

Отчёт по лабораторной работе №1

Дисциплина: Основы информационной безопасности

Кузьмина Мария Константиновна

Table of contents

Цель работы	4
Задание	5
Выполнение лабораторной работы	6
Подготовка виртуального окружения	6
Использование команды dmesg	7
Фильтрация вывода dmesg	7
Контрольные вопросы	9
Выводы	13

List of Figures

1	снимок экрана	6
2	снимок экрана	7
3	снимок экрана	8

Цель работы

Знакомство с командами для получения информации об аппаратном обеспечении и ядре операционной системы Linux

Задание

1. Установить и настроить виртуальную машину с ОС Rocky Linux
2. Ознакомиться с выводом команды 'dmesg'
3. Выполнить фильтрацию вывода 'dmesg' с помощью 'grep' для поиска информации о конкретных устройствах
4. Ответить на контрольные вопросы

Выполнение лабораторной работы

Подготовка виртуального окружения

Для выполнения работы была установлена виртуальная машина с операционной системой Rocky. На скриншоте ниже представлен процесс настройки параметров виртуальной машины (имя, тип ОС, объем памяти и размер диска) (рис.1)

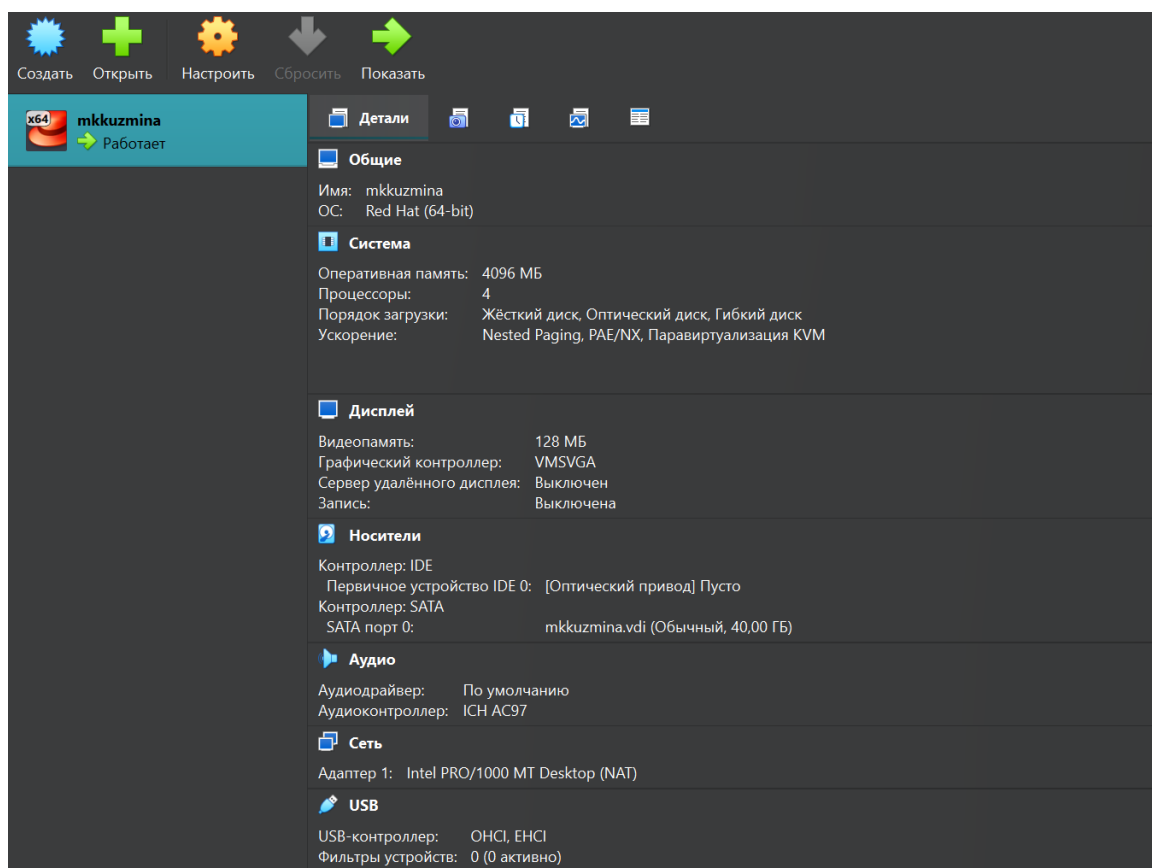
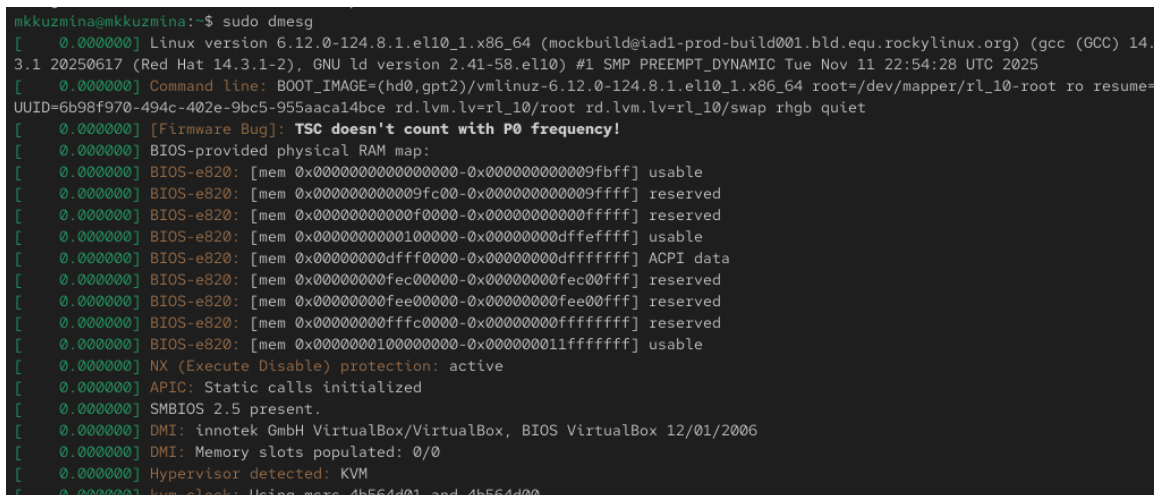


