

Отчет по лабораторной работе №1

Основы информационной безопасности

Гасанова Шакира, НКАбд-03-24

Содержание

| | | |
|----------|---|-----------|
| 1 | Цель работы | 5 |
| 2 | Задание | 6 |
| 3 | Выполнение лабораторной работы | 7 |
| 4 | Выполнение дополнительного задания | 17 |
| 5 | Ответы на контрольные вопросы | 21 |
| 6 | Выводы | 23 |

Список иллюстраций

| | | |
|------|--|----|
| 3.1 | Окно создания виртуальной машины | 7 |
| 3.2 | Окно установки гостевой ОС | 8 |
| 3.3 | Окно выбора основных характеристик для гостевой ОС | 8 |
| 3.4 | Окно выбора объема памяти | 9 |
| 3.5 | Итоговые настройки | 9 |
| 3.6 | Загрузка операционной системы Rocky | 10 |
| 3.7 | Подключенные носители | 10 |
| 3.8 | Выбор языка установки | 11 |
| 3.9 | Окно настроек | 11 |
| 3.10 | Выбор раскладки | 12 |
| 3.11 | Изменение часового пояса | 12 |
| 3.12 | Настройка аккаунта root | 13 |
| 3.13 | Настройка пользователя | 13 |
| 3.14 | Выбор окружения | 14 |
| 3.15 | Отключение kdump | 14 |
| 3.16 | Выбор сети | 15 |
| 3.17 | Установка | 15 |
| 3.18 | Проверка носителей | 16 |
| 3.19 | Окно входа в операционную систему | 16 |
| 4.1 | Окно терминала | 17 |
| 4.2 | Версия ядра | 17 |
| 4.3 | Частота процессора | 18 |
| 4.4 | Модель процессора | 18 |
| 4.5 | Объем доступной оперативной памяти | 19 |
| 4.6 | Тип обнаруженного гипервизора | 19 |
| 4.7 | Тип файловой системы | 19 |
| 4.8 | Последовательность монтирования файловых систем | 20 |

Список таблиц

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. Установка и настройка операционной системы.
2. Найти 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.

3 Выполнение лабораторной работы

Я выполняю лабораторную работу на домашнем оборудовании, поэтому создаю новую виртуальную машину в VirtualBox, выбираю имя, местоположение и образ ISO, устанавливать будем операционную систему Rocky DVD (рис. 1).

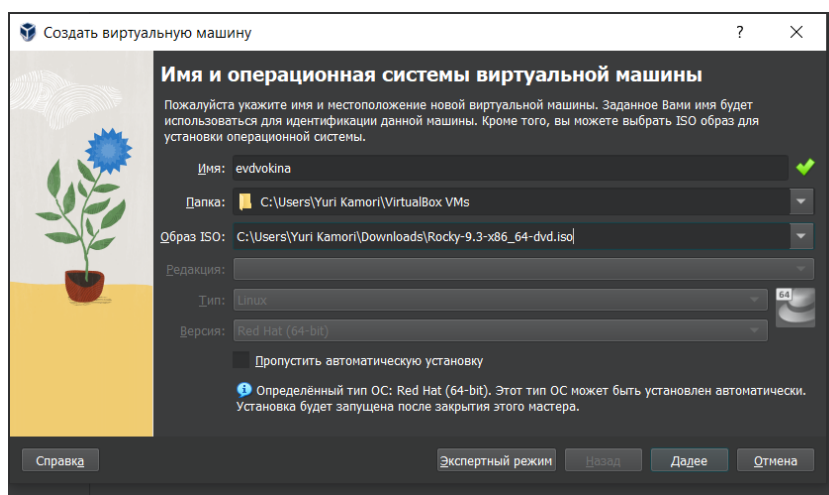


Рис. 3.1: Окно создания виртуальной машины

Предварительно выбираю имя пользователя и имя хоста (рис. 2).

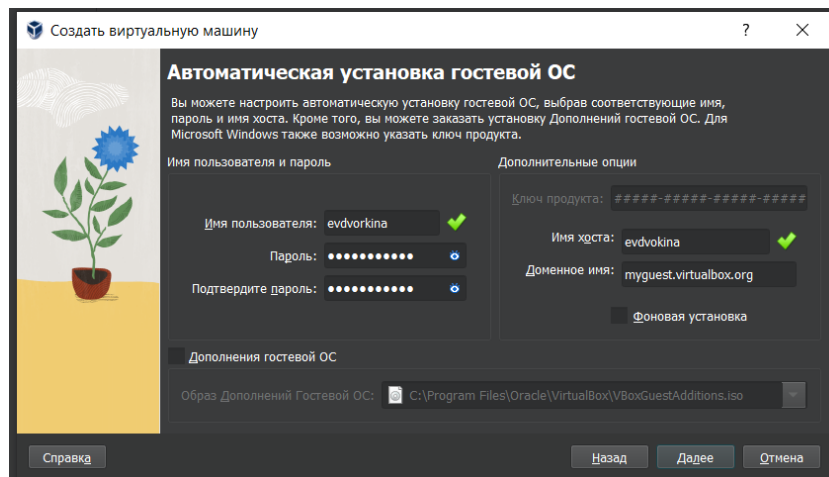


Рис. 3.2: Окно установки гостевой ОС

Выставляю основной памяти размер 2048 Мб, выбираю 3 процессора, чтобы ничего не висло (рис. 3).

