

Структура научной презентации

Простейший шаблон

Казначеев С. И.

01 ноября 2025

Российский университет дружбы народов

Информация

- Казначеев Сергей Ильич
- Студент
- Российский университет дружбы народов

Цель работы

Получить навыки работы с утилитами управления модулями ядра операционной системы.

После запуска терминала перейдем в супер пользователя

```
[sikaznacheev@localhost ~]$ su -  
Пароль:
```

Рис. 1: 1

Просмотр какие устройства имеются в нашей системе и модуля ядра

Далее посмотрим какие устройства имеются в нашей системе и модули ядра с ними связаны

Команда `lspci -k` отображает список всех PCI-устройств в системе, а также информацию о том, какие драйверы ядра и модули с ними связаны. Это позволяет проанализировать как операционная система взаимодействует с “железом”

```
[root@localhost ~]# lspci -k
00:00.0 Host bridge: Intel Corporation 440FX - 82441FX PNC [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
```

Просмотр модулей ядра

После чего просмотрим какие модули ядра загружены

```
[root@localhost ~]# 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
cfg80211          1359872  0
crc32c_intel       24576  1
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_log,dm_mirror
dm_region_hash      28672  1 dm_mirror
drm                 815104  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
ghash_clmulni_intel 16384  0
i2c_piix4          28672  0
intel_pmc_core     122880  0
intel_rapl_common   57344  1 intel_rapl_ms
intel_rapl_ms       20480  0
intel_uncore_frequency_common 16384  0
intel_vsec          20480  1 intel_pmc_core
ip_set              69632  0
isofs               65536  1
joydev              28672  0
libahci              61440  1 ahci
libata              528192  4 ata_piix,libahci,ahci,ata_generic
libcrc32c          12288  4 nf_conntrack,nf_nat,nf_tables,xfs
Module             Size Used by
nf_conntrack        229376  2 nf_nat,nft_ct
nf_defrag_ipv4      12288  1 nf_conntrack
nf_defrag_ipv6      24576  1 nf_conntrack
nf_nat              65536  1 nft_chain_nat
```

Просмотр загруженных модулей

Далее просмотрим загружен ли модуль ext4, затем загрузим его и убедимся что модуль загружен,просмотрев список загруженных модулей

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

Рис. 4: 4

Просмотр информации о модуле ext4

После чего просмотрим инфомацию о модуле ядра ext4

Команда modinfo отображает информацию о модуле ядра Linux. В данном случае мы исследуем модуль ext4, который отвечает за поддержку одноименной файловой системы

```
[root@localhost ~]# modinfo ext4
filename:      /lib/modules/5.14.0-570.55.1.el9_6.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:   9.6
srcversion:    30C9EEDC227E3D26EE09D87
depends:       jbd2,mbcache
retpoline:     Y
intree:        Y
name:          ext4
vermagic:     5.14.0-570.55.1.el9_6.x86_64 SMP preempt mod_unload modversions
sig_id:        PKCS#7
signer:        Rocky kernel signing key
sig_key:       3B:C7:17:F1:CA:27:25:D0:CC:5E:DC:A8:E8:DF:63:6A:EB:0F:99:AD
sig_hashalgo:  sha256
signature:    A2:60:1D:98:91:A0:27:1B:28:83:35:63:7F:EC:23:B2:2C:B5:1E:F5:
D6:89:0E:88:47:36:91:78:D4:87:00:3E:E5:3B:DF:ED:FA:50:46:
FA:39:AC:87:F3:EC:51:CF:CE:93:D4:3D:F1:55:85:4C:5E:47:38:33:
24:06:88:03:C1:D0:5D:44:73:99:44:8D:A2:88:FD:09:E7:5D:58:42:
70:AA:2C:21:64:CC:11:63:DF:BC:0B:0A:0B:75:D8:2F:0B:A7:B9:79:
A9:4D:7D:CA:07:7A:80:50:28:66:17:8B:DA:A8:AA:D0:CE:87:42:1C:
9C:F2:90:5D:CB:AE:AD:A5:8C:90:33:18:1F:16:87:E4:76:34:62:54:
C2:D0:C3:AD:34:59:E7:71:33:46:F6:08:43:37:37:80:B1:AD:30:EC:
```

