

Отчёт по лабораторной работе №1

Основы информационной безопасности

Бережной Иван Александрович

Содержание

| | | |
|----------|---------------------------------------|-----------|
| 1 | Цель работы | 4 |
| 2 | Задание | 5 |
| 3 | Выполнение лабораторной работы | 6 |
| 4 | Выполнение домашнего задания | 13 |
| 5 | Ответы на контрольные вопросы | 15 |
| 6 | Выводы | 17 |

Список иллюстраций

| | | |
|------|--|----|
| 3.1 | Создание ВМ | 6 |
| 3.2 | Выбор языка ОС | 7 |
| 3.3 | Выбор доп языка ОС | 7 |
| 3.4 | Выбор языков клавиатуры | 7 |
| 3.5 | Дополнительное ПО | 8 |
| 3.6 | Выбор диска | 8 |
| 3.7 | Отключение KDUMP | 8 |
| 3.8 | Настройка Ethernet | 9 |
| 3.9 | Пароль для root | 9 |
| 3.10 | Добавление пользователя | 10 |
| 3.11 | Завершение установки | 10 |
| 3.12 | Первый взгляд | 11 |
| 3.13 | Установка дополнений гостевой ОС | 11 |
| 3.14 | Запуск установки | 12 |
| 4.1 | Обзор dmesg | 13 |
| 4.2 | Использование dmesg 1 | 13 |
| 4.3 | Использование dmesg 2 | 14 |
| 4.4 | Использование dmesg 3 | 14 |

1 Цель работы

Целью данной работы является приобретение практических навыков установки операционной системы на виртуальную машину, настройки минимально необходимых для дальнейшей работы сервисов.

2 Задание

1. Установить операционную систему Rocky на виртуальную машину
2. Получить следующую информацию:
 1. Версия ядра Linux (Linux version).
 2. Частота процессора (Detected Mhz processor).
 3. Модель процессора (CPU0).
 4. Объем доступной оперативной памяти (Memory available).
 5. Тип обнаруженного гипервизора (Hypervisor detected).
 6. Тип файловой системы корневого раздела.
 7. Последовательность монтирования файловых систем.

3 Выполнение лабораторной работы

Создадим новую виртуальную машину в VirtualBox. Зададим имя, количество ядер (4), объём оперативной памяти (4гб) и размер диска (45гб). Подключим ISO-файл с ос Rocky (рис. 3.1).

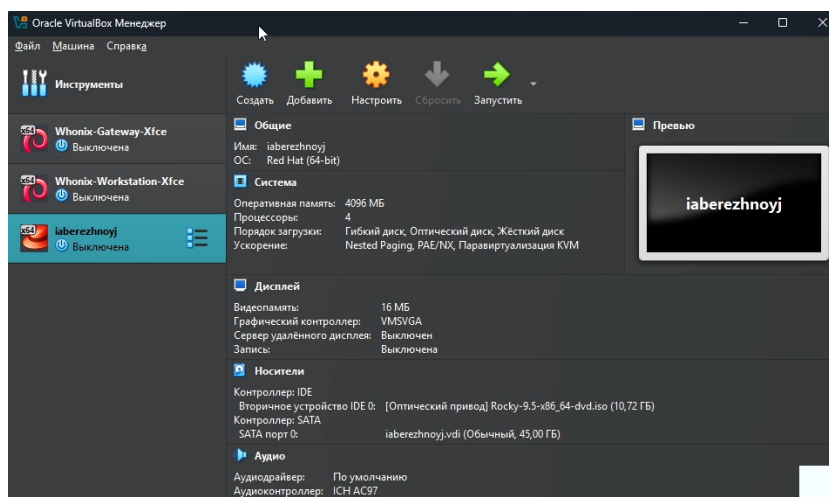


Рис. 3.1: Создание ВМ

Выберем основной язык ОС (рис. 3.2).