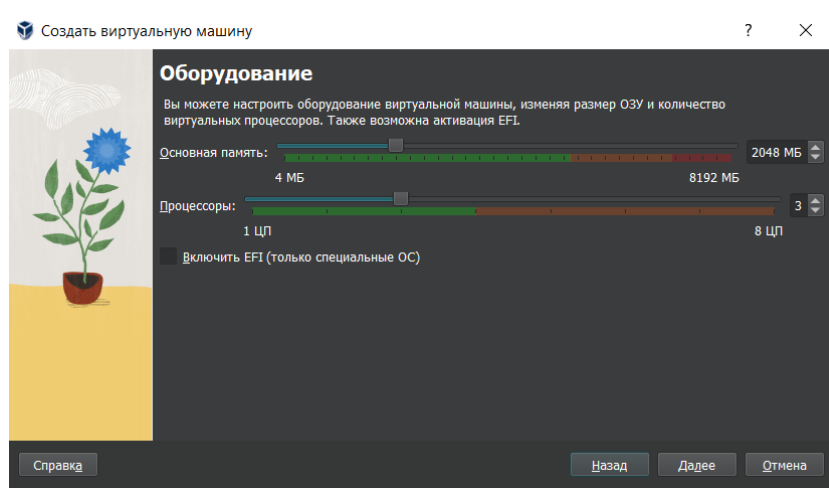


Рис. 3.3: Окно выбора основных характеристик для гостевой ОС

Выделаю 40 Гб памяти на виртуальном жестком диске (рис. 4).

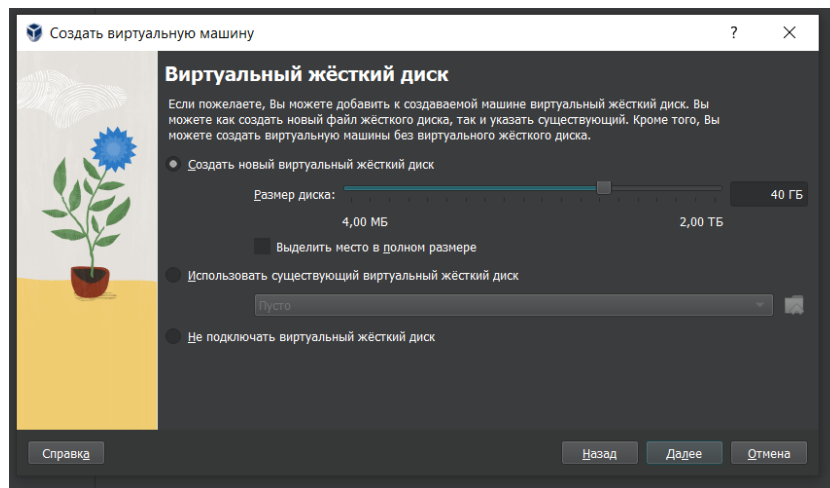


Рис. 3.4: Окно выбора объема памяти

Соглашаюсь с предоставленными настройками (рис. 5).

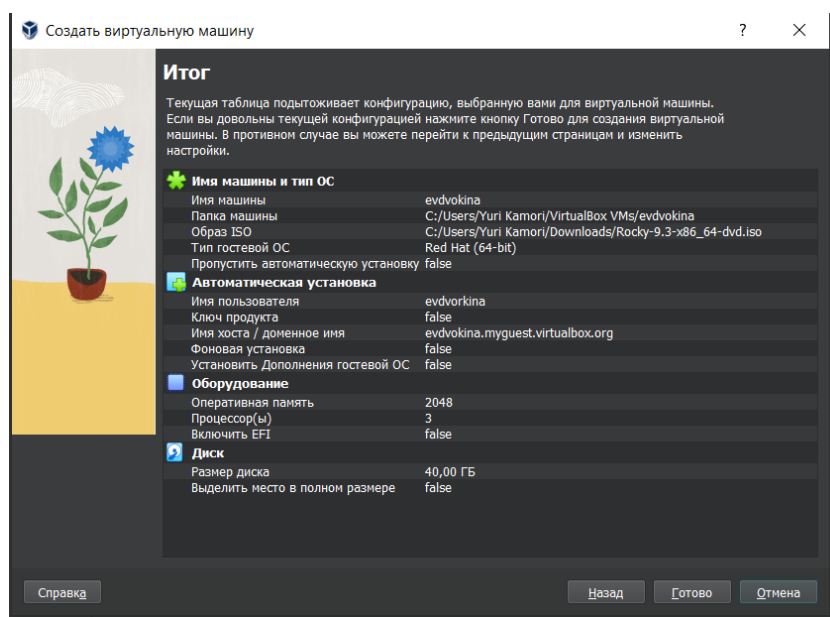


Рис. 3.5: Итоговые настройки

Начинается загрузка операционной системы (рис. 6).

