	16384	1	
snd_intel8x0	57344	2	
snd_pcm	184320	2	snd_intel8x0,snd_ac97_codec
snd_seq	131072	7	snd_seq_dummy
snd_seq_device	16384	1	snd_seq
snd_seq_dummy	16384	0	
snd_timer	53248	3	snd_seq,snd_hrtimer,snd_pcm
soundcore	16384	1	snd
sr_mod	28672	1	
syscopyarea	16384	1	drm_kms_helper
sysfillrect	16384	1	drm_kms_helper
sysimgblt	16384	1	drm_kms_helper
tl0_pi	24576	1	sd_mod
tls	159744	0	
ttm	98304	2	vmwgfx,drm_ttm_helper
uinput	24576	0	
vboxguest	491520	0	
video	73728	0	
vmwgfx	450560	1	
wmi	45056	1	video
xfs	2510848	2	

Рис. 2.3: Просмотр загруженных модулей ядра. Часть 2

Задание 3. Посмотрите, загружен ли модуль ext4. Загрузите модуль ядра ext4. Убедитесь, что модуль загружен, посмотрев список загруженных модулей.

Сначала я проверил, загружен ли модуль ext4 с помощью команды `lsmod | grep ext4`. Он не был загружен, поэтому я его загрузил с помощью команды `modprobe ext4`. Затем я снова использовал команду `lsmod | grep ext4`, чтобы убедиться, что модуль загружен (рис. 2.4).


```
[ ] # lsmod | grep ext4
[ ] # modprobe ext4
[ ] # lsmod | grep ext4
      1183744    0
          16384    1 ext4
          217088    1 ext4
[ ] #
```

Рис. 2.4: Проверка и установка модуля ядра ext4

Задание 4. Посмотрите информацию о модуле ядра ext4.

Я использовал команду `modinfo ext4` (рис. 2.5).

```
modinfo: version 1.1 module: ext4
filename: /lib/modules/5.14.0-427.13.1.el9_4.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: 9.4
srcversion: 2B896FAB53D489F1C7683E6
depends: mbcache,jbd2
retpoline: Y
intree: Y
name: ext4
vermagic: 5.14.0-427.13.1.el9_4.x86_64 SMP preempt mod_unload modversions
sig_id: PKCS#7
signer: Rocky kernel signing key
sig_key: 40:50:2E:3B:AE:2E:C1:1E:1C:80:04:11:83:6A:66:EF:D3:5F:BC:63
sig_hashalgo: sha256
signature: 7B:B9:A1:A0:26:39:88:C1:F9:E6:97:16:D7:F2:DC:A1:C0:EA:8C:2C:
67:0B:03:48:32:7C:F9:1F:03:3D:ED:28:21:68:EC:A7:4A:C8:53:36:
DF:70:6B:3A:12:D0:CF:99:5C:1A:3B:16:2E:2C:E8:16:25:6C:61:C4:
7D:57:84:2A:62:83:54:2A:55:82:87:93:8F:05:5B:08:BB:D1:EE:1F:
78:BD:73:35:96:97:A8:4B:DF:CE:0B:01:CE:B3:41:94:15:92:A9:7E:
CB:20:63:A7:21:6F:44:48:F5:1D:2F:FE:76:17:C4:9A:F2:DA:23:F2:
31:84:F8:DC:FD:CA:25:F4:F8:F9:F2:13:53:61:C4:E0:40:8E:0F:A0:
B5:94:F1:01:80:08:C0:5F:3A:C4:45:2B:C3:83:6B:0D:72:AB:18:2A:
CD:E0:80:C3:0D:80:3F:AB:EF:1E:13:D5:3D:21:41:0C:C8:5D:61:19:
DF:34:4D:D4:E5:D2:3C:46:15:29:9F:8A:D4:57:60:63:57:CB:D1:2B:
72:B3:72:AF:AB:2D:33:88:CE:92:4C:59:42:8C:4E:A8:29:1D:D7:C2:
81:26:6F:AF:7C:E2:AA:5F:B6:FA:D5:0A:2A:1D:B1:8F:FB:18:A0:ED:
1B:EE:D6:54:33:1A:0B:D7:A4:37:23:D6:A7:A9:6C:D3:EF:42:E4:F2:
CA:E8:C3:B4:39:AC:FE:F5:4B:8D:B6:E9:DC:A8:91:C4:3B:D8:E7:10:
DC:3F:DC:6A:6B:03:23:BD:16:4C:67:59:FC:71:09:89:15:26:27:C2:
C5:EA:4C:48:60:ED:C3:D8:15:60:D4:91:5B:60:26:01:51:E4:F2:53:
```