Попытка выгрузить модуль ядра ext4

После пробуем выгрузить модуль ядра ext4, мы увидим что нам система сообщает что не может выгрузить ext4 потому что от него зависит другой модуль crc32c_intel который в данный момент используется. Ядро Linux не позволяет выгрузить модуль от которого зависят другие активные модули, чтобы избежать сбоев. Затем пробуем выгрузить модуль ядра xfs

```
[root@localhost ~]# modprobe -r ext4
modprobe: FATAL: Module crc32c_intel is in use.
[root@localhost ~]# modprobe -r ext4
[root@localhost ~]# modprobe -r xfs
modprobe: FATAL: Module xfs is in use.
```

Рис. 6: 6

Просмотр загружен ли модуль bluetooth

Далее просмотрим загружен ли модуль bluetooth увидим что нет и загрузим его,затем просмотрим список модулей ядра , отвечающий за работу Bluetooth

```
[root@localhost ~]# modprobe bluetooth
[root@localhost ~]# lsmod | grep bluetooth
bluetooth            1114112  0
rfkill                40960   5 bluetooth,cfg80211
[root@localhost ~]#
```

Рис. 7: 7

Просмотр информации о модуле bluetooth

После чего просмотрим информацию о модуле bluetooth

Команда отображает информацию о модуле ядра, отвечающем за работу Bluetooth-стека в системе. В выводе команды содержатся параметры, которые могут быть установлены для настройки работы этого модуля

```
[root@localhost ~]# modinfo bluetooth
filename:      /lib/modules/5.14.0-570.55.1.el9_6.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:   9.6
srcversion:    C6ED64B96F8C9A31FC91B17
depends:       rfkill
retpoline:     Y
intree:        Y
name:          bluetooth
vermagic:     5.14.0-570.55.1.el9_6.x86_64 SMP preempt mod_unload modversions
sig_id:        PKCS#7
signer:        Rocky kernel signing key
sig_key:       38:C7:17:F1:CA:27:25:D0:CC:5E:DC:A8:E8:DF:63:6A:EB:0F:99:AD
sig_hashalgo:  sha256
signature:    01:AF:AC:51:D7:84:99:60:4D:71:92:C1:DE:8E:39:59:50:13:91:75:
              6E:9E:0B:5A:FF:47:CC:E9:80:85:58:9A:3A:B9:91:D9:D8:0E:32:FD:
              BA:15:48:51:13:44:3C:19:CF:C6:96:DD:D1:16:F8:38:7F:AF:08:51:
              50:19:FB:67:0F:39:E3:CE:DC:ED:90:B2:04:90:F2:5E:A0:24:10:C3:
              CF:F3:17:26:E2:C6:78:3A:D8:C4:00:64:18:BF:7A:69:87:2F:69:CF:
              46:CC:FE:44:A8:31:BE:4D:58:74:17:33:9B:5F:CD:0C:43:56:FF:5B:
              85:E1:32:0C:3E:73:29:F2:C2:83:77:A2:72:C7:BE:94:72:FA:88:CF:
              F1:74:B3:C1:89:21:54:A6:40:91:A2:E3:75:5E:6C:55:90:1D:EC:29:
```

Выгружаем модуль ядра bluetooth

И выгрузим модуль ядра bluetooth

```
[root@localhost ~]# modprobe -r bluetooth
```

Рис. 9: 9

Проверка ядра в используемой операционной системе

После чего перейдем в супер пользователя, просмотрим версию ядра используемую в операционной системе и выведем на экран список пакетов относящихся к ядру операционной системы

```
[root@localhost ~]# su -
[root@localhost ~]# uname -r
5.14.0-570.55.1.el9_6.x86_64
[root@localhost ~]# dnf list kernel
Последняя проверка окончания срока действия метаданных: 2:12:44 назад, Сб 01 ноя 2025 11:01:40.
Установленные пакеты
kernel.x86_64                  5.14.0-570.17.1.el9_6          @anaco
kernel.x86_64                  5.14.0-570.37.1.el9_6          @baseo
kernel.x86_64                  5.14.0-570.55.1.el9_6          @baseo
[root@localhost ~]#
```

