

Презентация по лабораторной работе №1

Основы информационной безопасности

Татьяна Александровна Пономарева

2026-02-21

Содержание I

- 1 Информация
- 2 Вводная часть
- 3 Теоретическое введение
- 4 Выполнение лабораторной работы
- 5 Выполнение задания часть 1
- 6 Выводы

Раздел 1

Информация

Докладчик

- Пономарева Татьяна Александровна



Докладчик

- Пономарева Татьяна Александровна
- Студент группы НКАбд-04-24



Докладчик

- Пономарева Татьяна Александровна
- Студент группы НКАбд-04-24
- Российский университет дружбы народов им.
П. Лумумбы



Докладчик

- Пономарева Татьяна Александровна
- Студент группы НКАбд-04-24
- Российский университет дружбы народов им.
П. Лумумбы
- taponomareva6742@gmail.com



Докладчик

- Пономарева Татьяна Александровна
- Студент группы НКАбд-04-24
- Российский университет дружбы народов им.
П. Лумумбы
- taponomareva6742@gmail.com
- <https://github.com/taponomareva>



Раздел 2

Вводная часть

Цель работы

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

Задание

Настроить VirtualBox, установить соответствующие параметры для виртуальной машины, добавить образ диска гостевой ОС.

Раздел 3

Теоретическое введение

Теоретическое введение

Виртуализация - это технология, позволяющая запускать на одном физическом компьютере несколько независимых «виртуальных» систем (ВМ).

Гипервизор (VirtualBox) - программа, которая создает и управляет виртуальными машинами, распределяя ресурсы (процессор, память, диск) между ними.

Rocky Linux - дистрибутив, полностью совместимый с Red Hat Enterprise Linux (RHEL).
Оrientирован на стабильность и использование в серверных средах.

Раздел 4

Выполнение лабораторной работы

Запуск VirtualBox

Сначала открываем VirtualBox (рис. 1).

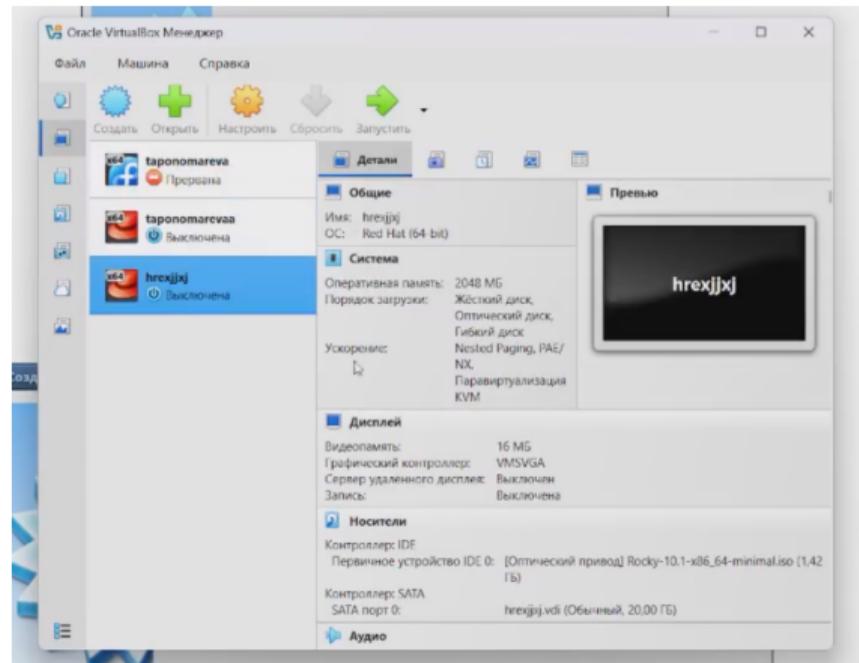


Рисунок 1: Открытая программа VirtualBox

Создание виртуальной машины

Затем создаем виртуальную машину с именем `taponomareva1`, задаем папку расположения и образ ISO (рис. 2).