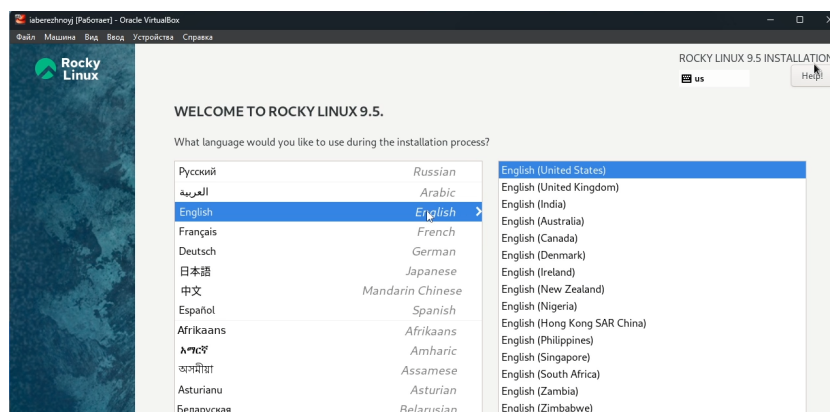


Рис. 3.2: Выбор языка ОС

И дополнительный язык. Разумеется, русский (рис. 3.3).

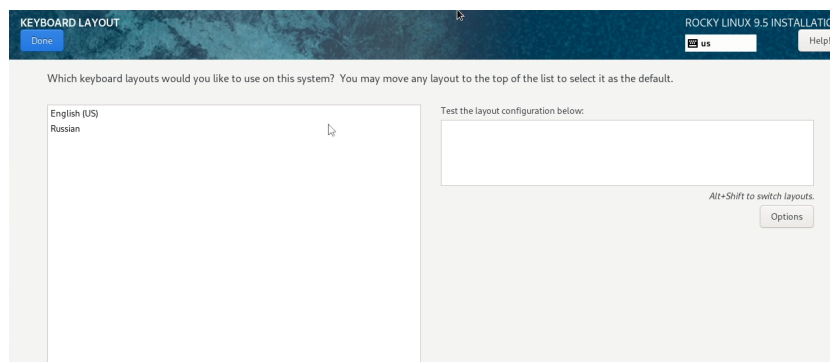


Рис. 3.3: Выбор доп языка ОС

Выберем языки для клавиатуры. Также русский и английский. Не забудем установить горячие клавиши для смены языка. В данном случае alt+shift (рис. 3.4).

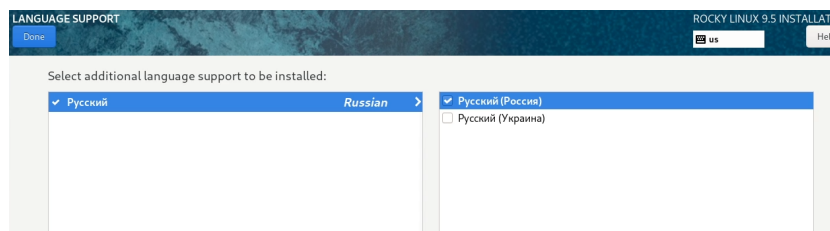


Рис. 3.4: Выбор языков клавиатуры

Установим дополнительное ПО, которое нам пригодится, а именно графическую оболочку и инструменты разработчика (рис. 3.5).

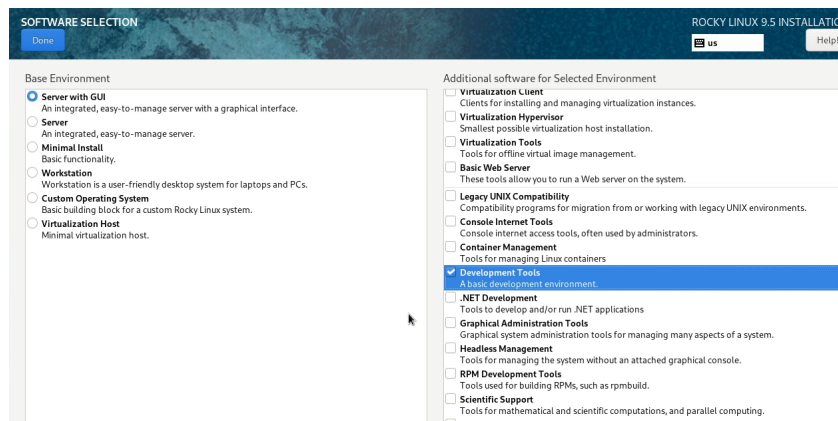


Рис. 3.5: Дополнительное ПО

Выберем диск, куда установим ОС (рис. 3.6), отключим KDUMP (рис. 3.7) и настроим выход в интернет (рис. 3.8).