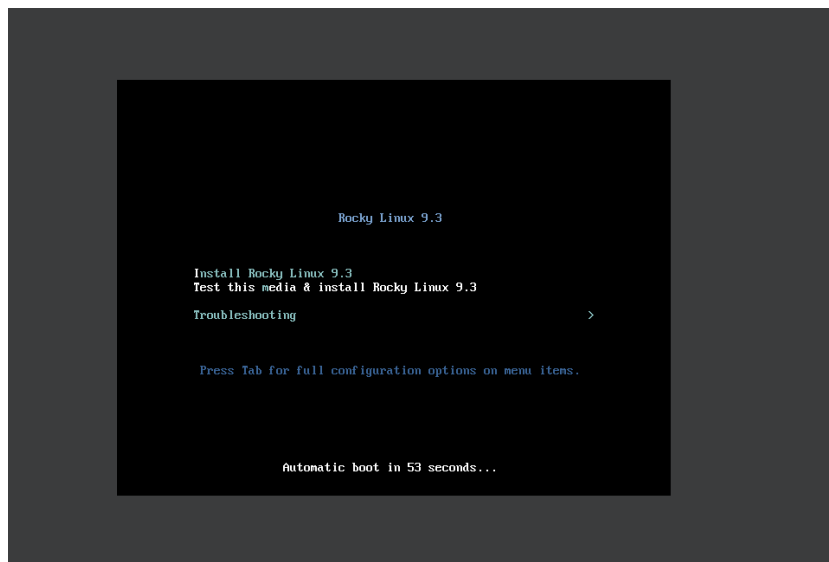


Рис. 3.6: Загрузка операционной системы Rocky

При этом должен быть подключен в носители образ диска! (рис. 7).

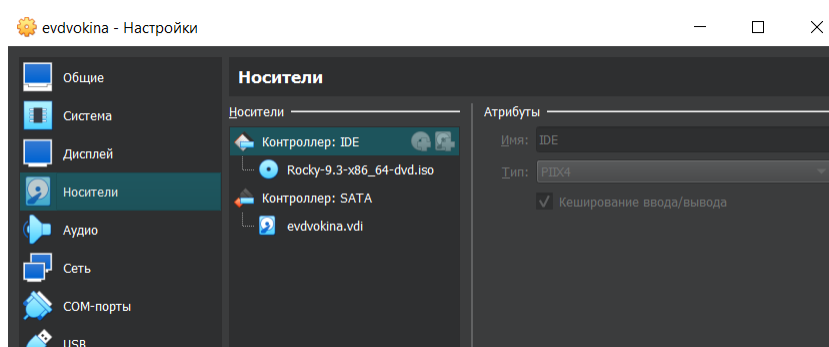


Рис. 3.7: Подключенные носители

Выбираю язык установки (рис. 8).

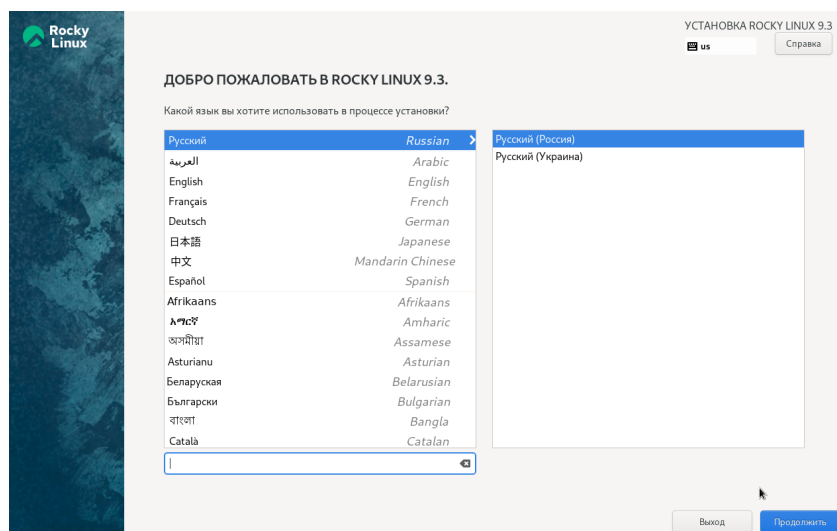


Рис. 3.8: Выбор языка установки

В обзоре установки будем проверять все настройки и менять на нужные (рис. 9).

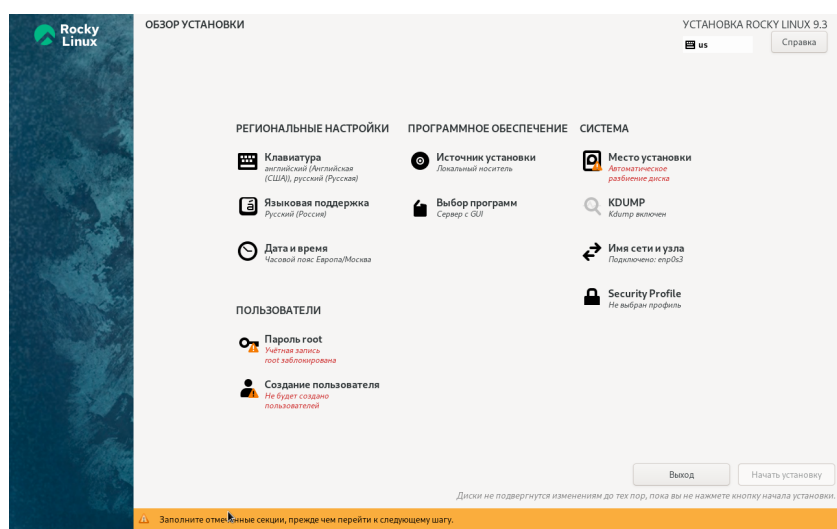


Рис. 3.9: Окно настроек

Язык раскладки должен быть русский и английский (рис. 10).