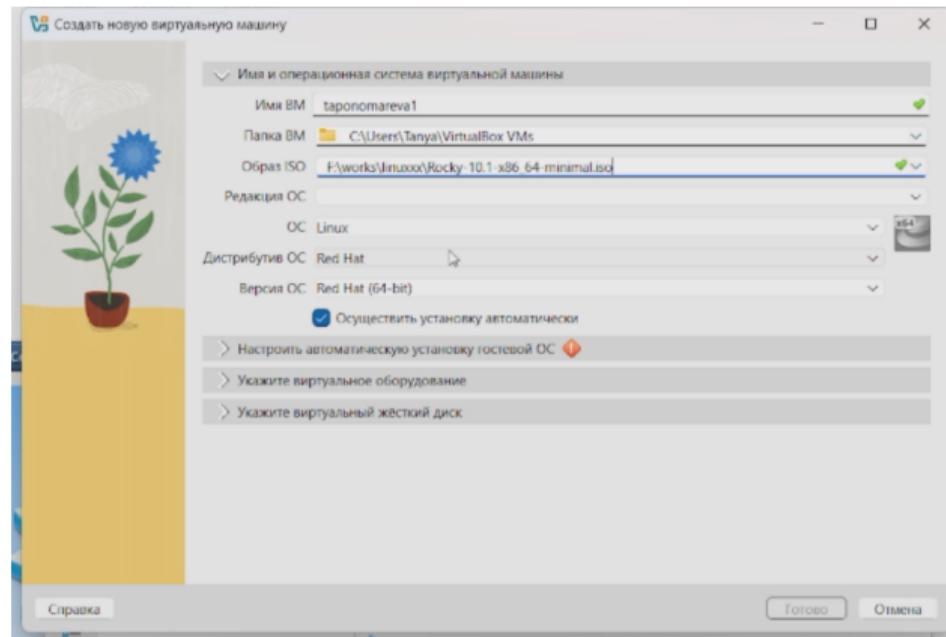


Рисунок 2: Настройка виртуальной машины

Настройка параметры гостевой ОС

Потом настраиваем параметры гостевой ОС, задаем имя пользователя и пароль (рис. 3).

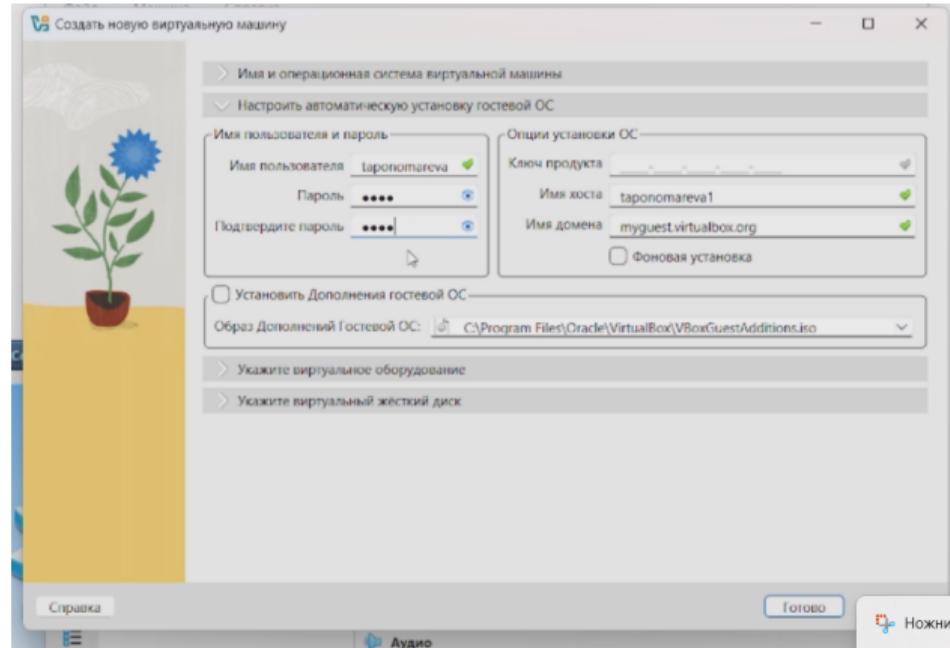


Рисунок 3: Настройка параметров гостевой ОС

Настройка виртуального оборудования

Указываем виртуальное оборудование (рис. 4).