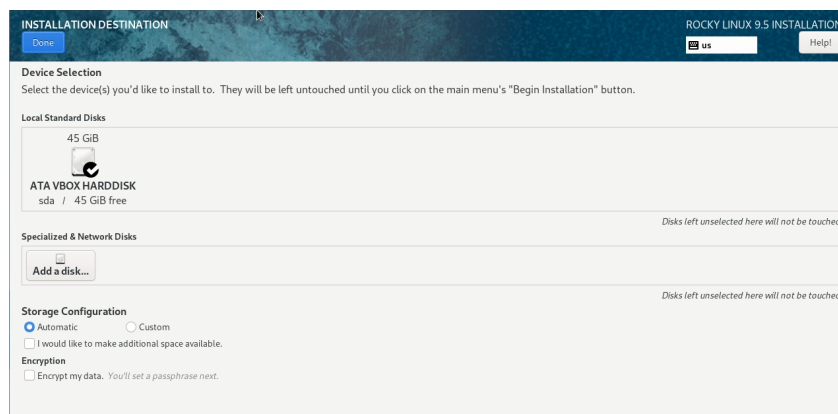


Рис. 3.6: Выбор диска

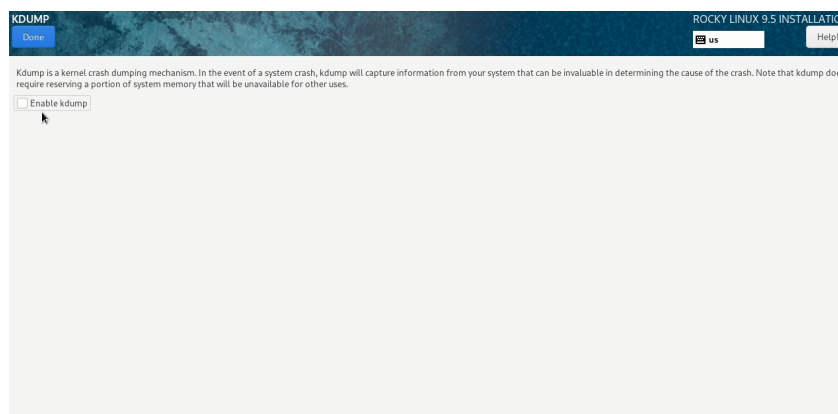


Рис. 3.7: Отключение KDUMP

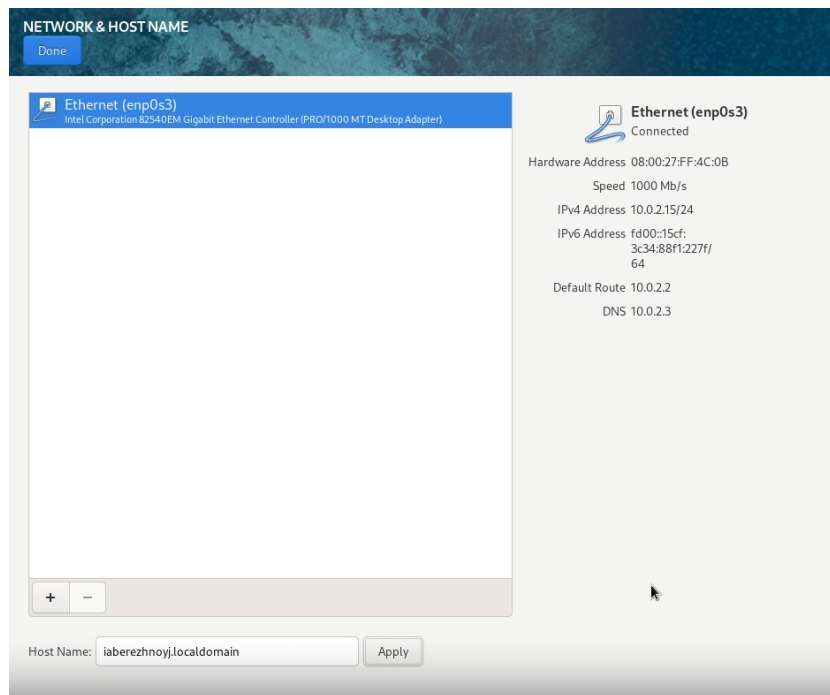


Рис. 3.8: Настройка Ethernet

Установим пароль для рута (рис. 3.9), а также создадим пользователя с правами администратора (рис. 3.10).