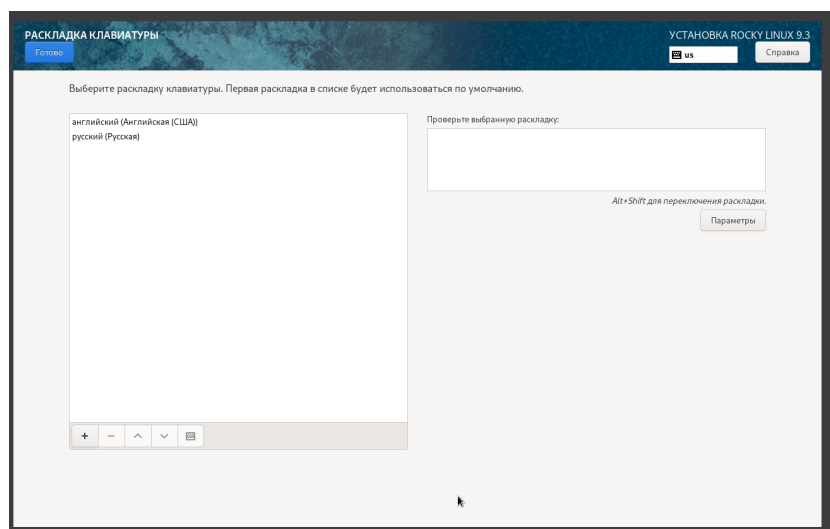


Рис. 3.10: Выбор раскладки

Часовой пояс поменяла на московское время (рис. 11).

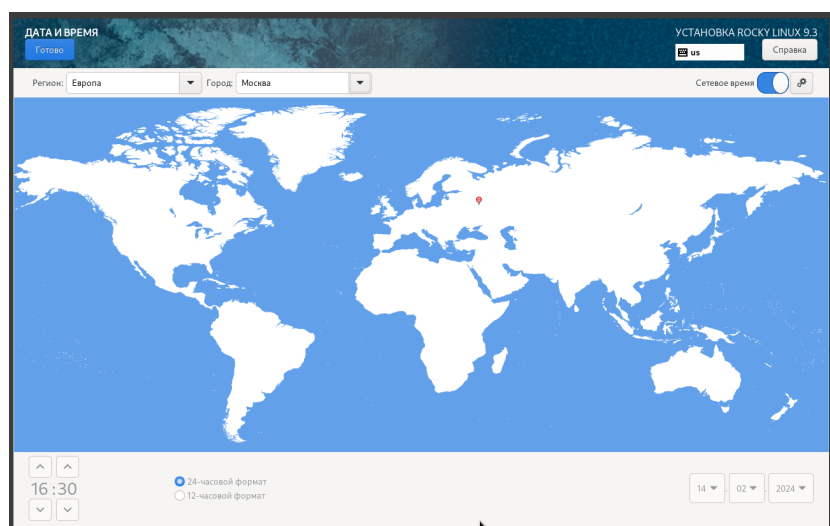


Рис. 3.11: Изменение часового пояса

Установила пароль для администратора (рис. 12).

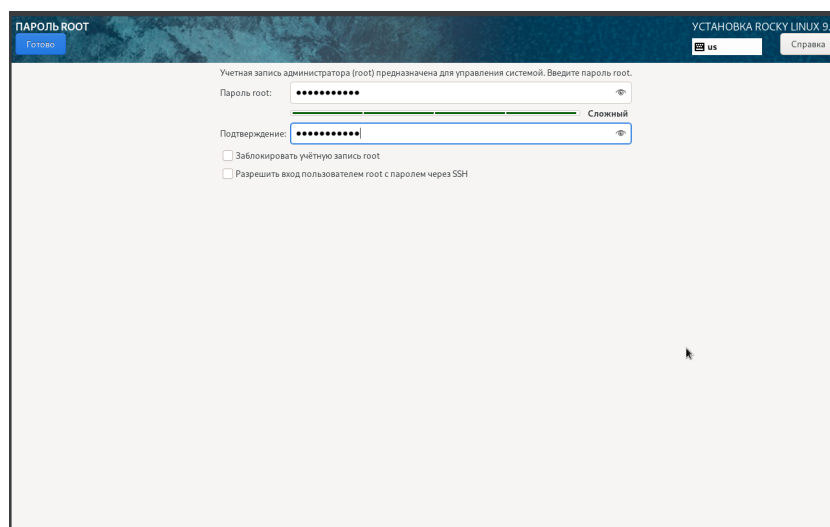


Рис. 3.12: Настройка аккаунта root

Для пользователя так же сделала пароль и сделала этого пользователя администратором (рис. 13).