Рис. 10: 10

Обновляем систему

Обновим систему, чтобы убедиться, что все существующие пакеты обновлены

```
[root@localhost ~]# dnf upgrade --refresh
Extra Packages for Enterprise Linux 9 - x86_64
Extra Packages for Enterprise Linux 9 - x86_64
Extra Packages for Enterprise Linux 9 openh264 (From Cisco) - x86_64
Rocky Linux 9 - BaseOS
Rocky Linux 9 - AppStream
Rocky Linux 9 - Extras
Зависимости разрешены.
Отсутствуют действия для выполнения.
Выполнено!
```

Рис. 11: 11

Обновляем ядро операционной системы

Далее обновим ядро операционной системы, а затем саму операционную систему

```
[root@localhost ~]# dnf update kernel
Последняя проверка окончания срока действия метаданных: 0:00:14 назад, Сб 01 мая 2025 13:15:21.
Зависимости разрешены.
Отсутствуют действия для выполнения.
Выполнено!
[root@localhost ~]# dnf update
Последняя проверка окончания срока действия метаданных: 0:00:26 назад, Сб 01 мая 2025 13:15:21.
Зависимости разрешены.
Отсутствуют действия для выполнения.
Выполнено!
[root@localhost ~]# dnf upgrade --refresh
Extra Packages for Enterprise Linux 9 - x86_64                               12 kB/s | 11 kB     00:00
Extra Packages for Enterprise Linux 9 openh264 (From Cisco) - x86_64           1.3 kB/s | 993 B    00:00
Rocky Linux 9 - BaseOS                                         4.0 kB/s | 4.1 kB     00:01
Rocky Linux 9 - AppStream                                     6.7 kB/s | 4.5 kB     00:00
Rocky Linux 9 - Extras                                       5.0 kB/s | 2.9 kB     00:00
Зависимости разрешены.
Отсутствуют действия для выполнения.
Выполнено!
```

Рис. 12: 12

Проверка версии ядра после перезагрузки

После перезагрузки просмотрим версию ядра, используемую в операционной системе

```
[sikaznacheev@localhost ~]$ uname -r
5.14.0-570.55.1.el9_6.x86_64
[sikaznacheev@localhost ~]$ hostname
localhost.localdomain
[sikaznacheev@localhost ~]$ hostnamectl
    Static hostname: (unset)
Transient hostname: localhost
        Icon name: computer-vm
          Chassis: vm [23]
Machine ID: 24895b72da964397829f89f50c792b1b
      Boot ID: d7aa89f40a5f4d3cb51237241ebfd7f9
Virtualization: oracle
Operating System: Rocky Linux 9.6 (Blue Onyx)
      CPE OS Name: cpe:/o:rocky:rocky:9::baseos
        Kernel: Linux 5.14.0-570.55.1.el9_6.x86_64
   Architecture: x86-64
Hardware Vendor: innotek GmbH
  Hardware Model: VirtualBox
Firmware Version: VirtualBox
[sikaznacheev@localhost ~]$
```

Рис. 13: 13

Контрольные вопросы

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

Ответ - команда uname -r

Контрольные вопросы 2

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

Ответ - чтобы просмотреть более подробную информацию о текущей версии ядра операционной системы командой uname -a

Контрольные вопросы 3

3. Какая команда показывает список загруженных модулей ядра?

Ответ - команда lsmod

Контрольные вопросы 4

4. Какая команда позволяет вам определять параметры модуля ядра?

Ответ - команда modinfo “имя_модуля”

Контрольные вопросы 5

5. Как выгрузить модуль ядра?

Ответ - с помощью команды modprobe -r "имя_модуля"

Контрольные вопросы 6

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

Ответ - при ошибки выгрузки модуля надо проверить зависимость `lsmod | grep` “модуль”, размонтировать связанные файловые системы, остановить использующие сервисы и выгрузить зависимые модули

Контрольные вопросы 7

7. Как определить, какие параметры модуля ядра поддерживаются?

Ответ - определение поддерживаемых параметров командой modinfo “имя_модуль”

Контрольные вопросы 8

8. Как установить новую версию ядра?

Ответ - установка новой версии ядра через пакетный менеджер или обновление дистрибутива, после чего перезагружаем систему

Выводы

После выполнения лабораторной работы я получил навыки работы с утилитами управления модулями ядра операционной системы