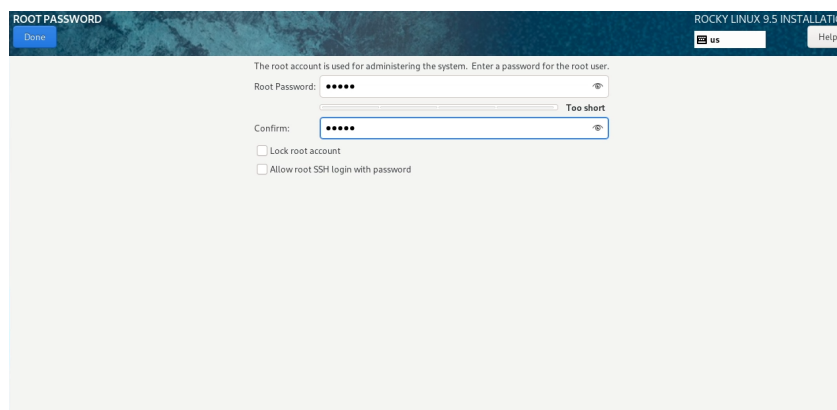


Рис. 3.9: Пароль для root

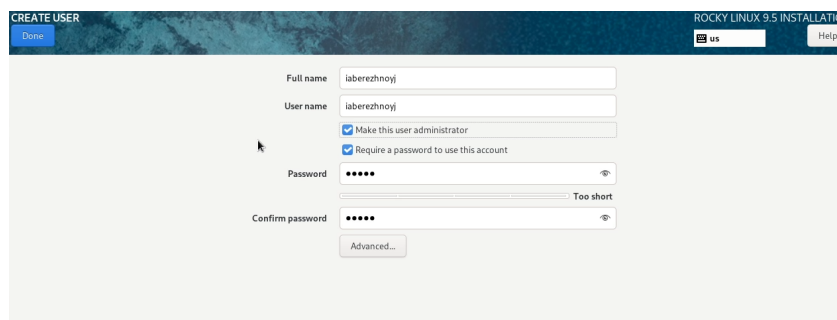
The image shows the 'CREATE USER' screen in the Rocky Linux 9.5 installer. The title bar at the top says 'CREATE USER' on the left and 'ROCKY LINUX 9.5 INSTALLATION' on the right. Below the title bar, there are input fields for 'Full name' and 'User name', both containing the text 'laberezhnoy'. Below these are two checkboxes: 'Make this user administrator' (checked) and 'Require a password to use this account' (checked). There are password fields for 'Password' and 'Confirm password', both showing six dots. A 'Too short' error message is visible next to the 'Confirm password' field. An 'Advanced...' button is located below the password fields. A 'Done' button is in the top left corner, and a 'Help' button is in the top right corner.

Рис. 3.10: Добавление пользователя

После завершения установки (рис. 3.11) нам откроется графическая оболочка (рис. 3.12). Установим дополнения гостевой ОС в разделе “Устройства” -> “Подключить образы дополнений гостевой ОС” в верхнем меню VirtualBox (рис. 3.13). Жмём “Run” (рис. 3.14).