Figure 1: снимок экрана

Использование команды dmesg

Команда используется для просмотра кольцевого буфера сообщений ядра. Она позволяет увидеть информацию об обнаруженных устройствах, драйверах и ошибках при загрузке системы

Был выполнен базовый вызов команды ‘dmesg’, который выводит все сообщения ядра с момента загрузки (рис.2)



```
mkkuzmina@mkkuzmina:~$ sudo dmesg
[ 0.000000] Linux version 6.12.0-124.8.1.el10_1.x86_64 (mockbuild@iad1-prod-build001.bld.equ.rockylinux.org) (gcc (GCC) 14.3.1 20250617 (Red Hat 14.3.1-2), GNU ld version 2.41-58.el10) #1 SMP PREEMPT_DYNAMIC Tue Nov 11 22:54:28 UTC 2025
[ 0.000000] Command line: BOOT_IMAGE=(hd0,gpt2)/vmlinuz-6.12.0-124.8.1.el10_1.x86_64 root=/dev/mapper/rl_10-root ro resume=UUID=6b98f970-494c-402e-9bc5-955aaca14bce rd.lvm.lv=rl_10/root rd.lvm.lv=rl_10/swap rhgb quiet
[ 0.000000] [Firmware Bug]: TSC doesn't count with P0 frequency!
[ 0.000000] BIOS-provided physical RAM map:
[ 0.000000] BIOS-e820: [mem 0x0000000000000000-0x000000000009fbff] usable
[ 0.000000] BIOS-e820: [mem 0x000000000009fc00-0x000000000009ffff] reserved
[ 0.000000] BIOS-e820: [mem 0x00000000000f0000-0x00000000000fffff] reserved
[ 0.000000] BIOS-e820: [mem 0x0000000000100000-0x0000000000dfffff] usable
[ 0.000000] BIOS-e820: [mem 0x000000000dfff0000-0x000000000dffffff] ACPI data
[ 0.000000] BIOS-e820: [mem 0x00000000fec00000-0x00000000fec0ffff] reserved
[ 0.000000] BIOS-e820: [mem 0x00000000fee00000-0x00000000fee0ffff] reserved
[ 0.000000] BIOS-e820: [mem 0x00000000fffc0000-0x00000000ffffffff] reserved
[ 0.000000] BIOS-e820: [mem 0x0000000100000000-0x000000011fffffffff] usable
[ 0.000000] NX (Execute Disable) protection: active
[ 0.000000] APIC: Static calls initialized
[ 0.000000] SMBIOS 2.5 present.
[ 0.000000] DMI: innotek GmbH VirtualBox/VirtualBox, BIOS VirtualBox 12/01/2006
[ 0.000000] DMI: Memory slots populated: 0/0
[ 0.000000] Hypervisor detected: KVM
[ 0.000000] kvm-clock: Using msrc 4b564d01 and 4b564d00
```