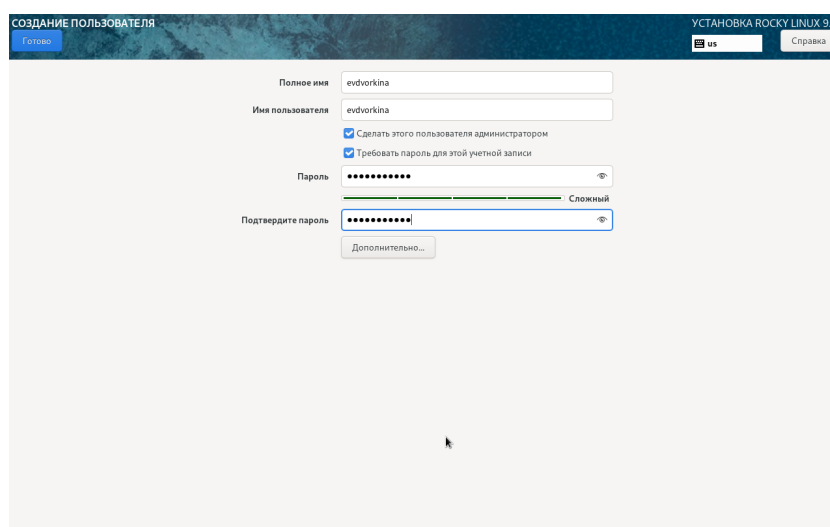


Рис. 3.13: Настройка пользователя

В соответствии с требованием лабораторной работы выбираю окружение сервер с GUB и средства разработки в дополнительном программном обеспечении (рис. 14).

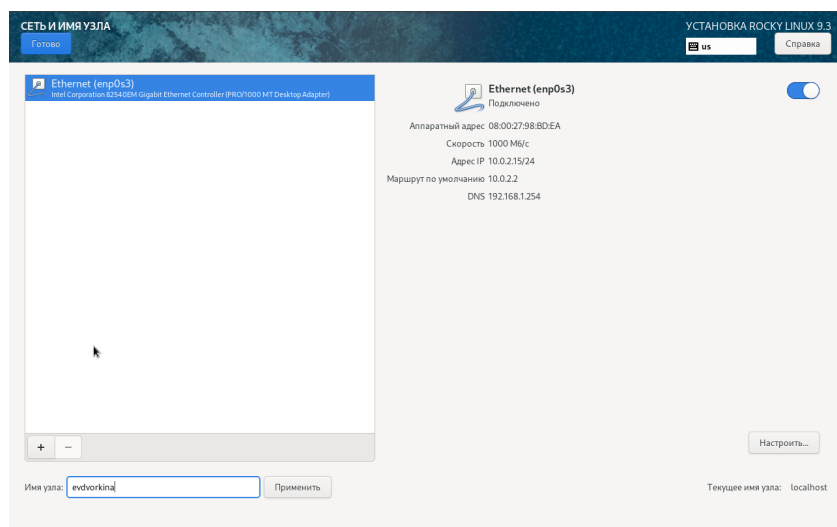


Рис. 3.16: Выбор сети

Начало установки (рис. 17).

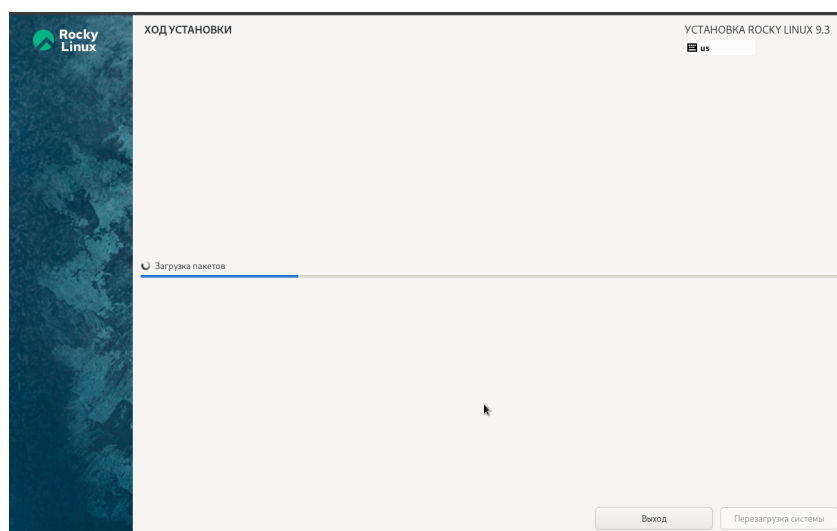


Рис. 3.17: Установка

После завершения установки образ диска сам пропадет из носителей (рис. 18).

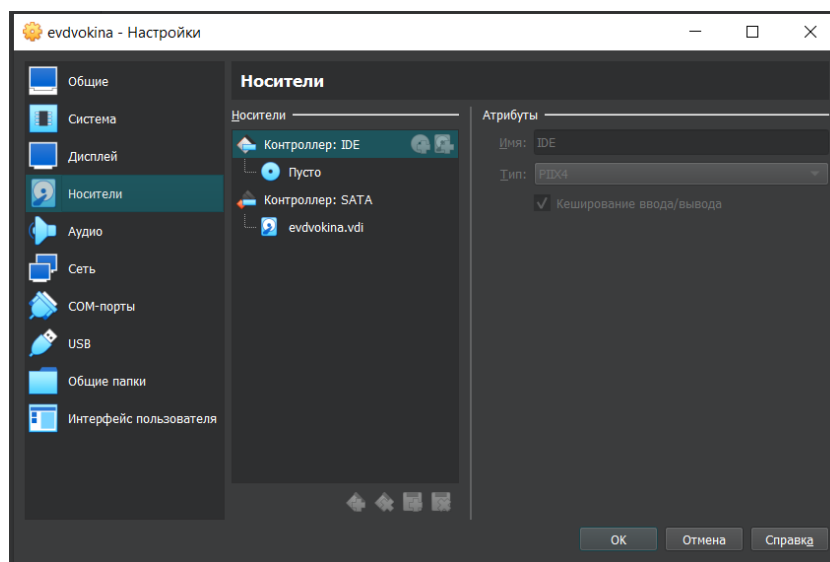


Рис. 3.18: Проверка носителей

После установки при запуске операционной системы появляется окно выбора пользователя (рис. 19).