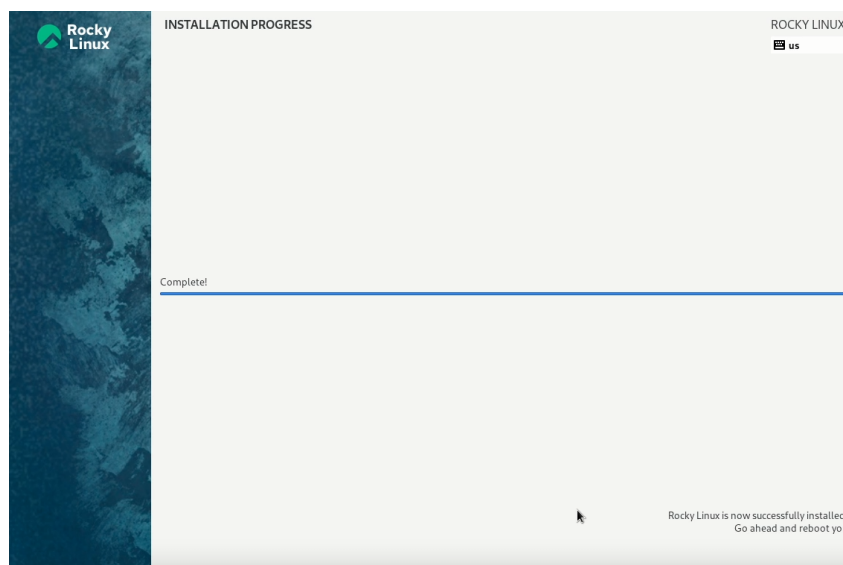


Рис. 3.11: Завершение установки

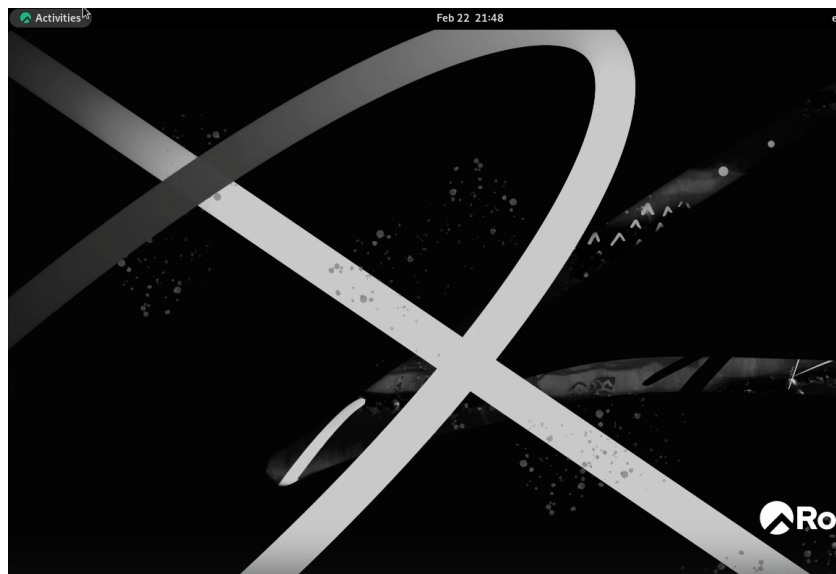


Рис. 3.12: Первый взгляд

```
VirtualBox Guest Additions installation
Verifying archive integrity... 100% MD5 checksums are OK. All good.
Uncompressing VirtualBox 7.1.4 Guest Additions for Linux 100%
VirtualBox Guest Additions installer
VirtualBox Guest Additions: Starting.
VirtualBox Guest Additions: Setting up modules
VirtualBox Guest Additions: Building the VirtualBox Guest Additions kernel
modules. This may take a while.
VirtualBox Guest Additions: To build modules for other installed kernels, run
VirtualBox Guest Additions: /sbin/rcvboxadd quicksetup <version>
VirtualBox Guest Additions: or
VirtualBox Guest Additions: /sbin/rcvboxadd quicksetup all
VirtualBox Guest Additions: Building the modules for kernel
5.14.0-503.14.1.el9_5.x86_64.
VirtualBox Guest Additions: reloading kernel modules and services
VirtualBox Guest Additions: kernel modules and services 7.1.4 r165100 reloaded
VirtualBox Guest Additions: NOTE: you may still consider to re-login if some
user session specific services (Shared Clipboard, Drag and Drop, Seamless or
Guest Screen Resize) were not restarted automatically
Press Return to close this window...
```