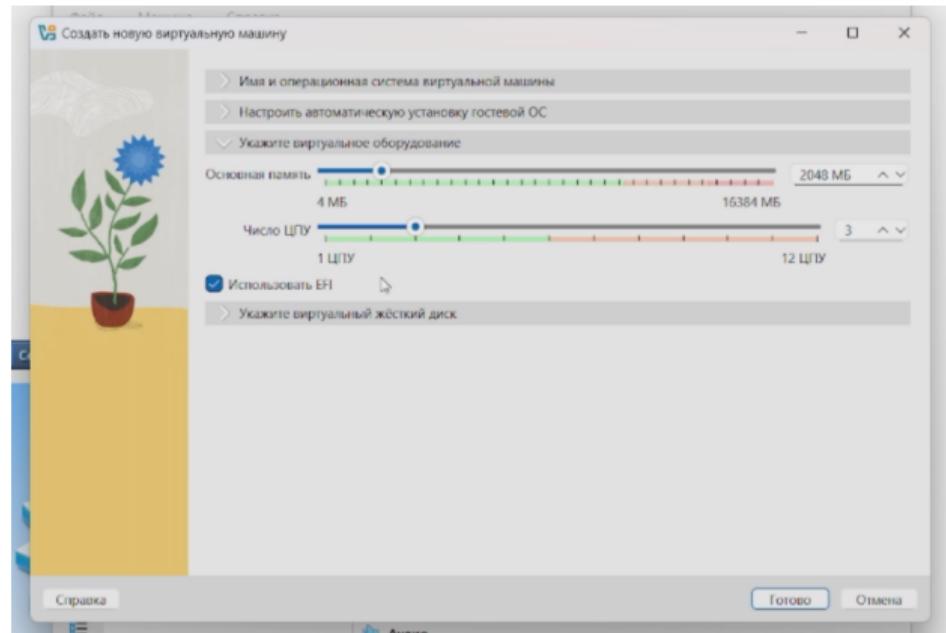


Рисунок 4: Настройка виртуального оборудования

Указание виртуального жесткого диска

Указываем виртуальный жесткий диск в 40,08 Гб (рис. 5).