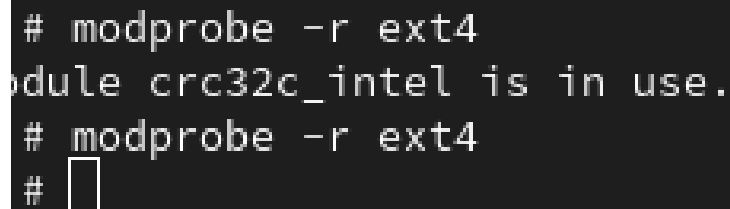
Рис. 2.5: Просмотр информации о модуле ядра ext4

Команда `modinfo` выдает следующую информацию:

1. filename: Путь к файлу модуля.
2. version: Версия модуля.
3. description: Краткое описание модуля.
4. author: Имя автора модуля.
5. license: Лицензия, под которой распространяется модуль.
6. srcversion: Исходная версия модуля, если доступна.
7. depends: Список зависимостей модуля (другие модули, которые должны быть загружены для работы данного модуля).
8. parm: Параметры, которые могут быть переданы модулю при его загрузке.

Задание 5. Попробуйте выгрузить модуль ядра ext4.

Чтобы выгрузить модуль ядра ext4, мне потребовалось дважды использовать команду `modprobe -r ext4` (рис. 2.6).



```
# modprobe -r ext4
module crc32c_intel is in use.
# modprobe -r ext4
#
```

Рис. 2.6: Выгрузка модуля

В первый раз выдалась ошибка “modprobe: FATAL: Module crc32c_intel is in use.”.

Задание 6. Попробуйте выгрузить модуль ядра xfs.

Я использовал команду `modprobe -r xfs` несколько раз, но каждый раз получал сообщение, что этот модуль сейчас используется (рис. 2.7).

```
[~]# modprobe -r xfs
module xfs is in use.
[~]# modprobe -r xfs
module xfs is in use.
[~]# modprobe -r xfs
module xfs is in use.
[~]#
```

Рис. 2.7: Попытка выгрузить модуль

Задание 7. Посмотрите, загружен ли модуль bluetooth. Загрузите модуль ядра bluetooth. Посмотрите список модулей ядра, отвечающих за работу с Bluetooth.

Я использовал команду `lsmod | grep bluetooth`, он был не загружен. Я загрузил его с помощью команды `modprobe bluetooth`, а затем снова использовал команду `lsmod | grep bluetooth` (рис. 2.8).

```
# lsmod | grep bluetooth
# modprobe bluetooth
# lsmod | grep bluetooth
1101824 0
40960 4 bluetooth
#
```

Рис. 2.8: Проверка и загрузка модуля bluetooth

Задание 8. Посмотрите информацию о модуле bluetooth.