Рис. 3.13: Установка дополнений гостевой ОС

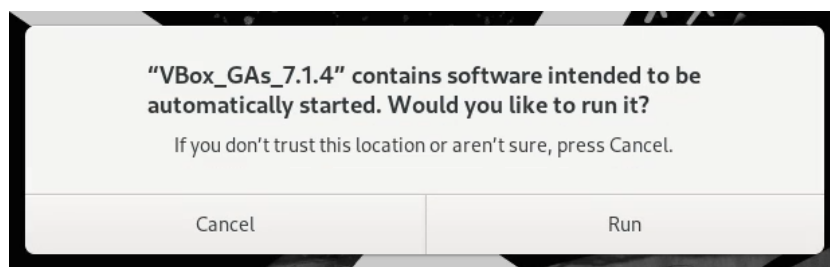
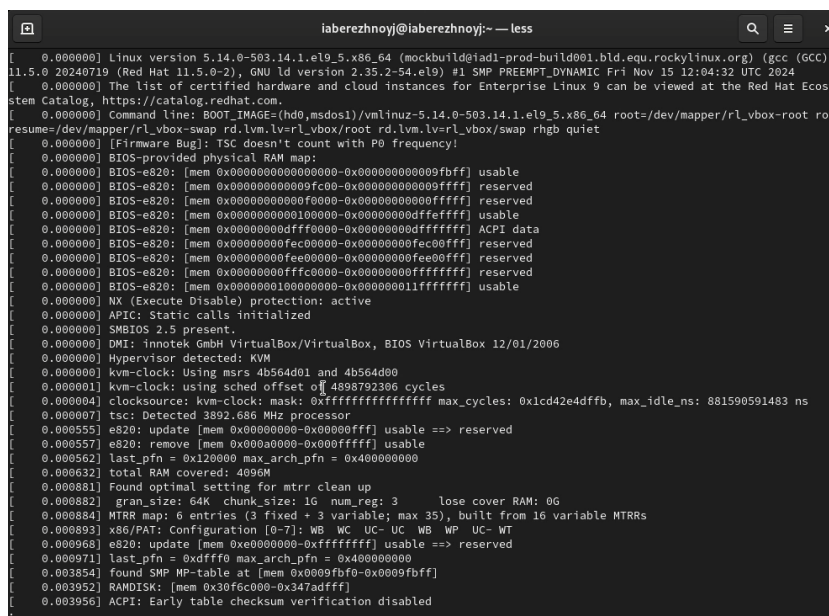


Рис. 3.14: Запуск установки

4 Выполнение домашнего задания

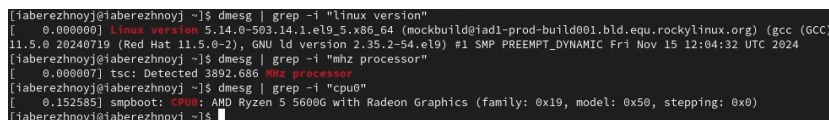
Откроем командную строку и впишем `dmesg | less` (рис. 4.1). Чтобы выйти из процесса, нажмём `Ctrl+Z`.



```
iaberezhnoyj@iaberezhnoyj:~ — less
[ 0.000000] Linux version 5.14.0-503.14.1.el9_5.x86_64 (mockbuild@iad1-prod-build001.bld.equ.rockylinux.org) (gcc (GCC)
11.5.0 20240719 (Red Hat 11.5.0-2), GNU ld version 2.35.2-54.el9) #1 SMP PREEMPT_DYNAMIC Fri Nov 15 12:04:32 UTC 2024
[ 0.000000] The list of certified hardware and cloud instances for Enterprise Linux 9 can be viewed at the Red Hat Ecosy
stem Catalog, https://catalog.redhat.com.
[ 0.000000] Command line: BOOT_IMAGE=(hd0,msdos1)/vmlinuz-5.14.0-503.14.1.el9_5.x86_64 root=/dev/mapper/r1_vbox-root ro
resume=/dev/mapper/r1_vbox-swap rd.lvm.lv=r1_vbox/root rd.lvm.lv=r1_vbox/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 0x000000000009fc00-0x0000000000009fff] reserved
[ 0.000000] BIOS-e820: [mem 0x000000000000f000-0x000000000000ffff] reserved
[ 0.000000] BIOS-e820: [mem 0x0000000000100000-0x00000000000dffff] usable
[ 0.000000] BIOS-e820: [mem 0x00000000000dffff0000-0x00000000000dffffffffff] ACPI data
[ 0.000000] BIOS-e820: [mem 0x0000000000000000-0x00000000000000ff] reserved
[ 0.000000] BIOS-e820: [mem 0x0000000000000000-0x00000000000000ff] reserved
[ 0.000000] BIOS-e820: [mem 0x0000000000000000-0x00000000000000ff] reserved
[ 0.000000] BIOS-e820: [mem 0x0000000000000000-0x00000000000000ff] reserved
[ 0.000000] BIOS-e820: [mem 0x0000000000000000-0x00000000000000ff] 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] Hypervisor detected: KVM
[ 0.000000] kvm-clock: Using msrc 4b564d01 and 4b564d00
[ 0.000000] kvm-clock: using sched offset of 4898792306 cycles
[ 0.000000] clocksource: kvm-clock: mask: 0xffffffffffffffff max_cycles: 0x1cd42e4dffb, max_idle_ns: 881590591483 ns
[ 0.000000] tsc: Detected 3892.686 MHz processor
[ 0.000555] e820: update [mem 0x00000000-0x000000ff] usable ==> reserved
[ 0.000557] e820: remove [mem 0x000a0000-0x0000ffff] usable
[ 0.000562] last_pfn = 0x120000 max_arch_pfn = 0x400000000
[ 0.000632] total RAM covered: 4096M
[ 0.000881] Found optimal setting for mtrr Clean up
[ 0.000882] gran_size: 64K chunk_size: 16 num_reg: 3 lose cover RAM: 0G
[ 0.000884] MTRR map: 6 entries (3 fixed + 3 variable; max 35), built from 16 variable MTRRs
[ 0.000893] x86/PAT: Configuration [0-7]: WB WC UC- UC WB WP UC- WT
[ 0.000968] e820: update [mem 0x00000000-0xffffffff] usable ==> reserved
[ 0.000971] last_pfn = 0xdffff0 max_arch_pfn = 0x400000000
[ 0.003854] found SMP MP-table at [mem 0x0009fbf0-0x0009fbff]
[ 0.003952] RAMDISK: [mem 0x30f6c000-0x347adfff]
[ 0.003956] ACPI: Early table checksum verification disabled
```