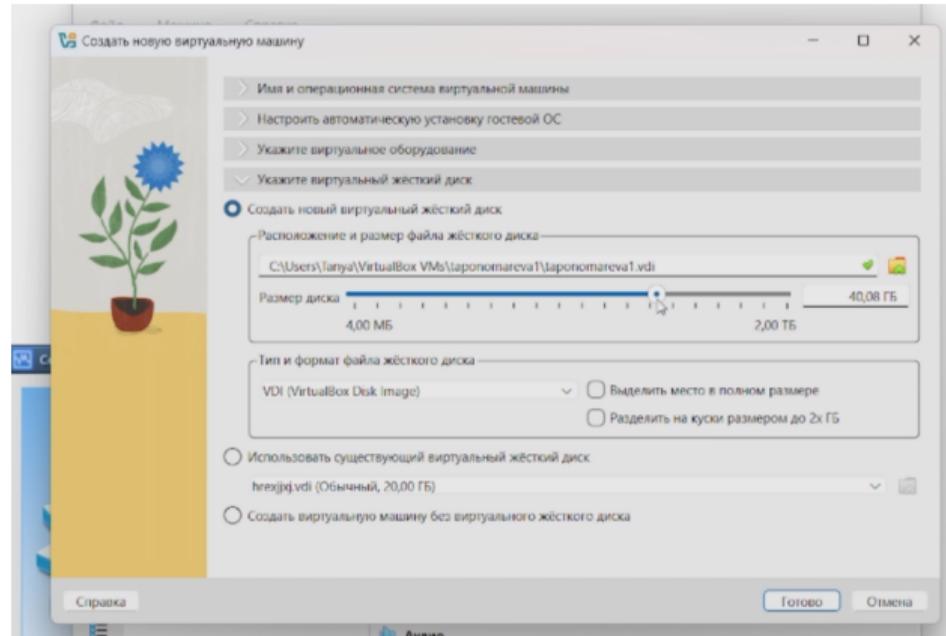


Рисунок 5: Настройка виртуального жесткого диска

Общие настройки

Проверяем общие настройки на правильность (рис. 6).

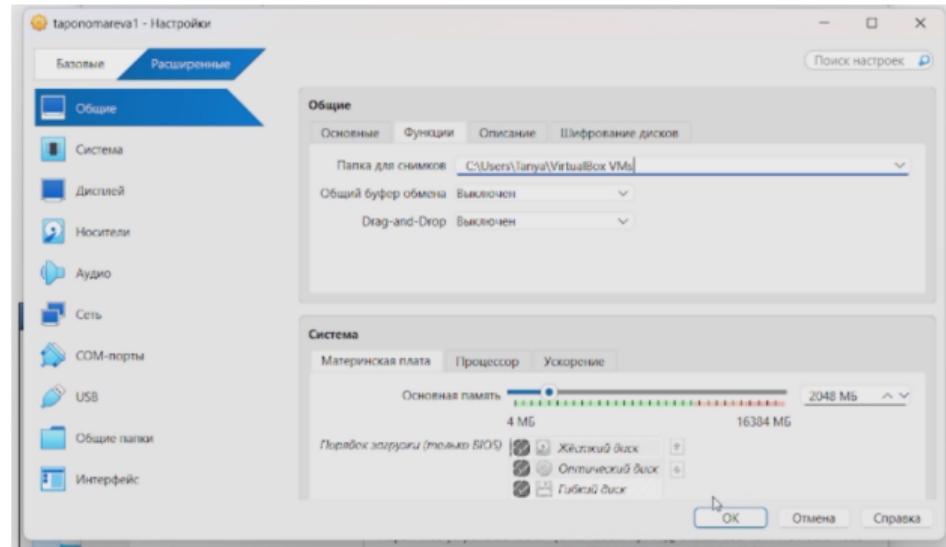


Рисунок 6: Общие настройки

Запуск виртуальной машины

Запускаем виртуальную машину (рис. 7).