Я использовал команду `modinfo bluetooth` (рис. 2.9).

```
~]# modinfo bluetooth
/lib/modules/5.14.0-427.42.1.el9_4.x86_64/kernel/net/bluetooth/bluetooth.ko.xz
net-pf-31
GPL
2.22
Bluetooth Core ver 2.22
Marcel Holtmann <marcel@holtmann.org>
9.4
4944688C29B6065B954BEA5
rfkill
Y
Y
bluetooth
5.14.0-427.42.1.el9_4.x86_64 SMP preempt mod_unload modversions
PKCS#7
Rocky kernel signing key
65:C7:3E:D7:79:A3:BA:93:1D:EE:C5:5C:9F:93:5D:EB:39:8D:43:59
sha256
9C:D8:30:14:83:46:67:C6:E7:A4:54:01:E1:99:76:A1:12:61:D3:D4:
71:9D:7E:22:45:D9:8B:0D:BC:BF:91:33:86:78:6C:AC:07:B2:43:7D:
22:2A:66:F1:B2:89:D0:10:57:FD:36:0F:D4:13:10:53:D0:1D:F9:C0:
3D:0A:78:11:C3:85:AF:39:31:FA:A1:62:06:50:FF:8C:A1:77:D7:60:
95:51:19:EF:D3:6E:0A:79:66:B7:FE:B7:A7:A4:16:F0:E9:D7:08:3C:
C4:21:85:52:EE:66:45:DD:95:9F:FC:6D:18:84:39:71:8E:28:DB:A3:
13:39:A8:9B:C0:05:53:25:67:37:7B:3C:95:E4:EA:1C:64:BD:9A:0E:
E3:BF:75:C2:5A:7E:28:6D:3A:72:CA:8A:60:5F:6D:AA:3C:8E:6C:82:
78:A8:DF:06:C9:31:CE:C3:92:29:90:DF:9F:18:A8:BE:B6:40:6C:05:
0D:78:78:CC:54:31:AF:F9:A9:19:2D:FA:02:4C:83:54:5F:BB:48:ED:
6E:6C:D5:85:D2:D9:AD:49:D8:7C:F0:FE:2C:00:7B:B3:D6:8A:7D:D4:
99:BC:17:C7:B6:4C:1E:28:38:DF:BC:D3:76:F3:62:94:CD:10:56:AE:
B9:D3:4A:EF:77:16:45:C4:A0:20:A2:62:AF:70:9D:62:2F:33:0E:46:
82:DC:51:29:BD:FE:4C:E9:EC:2E:C7:6B:2E:86:03:F5:82:16:69:E3:
B4:FC:6A:36:5A:63:24:BA:79:2B:E1:3A:2C:24:D3:20:B5:91:B6:92:
77:45:8F:AA:DE:2B:AD:14:9E:31:A2:C6:A6:34:9C:9D:6A:18:07:22:
BA:80:BB:14:3C:25:64:6C:55:5A:2D:42:98:49:42:A2:AA:25:49:DA:
34:29:C8:F8:77:51:D2:50:2D:27:B1:62:9B:C4:3F:88:A2:A4:D2:DA:
E2:D5:B4:92:9E:46:B9:32:28:83:0A:8D:28:5B:2B:E0:F0:7B:9D:BF:
E4:F6:25:30
disable_esco:Disable eSCO connection creation (bool)
disable_ertm:Disable enhanced retransmission mode (bool)
enable_ecred:Enable enhanced credit flow control mode (bool)
~]#
```

Рис. 2.9: Просмотр информации о модуле bluetooth

Этот модуль имеет следующие параметры: `disable_esco`, `disable_ertm`, `enable_ecred`.

Задание 9. Выгрузите модуль ядра bluetooth

Я использовал команду `modprobe -r bluetooth` и не столкнулся ни с какими ошибками (рис. 2.10).

```
~]# modprobe -r bluetooth
~]#
```

Рис. 2.10: Выгрузка модуля bluetooth

Задание 10. Посмотрите версию ядра, используемую в операционной системе. Я использовал команду `uname -r` (рис. 2.11).

```
5.14.0-427.13.1.el9_4.x86_64
```

Рис. 2.11: Просмотр версии ядра

Задание 11. Выведите на экран список пакетов, относящихся к ядру операционной системы.

Я использовал команду `dnf list kernel` (рис. 2.12).

```
Rocky Linux 9 - BaseOS              7.2 kB/s | 4.1 kB  00:00
Rocky Linux 9 - AppStream           8.5 kB/s | 4.5 kB  00:00
Rocky Linux 9 - Extras              6.4 kB/s | 2.9 kB  00:00
Installed Packages
kernel.x86_64                      5.14.0-427.13.1.el9_4      @anaconda
Available Packages
kernel.x86_64                      5.14.0-427.42.1.el9_4      baseos
```

Рис. 2.12: Вывод на экран список пакетов, относящихся к ядру

Задание 12. Обновите систему, чтобы убедиться, что все существующие пакеты обновлены.