Рис. 4.1: Обзор dmesg

Теперь найдём запрашиваемую информацию, такую как версию ядра, частота процессора и модель процессора (рис. 4.2).



```
iaberezhnoyj@iaberezhnoyj:~$ dmesg | grep -i "linux version"
[ 0.000000] Linux version 5.14.0-503.14.1.el9_5.x86_64 (mockbuild@iad1-prod-build001.bld.equ.rockylinux.org) (gcc (GCC)
11.5.0 20240719 (Red Hat 11.5.0-2), GNU ld version 2.35.2-54.el9) #1 SMP PREEMPT_DYNAMIC Fri Nov 15 12:04:32 UTC 2024
iaberezhnoyj@iaberezhnoyj:~$ dmesg | grep -i "tsc: Detected"
[ 0.000000] tsc: Detected 3892.686 MHz processor
iaberezhnoyj@iaberezhnoyj:~$ dmesg | grep -i "cpu0"
[ 0.152585] smpboot: CPU0: AMD Ryzen 5 5600G with Radeon Graphics (family: 0x19, model: 0x50, stepping: 0x0)
iaberezhnoyj@iaberezhnoyj:~$
```

Рис. 4.2: Использование dmesg 1

Посмотрим на свободный объём оперативной памяти (рис. 4.3).

```

[laberezhnoy@laberezhnoy ~]$ dmesg | grep -i "memory"
[ 0.003973] ACPI: Reserving FACP table memory at [mem 0xdfff00f0-0xdfff01e3]
[ 0.003974] ACPI: Reserving DSDT table memory at [mem 0xdfff0620-0xdfff2972]
[ 0.003974] ACPI: Reserving FACS table memory at [mem 0xdfff0200-0xdfff023f]
[ 0.003975] ACPI: Reserving FACS table memory at [mem 0xdfff0200-0xdfff023f]
[ 0.003975] ACPI: Reserving APIC table memory at [mem 0xdfff0240-0xdfff02ab]
[ 0.003976] ACPI: Reserving SSDT table memory at [mem 0xdfff02b0-0xdfff061b]
[ 0.004245] Early memory node ranges
[ 0.009329] PM: hibernation: Registered nosave memory: [mem 0x00000000-0x00000fff]
[ 0.009330] PM: hibernation: Registered nosave memory: [mem 0x0000f000-0x0000ffff]
[ 0.009330] PM: hibernation: Registered nosave memory: [mem 0x000a0000-0x000aefff]
[ 0.009331] PM: hibernation: Registered nosave memory: [mem 0x000f0000-0x000fffff]
[ 0.009332] PM: hibernation: Registered nosave memory: [mem 0xdfff0000-0xdfff0fff]
[ 0.009332] PM: hibernation: Registered nosave memory: [mem 0xe0000000-0xfefbffff]
[ 0.009332] PM: hibernation: Registered nosave memory: [mem 0xfec00000-0xfec0ffff]
[ 0.009333] PM: hibernation: Registered nosave memory: [mem 0xfec01000-0xfedfffff]
[ 0.009333] PM: hibernation: Registered nosave memory: [mem 0xfec00000-0xfec0ffff]
[ 0.009334] PM: hibernation: Registered nosave memory: [mem 0xfec01000-0xfefbffff]
[ 0.009334] PM: hibernation: Registered nosave memory: [mem 0xfec00000-0xfefbffff]
[ 0.023322] memmap 3625328K/4193848K available (16384K kernel code, 5685K rdata, 12904K rodata, 3976K init, 5672K bss,
250848K reserved, 0K cma-reserved)
[ 0.044829] Freeing SMP alternatives memory: 48K