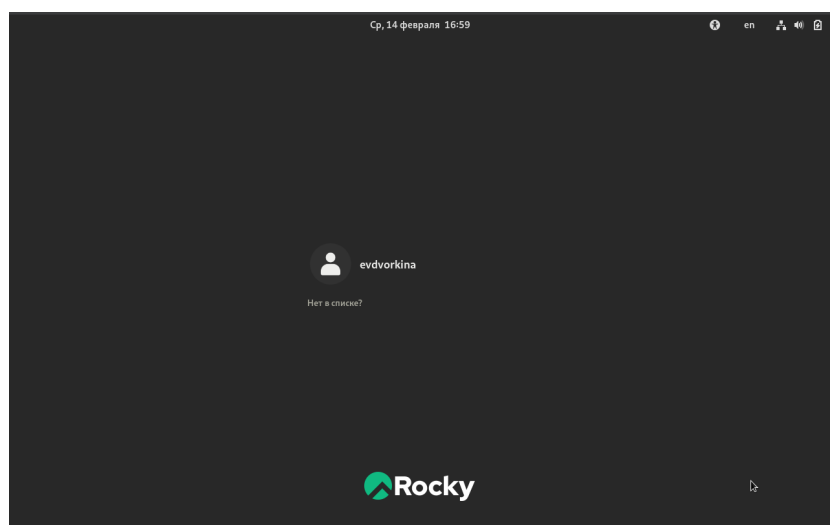


Рис. 3.19: Окно входа в операционную систему

4 Выполнение дополнительного задания

Открываю терминал, в нем прописываю `dmesg | less` (рис. 20).