Я использовал команду `dnf upgrade --refresh` и дождалась, когда все пакеты обновились (рис. 2.13).

```
Installed:
composefs-1.0.3-2.el9.x86_64          composefs-libs-1.0.3-2.el9.x86_64
grub2-tools-efi-1:2.06-82.el9_4.x86_64 grub2-tools-extra-1:2.06-82.el9_4.x86_64
kernel-5.14.0-427.42.1.el9_4.x86_64    kernel-core-5.14.0-427.42.1.el9_4.x86_64
kernel-devel-5.14.0-427.42.1.el9_4.x86_64 kernel-modules-5.14.0-427.42.1.el9_4.x86_64
kernel-modules-core-5.14.0-427.42.1.el9_4.x86_64
Complete!
```

Рис. 2.13: Обновление системы

Задание 13. Обновите ядро операционной системы, а затем саму операционную систему.

Я выполнил команду `dnf update kernel` и `dnf update` (рис. 2.14).

```
Last metadata expiration check: 0:09:07 ago on Sun 03 Nov 2024 05:37:20 PM MSK.
Dependencies resolved.
Nothing to do.
Complete!
```

Рис. 2.14: Обновление ядра

Затем я обновил систему с помощью команды `dnf upgrade –refresh` (рис. 2.15).

```
Rocky Linux 9 - BaseOS                               6.7 kB/s | 4.1 kB   00:00
Rocky Linux 9 - AppStream                             9.7 kB/s | 4.5 kB   00:00
Rocky Linux 9 - Extras                               5.7 kB/s | 2.9 kB   00:00
Dependencies resolved.
Nothing to do.
Complete!
```

Рис. 2.15: Обновление системы

Задание 14. Перезагрузите систему.

Я перезагрузил систему с помощью команды `reboot`.

Задание 15. Посмотрите версию ядра, используемую в операционной системе.

Я посмотрел версию ядра с помощью команды `uname -r` (рис. 2.16).

```
5.14.0-427.42.1.el9_4.x86_64
```

Рис. 2.16: Просмотр версии ядра

Как можно видеть, версия изменилась.

Затем я ввел команду `hostnamectl` (рис. 2.17).

```
5.14.0-427.42.1.el9_4.x86_64
[pvbarabash@pvbarabash ~]$ hostnamectl
  Static hostname: pvbarabash.localdomain
        Icon name: computer-vm
        Chassis: vm
        Machine ID: 4158dfe3c77d4dbda233ff2b4ec22fbe
        Boot ID: 8c608866f9bd49fda8f598366a382870
  Virtualization: oracle
Operating System: Rocky Linux 9.4 (Blue Onyx)
        CPE OS Name: cpe:/o:rocky:rocky:9::baseos
        Kernel: Linux 5.14.0-427.42.1.el9_4.x86_64
  Architecture: x86-64
  Hardware Vendor: innotek GmbH
  Hardware Model: VirtualBox
  Firmware Version: VirtualBox
```

Рис. 2.17: Команда `hostnamectl`

3 Ответы на контрольные вопросы

1. Какая команда показывает текущую версию ядра, которая используется на вашей системе? Команда: `uname -r`.
2. Как можно посмотреть более подробную информацию о текущей версии ядра операционной системы? Команда: `uname -a`.
3. Какая команда показывает список загруженных модулей ядра? Команда: `lsmod`.
4. Какая команда позволяет вам определять параметры модуля ядра? Команда: `modinfo` .
5. Как выгрузить модуль ядра? `modprobe -r` .
6. Что вы можете сделать, если получите сообщение об ошибке при попытке выгрузить модуль ядра? Убедиться, что модуль не используется другими процессами или модулями. Для этого можно использовать команду `lsmod`, чтобы проверить зависимости. Если модуль занят, попробовать завершить процессы, которые его используют, или перезагрузить систему.
7. Как определить, какие параметры модуля ядра поддерживаются? Команда: `modinfo` .
8. Как установить новую версию ядра? `sudo dnf install kernel-`

4 Выводы

Я получил навыки работы с утилитами управления модулями ядра операционной системы.