[ 0.160530] x86/mm: Memory block size: 128MB
[ 0.237192] Non-volatile memory driver v1.3
[ 0.617430] Freeing initrd memory: 57608K
[ 0.757599] Freeing unused decrypted memory: 2028K
[ 0.758179] Freeing unused kernel image (initmem) memory: 3976K
[ 0.758548] Freeing unused kernel image (rodata/data gap) memory: 1432K
[ 2.057982] vmwgfx 0000:00:02.0: [drm] Legacy memory limits: VRAM = 16384 kB, FIFO = 2048 kB, surface = 507904 kB
[ 2.057989] vmwgfx 0000:00:02.0: [drm] Maximum display memory size is 16384 kiB
[laberezhnoy@laberezhnoy ~]$

```

Рис. 4.3: Использование dmesg 2

И, наконец, проверим тип виртуализации (KVM), тип файловой системы корневого раздела (XFS) и порядок монтирования файловых систем (рис. 4.4).

```

[laberezhnoy@laberezhnoy ~]$ dmesg | grep -i "hypervisor"
[ 0.000000] Hypervisor detected: KVM
[ 2.057943] vmwgfx 0000:00:02.0: [drm] *ERROR* vmwgfx seems to be running on an unsupported hypervisor.
[laberezhnoy@laberezhnoy ~]$ dmesg | grep -i "filesystem"
[ 3.965255] XFS (dm-0): Mounting V5 Filesystem 36ca9d6c-30ad-4050-b682-2940a9b57c83
[ 6.072923] XFS (sda1): Mounting V5 Filesystem 69f55c2c-8be4-4d8d-b48a-a98554629eff
[laberezhnoy@laberezhnoy ~]$

```

Рис. 4.4: Использование dmesg 3

5 Ответы на контрольные вопросы

1. Учётная запись пользователя в Linux содержит: имя пользователя (логин), UID (идентификатор пользователя), GID (идентификатор основной группы), домашний каталог, оболочку (shell), хеш пароля, дополнительные группы.
2. Перечислим команды для:
 - Получения справки по команде: `man`
 - Перемещения по файловой системе: `cd`
 - Просмотра содержимого каталога: `ls`
 - Определения объёма каталога: `du`
 - Работы с директориями и файлами: `mkdir` - создание папки, `rmdir` - удаление папки, `rm` - удаление файла, `touch` - создание файла.
 - Изменения прав: `chmod`
 - Просмотра истории команд: `history`
3. Файловая система (ФС) – это способ хранения и организации данных на диске. Примеры файловых систем:
 - `ext4` – стандартная для Linux, поддерживает файлы до 16 ТБ.
 - `XFS` – быстрая, подходит для больших файлов и серверов.
 - `NTFS` – файловая система Windows, поддерживается в Linux.
 - `FAT32` – совместима с разными ОС, но ограничение на файл 4 ГБ.
4. Чтобы посмотреть, какие файлы подмонтированы в ОС, нужно вбить команду `mount`.

5. Зависший процесс можно удалить командой `kill`.

6 Выводы

В ходе выполнения лабораторной работы мы научились создавать виртуальные машины и устанавливать ОС на них. Также поработали с терминалом и вспомнили основные команды.