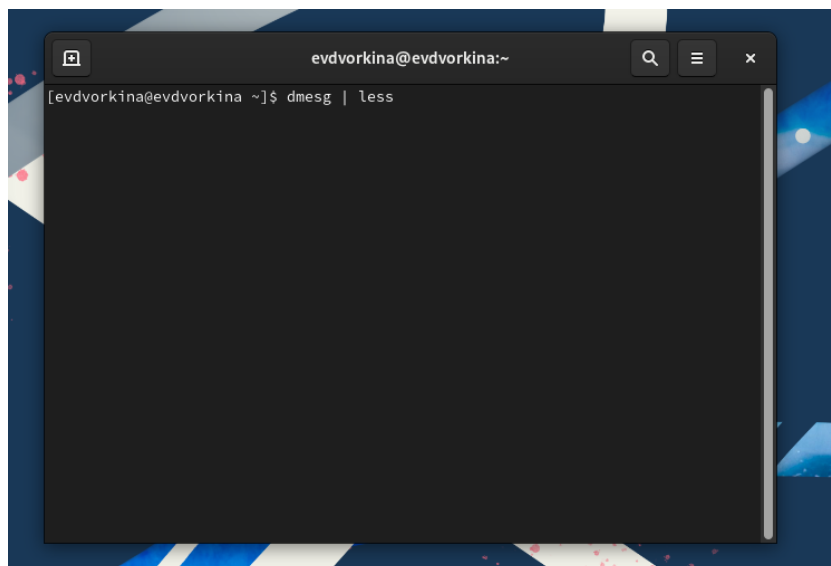


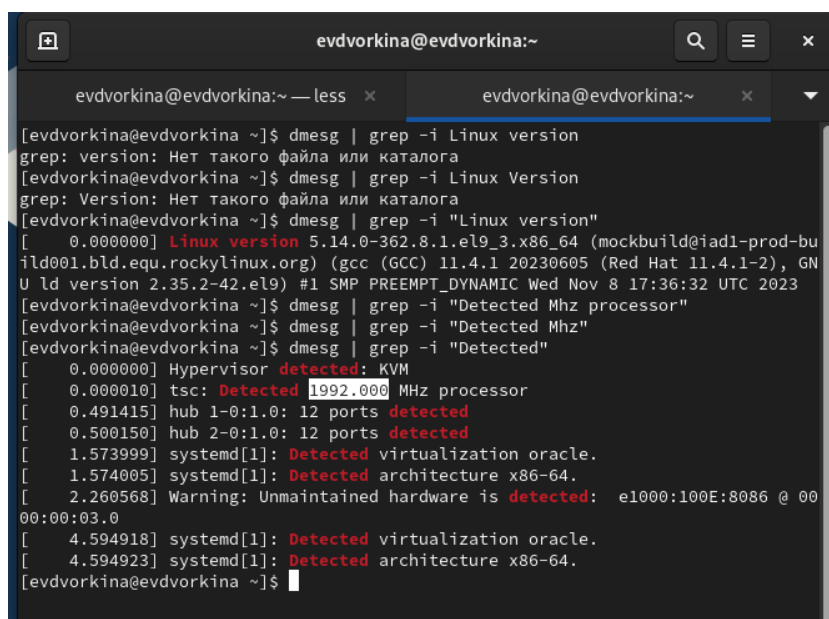
Рис. 4.1: Окно терминала

Версия ядра 5.14.0-362.8.1.el9_3.x86_64 (рис. 21).

```
[evdvorkina@evdvorkina ~]$ dmesg | grep -i "Linux version"
[    0.000000] Linux version 5.14.0-362.8.1.el9_3.x86_64 (mockbuild@iad1-prod-bu
ild001.bld.equ.rockylinux.org) (gcc (GCC) 11.4.1 20230605 (Red Hat 11.4.1-2), GN
U ld version 2.35.2-42.el9) #1 SMP PREEMPT_DYNAMIC Wed Nov 8 17:36:32 UTC 2023
[evdvorkina@evdvorkina ~]$
```

Рис. 4.2: Версия ядра

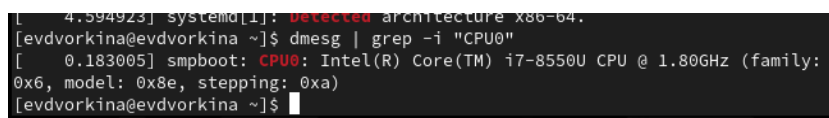
Частота процессора 1993 МГц (рис. 22).



```
evdvorkina@evdvorkina:~$ dmesg | grep -i Linux version
grep: version: Нет такого файла или каталога
evdvorkina@evdvorkina:~$ dmesg | grep -i Linux Version
grep: Version: Нет такого файла или каталога
evdvorkina@evdvorkina:~$ dmesg | grep -i "Linux version"
[ 0.000000] Linux version 5.14.0-362.8.1.el9_3.x86_64 (mockbuild@iad1-prod-bu
ild001.bld.equ.rockylinux.org) (gcc (GCC) 11.4.1 20230605 (Red Hat 11.4.1-2), GN
U ld version 2.35.2-42.el9) #1 SMP PREEMPT_DYNAMIC Wed Nov 8 17:36:32 UTC 2023
evdvorkina@evdvorkina:~$ dmesg | grep -i "Detected Mhz processor"
evdvorkina@evdvorkina:~$ dmesg | grep -i "Detected Mhz"
evdvorkina@evdvorkina:~$ dmesg | grep -i "Detected"
[ 0.000000] Hypervisor detected: KVM
[ 0.000010] tsc: Detected 1992.000 MHz processor
[ 0.491415] hub 1-0:1.0: 12 ports detected
[ 0.500150] hub 2-0:1.0: 12 ports detected
[ 1.573999] systemd[1]: Detected virtualization oracle.
[ 1.574005] systemd[1]: Detected architecture x86-64.
[ 2.260568] Warning: Unmaintained hardware is detected: e1000:100E:8086 @ 00
00:00:03.0
[ 4.594918] systemd[1]: Detected virtualization oracle.
[ 4.594923] systemd[1]: Detected architecture x86-64.
evdvorkina@evdvorkina:~$
```

Рис. 4.3: Частота процессора

Модель процессора Intel Core i7-8550U (рис. 23).



```
[ 4.594923] systemd[1]: Detected architecture x86-64.
evdvorkina@evdvorkina:~$ dmesg | grep -i "CPU0"
[ 0.183005] smpboot: CPU0: Intel(R) Core(TM) i7-8550U CPU @ 1.80GHz (family:
0x6, model: 0x8e, stepping: 0xa)
evdvorkina@evdvorkina:~$
```

Рис. 4.4: Модель процессора

Доступно 260860 Кб из 2096696 Кб (рис. 24).

```
[evdvorkina@evdvorkina ~]$ dmesg | grep -i "Memory:"
[ 0.003247] PM: hibernation: Registered nosave memory: [mem 0x00000000-0x00000fff]
[ 0.003249] PM: hibernation: Registered nosave memory: [mem 0x0009f000-0x0009ffff]
[ 0.003250] PM: hibernation: Registered nosave memory: [mem 0x000a0000-0x000effff]
[ 0.003250] PM: hibernation: Registered nosave memory: [mem 0x000f0000-0x000fffff]
[ 0.015632] Memory: 260860K/2096696K available (16384K kernel code, 5596K rwd
ata, 11444K rodata, 3824K init, 18424K bss, 158276K reserved, 0K cma-reserved)
[ 0.089223] Freeing SMP alternatives memory: 36K
[ 1.203111] Freeing initrd memory: 57244K
[ 1.460019] Freeing unused decrypted memory: 2036K
[ 1.460771] Freeing unused kernel image (initmem) memory: 3824K
[ 1.465494] Freeing unused kernel image (rodata/data gap) memory: 844K
[evdvorkina@evdvorkina ~]$
```

Рис. 4.5: Объем доступной оперативной памяти

Обнаруженный гипервизор типа KVM (рис. 25).

```
[evdvorkina@evdvorkina ~]$ dmesg | grep -i "Hypervisor"
[ 0.000000] Hypervisor detected: KVM
[ 0.073694] SRBDS: Unknown: Dependent on hypervisor status
[ 0.073695] GDS: Unknown: Dependent on hypervisor status
[evdvorkina@evdvorkina ~]$
```