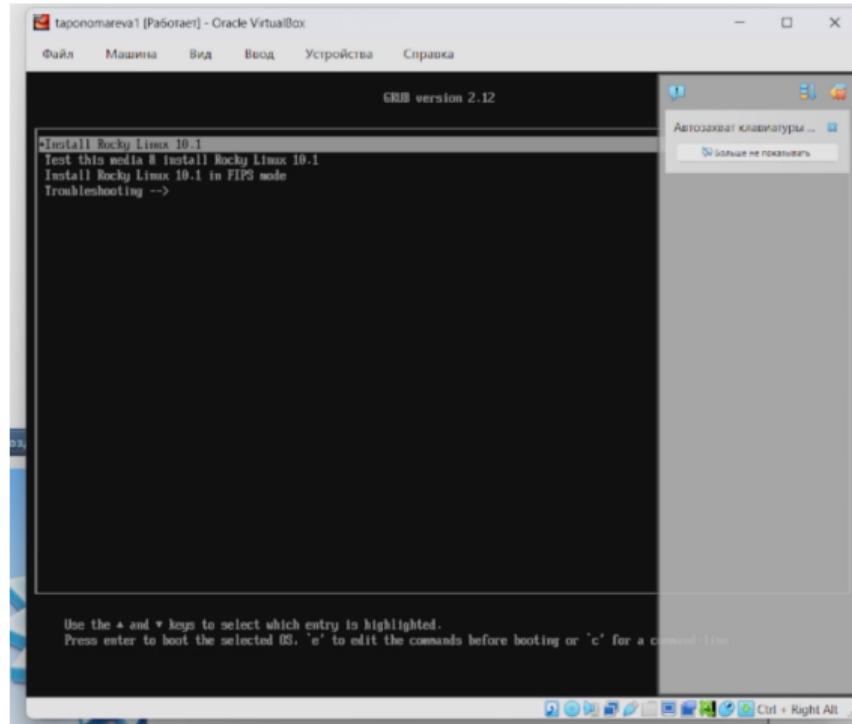


Рисунок 7: Запуск виртуальной машины

Установка Rocky Linux

Начинаем устанавливать Rocky Linux (рис. 8).

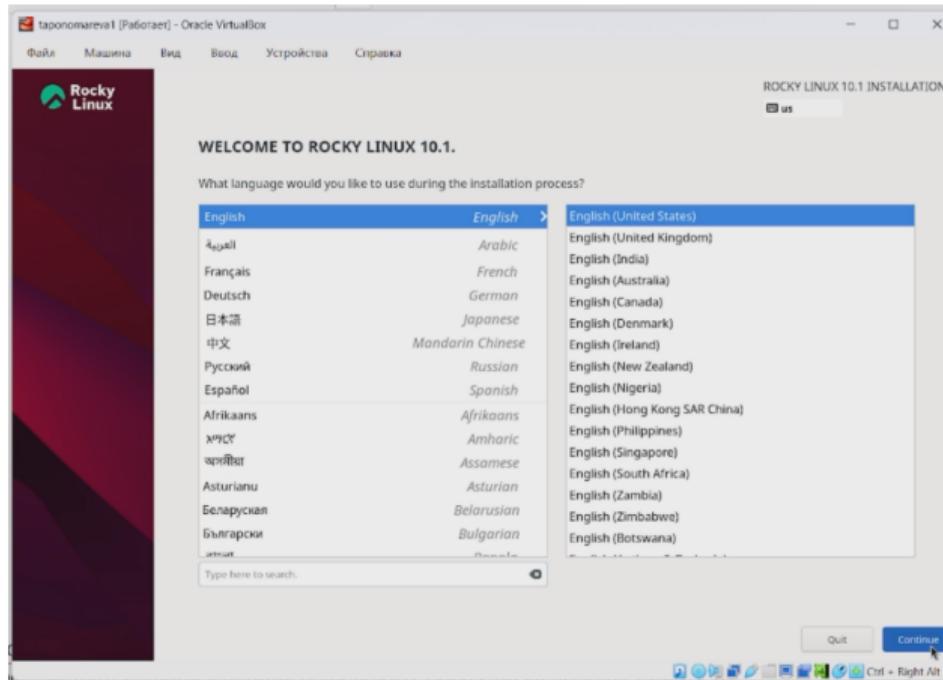


Рисунок 8: Установка Rocky Linux. Настройка языка

Выбор виртуального жесткого диска

Выбираем виртуальный жесткий диск для установки (рис. 9).