Figure 2: снимок экрана

Фильтрация вывода dmesg

Для поиска конкретной информации, вывод команды ‘dmesg’ был отфильтрован с помощью утилиты ‘grep’. Это позволяет, например, найти сообщения, связанные с конкретным оборудованием.

На скриншоте показан результат выполнения команды ‘dmesg | grep -i “CPU0”’, отображающий только строки, содержащие информацию о процессоре (рис.3)

```

mkkuzmina@mkkuzmina:~$ sudo dmesg | grep -i "linux version"
[ 0.000000] Linux version 6.12.0-124.8.1.el10_1.x86_64 (mockbuild@iad1-prod-build001.bld.equ.rockylinux.org) (gcc (GCC) 14.3.1 20250617 (Red Hat 14.3.1-2), GNU ld version 2.41-58.el10) #1 SMP PREEMPT_DYNAMIC Tue Nov 11 22:54:28 UTC 2025
mkkuzmina@mkkuzmina:~$ sudo dmesg | grep -i "detected mhz processor"
mkkuzmina@mkkuzmina:~$ sudo dmesg | grep -i "processor" | grep -i "MHz"
[ 0.000030] tsc: Detected 2096.060 MHz processor
mkkuzmina@mkkuzmina:~$ sudo dmesg | grep -i "CPU0"
[ 0.317101] smpboot: CPU0: AMD Ryzen 5 3500U with Radeon Vega Mobile Gfx (family: 0x17, model: 0x18, stepping: 0x1)
mkkuzmina@mkkuzmina:~$ sudo dmesg | grep -i "memory available"
mkkuzmina@mkkuzmina:~$ sudo dmesg | grep -i "memory.*available"
[ 0.327909] Memory: 3943616K/4193948K available (18432K kernel code, 5804K rwddata, 14268K rodata, 4344K init, 6696K bss, 245864K reserved, 0K cma-reserved)
mkkuzmina@mkkuzmina:~$ sudo dmesg | grep -i "hypervisor detected"
[ 0.000000] Hypervisor detected: KVM
mkkuzmina@mkkuzmina:~$ findmnt -no FSTYPE /
xfs

```

Figure 3: снимок экрана

Контрольные вопросы

1. Какую информацию содержит учётная запись пользователя? Учётная запись пользователя в Linux содержит следующую информацию:

- **Имя пользователя (username):** уникальное имя для входа в систему.
- **Пароль (password):** хранится в зашифрованном виде (обычно в файле `/etc/shadow`).
- **UID (User ID):** уникальный числовой идентификатор пользователя.
- **GID (Group ID):** идентификатор основной группы пользователя.
- **Домашний каталог (home directory):** путь к личной папке пользователя (например, `/home/username`).
- **Командная оболочка (shell):** программа-интерпретатор, запускаемая после входа (например, `/bin/bash`).