Рис. 4.6: Тип обнаруженного гипервизора

`sudo fdisk -l` показывает тип файловой системы, типа Linux, Linux LVM (рис. 26).

```
[evdvorkina@evdvorkina ~]$ sudo fdisk -l

Мы полагаем, что ваш системный администратор изложил вам основы
безопасности. Как правило, всё сводится к трём следующим правилам:

№1) Уважайте частную жизнь других.
№2) Думайте, прежде что-то вводить.
№3) С большой властью приходит большая ответственность.

[sudo] пароль для evdvorkina:
Диск /dev/sda: 40 GiB, 42949672960 байт, 83886080 секторов
Disk model: VBOX HARDDISK
Единицы: секторов по 1 * 512 = 512 байт
Размер сектора (логический/физический): 512 байт / 512 байт
Размер I/O (минимальный/оптимальный): 512 байт / 512 байт
Тип метки диска: dos
Идентификатор диска: 0x00b40096

Устр-во    Загрузочный  начало    Конец    Секторы  Размер  Идентификатор  Тип
/dev/sda1  *            2048      2099199  2097152   1G      83 Linux
/dev/sda2            2099200  83886079  81786880  39G      8e Linux LVM
```

Рис. 4.7: Тип файловой системы

Далее показана последовательно монтирования файловых систем (рис. 27).

```

[evdvorkina@evdvorkina ~]$ dmesg | grep -i "Mount"
[ 0.070880] Mount-cache hash table entries: 4096 (order: 3, 32768 bytes, linear)
[ 0.070880] Mountpoint-cache hash table entries: 4096 (order: 3, 32768 bytes, linear)
[ 3.968701] XFS (dm-0): Mounting V5 Filesystem
[ 3.990946] XFS (dm-0): Ending clean mount
[ 5.087934] systemd[1]: Set up automount Arbitrary Executable File Formats File System Automount Point.
[ 5.103176] systemd[1]: Mounting Huge Pages File System...
[ 5.105646] systemd[1]: Mounting POSIX Message Queue File System...
[ 5.114903] systemd[1]: Mounting Kernel Debug File System...
[ 5.117063] systemd[1]: Mounting Kernel Trace File System...
[ 5.153426] systemd[1]: Starting Remount Root and Kernel File Systems...
[ 5.183994] systemd[1]: Mounted Huge Pages File System.
[ 5.184506] systemd[1]: Mounted POSIX Message Queue File System.
[ 5.184983] systemd[1]: Mounted Kernel Debug File System.
[ 5.185737] systemd[1]: Mounted Kernel Trace File System.
[ 5.196437] systemd[1]: Finished Remount Root and Kernel File Systems.
[ 5.200572] systemd[1]: Mounting FUSE Control File System...
[ 5.203467] systemd[1]: Mounting Kernel Configuration File System...
[ 5.204176] systemd[1]: OSTree Remount OS/ Bind Mounts was skipped because of an unmet condition check (ConditionKernelCommandLine=ostree).
[ 7.229376] XFS (sda1): Mounting V5 Filesystem
[ 7.564957] XFS (sda1): Ending clean mount
[evdvorkina@evdvorkina ~]$

```

Рис. 4.8: Последовательность монтирования файловых систем

5 Ответы на контрольные вопросы

1. Учетная запись содержит необходимые для идентификации пользователя при подключении к системе данные, а так же информацию для авторизации и учета: системного имени (user name) (оно может содержать только латинские буквы и знак нижнее подчеркивание, еще оно должно быть уникальным), идентификатор пользователя (UID) (уникальный идентификатор пользователя в системе, целое положительное число), идентификатор группы (GID) (группа, к к-рой относится пользователь. Она, как минимум, одна, по умолчанию - одна), полное имя (full name) (Могут быть ФИО), домашний каталог (home directory) (каталог, в к-рый попадает пользователь после входа в систему и в к-ром хранятся его данные), начальная оболочка (login shell) (командная оболочка, к-рая запускается при входе в систему).
2. Для получения справки по команде: `—help`; для перемещения по файловой системе - `cd`; для просмотра содержимого каталога - `ls`; для определения объёма каталога - `du` ; для создания / удаления каталогов - `mkdir/rmdir`; для создания / удаления файлов - `touch/rm`; для задания определённых прав на файл / каталог - `chmod`; для просмотра истории команд - `history`
3. Файловая система - это порядок, определяющий способ организации и хранения и именования данных на различных носителях информации. Примеры: FAT32 представляет собой пространство, разделенное на три части: одна область для служебных структур, форма указателей в виде таблиц и зона для хранения самих файлов. ext3/ext4 - журналируемая файловая система, используемая в основном в ОС с ядром Linux.

4. С помощью команды `df`, введя ее в терминале. Это утилита, которая показывает список всех файловых систем по именам устройств, сообщает их размер и данные о памяти. Также посмотреть подмонтированные файловые системы можно с помощью утилиты `mount`.
5. Чтобы удалить зависший процесс, вначале мы должны узнать, какой у него `id`: используем команду `ps`. Далее в терминале вводим команду `kill < id процесса >`. Или можно использовать утилиту `killall`, что “убьет” все процессы, которые есть в данный момент, для этого не нужно знать `id` процесса.

6 Выводы

Я приобрела практические навыки установки операционной системы на виртуальную машину, настройки минимально необходимых для дальнейшей работы сервисов.