2. Укажите команды терминала и приведите примеры:

- **Для получения справки по команде:**
 - `man <команда>` — выводит подробное руководство.
* Пример: `man ls`
 - `<команда> --help` — выводит краткую справку по использованию.
* Пример: `ls --help`
- **Для перемещения по файловой системе:**
 - `cd <путь>` — переход в указанный каталог.
* Пример: `cd /home/user/documents`

- `cd ..` — переход на уровень выше.
- `cd ~` или просто `cd` — переход в домашний каталог.

- **Для просмотра содержимого каталога:**

- `ls` — выводит список файлов и папок в текущем каталоге.
- `ls -l` — выводит подробный список (права, владелец, размер, дата).
 - * *Пример:* `ls -l /etc`
- `ls -a` — показывает также скрытые файлы (начинающиеся с точки).

- **Для определения объёма каталога:**

- `du -sh <каталог>` — показывает общий размер каталога в удобочитаемом виде.
 - * *Пример:* `du -sh /home/user`

- **Для создания / удаления каталогов / файлов:**

- Создание каталога: `mkdir <имя_каталога>`
 - * *Пример:* `mkdir new_folder`
- Создание файла: `touch <имя_файла>` или `> <имя_файла>`
 - * *Пример:* `touch readme.txt`
- Удаление файла: `rm <имя_файла>`
 - * *Пример:* `rm readme.txt`
- Удаление пустого каталога: `rmdir <имя_каталога>`
- Удаление каталога с содержимым: `rm -r <имя_каталога>`
 - * *Пример:* `rm -r new_folder`

- **Для задания определённых прав на файл / каталог:**

- Команда: `chmod <права> <файл/каталог>`
- Права можно задавать символьно или цифрами.
 - * *Пример (символьный):* `chmod u+x script.sh` — добавить право на выполнение владельцу.

* *Пример (цифровой):* `chmod 755 script.sh` — владелец может читать/писать/выполнять, остальные — только читать/выполнять.

- **Для просмотра истории команд:**

- `history` — показывает список ранее введенных команд.
- `!!` — повторяет последнюю выполненную команду.
- `!<номер>` — выполняет команду с указанным номером из истории.

3. **Что такое файловая система? Приведите примеры с краткой характеристикой.**

Файловая система (ФС) — это способ организации, хранения и именования данных на носителях информации (жестких дисках, SSD, флеш-накопителях). Она определяет структуру каталогов, правила доступа к файлам и методы их размещения.

Примеры:

- **ext4 (Fourth Extended Filesystem):** стандартная файловая система для Linux. Поддерживает журналирование (защита от сбоев), большие файлы и разделы, высокую производительность.
- **NTFS (New Technology File System):** основная ФС для Windows. Поддерживает журналирование, шифрование, разграничение прав доступа. Linux умеет читать и писать на NTFS.
- **FAT32 (File Allocation Table):** старая ФС, совместимая со всеми ОС. Не поддерживает файлы больше 4 ГБ. Часто используется на флешках для совместимости.
- **XFS:** высокопроизводительная журналируемая ФС, хорошо работает с большими файлами. Часто используется на серверах.

4. **Как посмотреть, какие файловые системы смонтированы в ОС?** Для просмотра смонтированных файловых систем используются команды:

- `mount` — выводит список всех смонтированных ФС.

- `df -h` — показывает список смонтированных ФС, их размер, занятое и свободное место в удобочитаемом формате.
- `findmnt` — выводит список смонтированных ФС в древовидном формате.

5. **Как удалить зависший процесс?** Для удаления (завершения) зависшего процесса используется команда `kill`. Сначала нужно найти идентификатор процесса (PID):

1. Найти PID процесса: `ps aux | grep <имя_процесса>` или `top`.

2. Завершить процесс:

- `kill <PID>` — отправляет сигнал `SIGTERM` (мягкое завершение, процесс может попытаться закрыться корректно).
- `kill -9 <PID>` — отправляет сигнал `SIGKILL` (немедленное принудительное завершение, если обычный `kill` не помогает).

Пример: `kill -9 1234` (где 1234 — PID зависшего процесса).

Выводы

В ходе выполнения лабораторной работы были изучены основные команды для диагностики системы Linux. На практике освоено использование команды 'dmesg' и фильтрация ее вывода с помощью 'grep'.