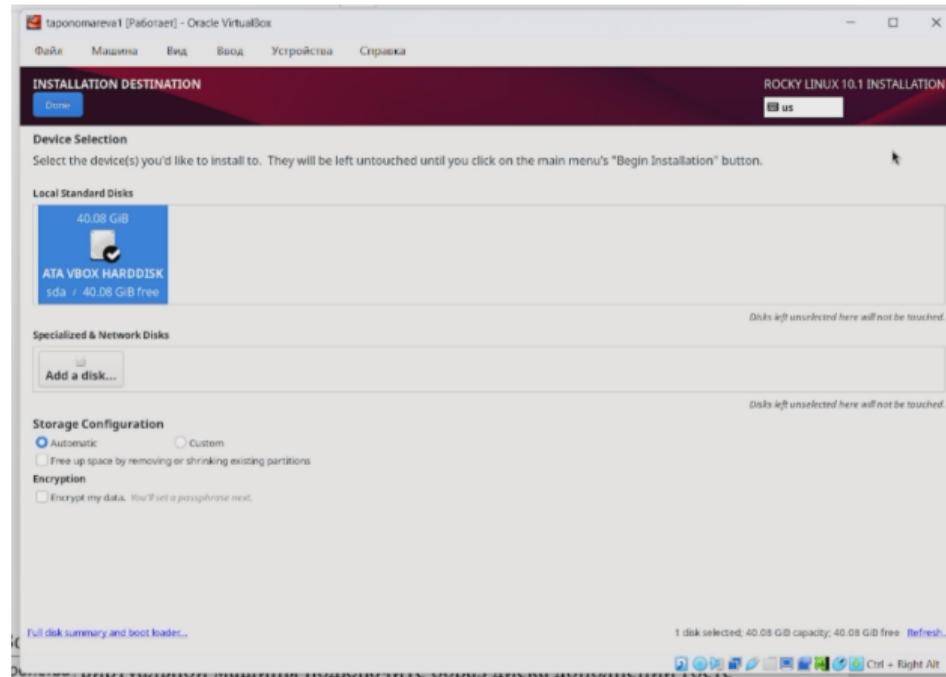


Рисунок 9: Выбор жесткого диска

Задание root пользователя

Задаем root пользователя и пароль (рис. 10).

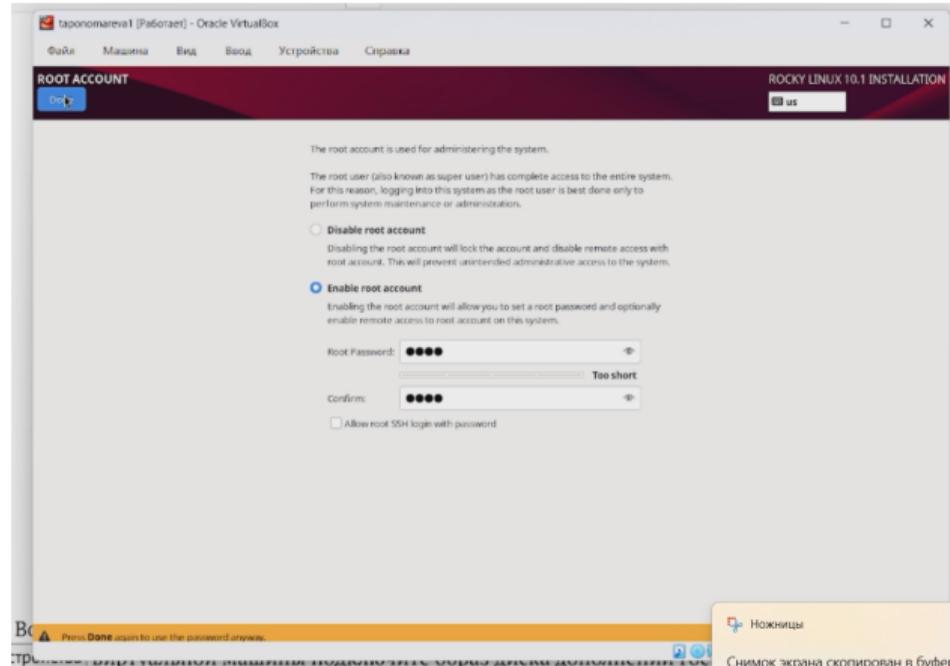


Рисунок 10: Настройка root пользователя и пароля

Создание системного пользователя и пароля

Создаем пользователя системы, через которого будет происходить вход в виртуальную среду по паролю (рис. 11).

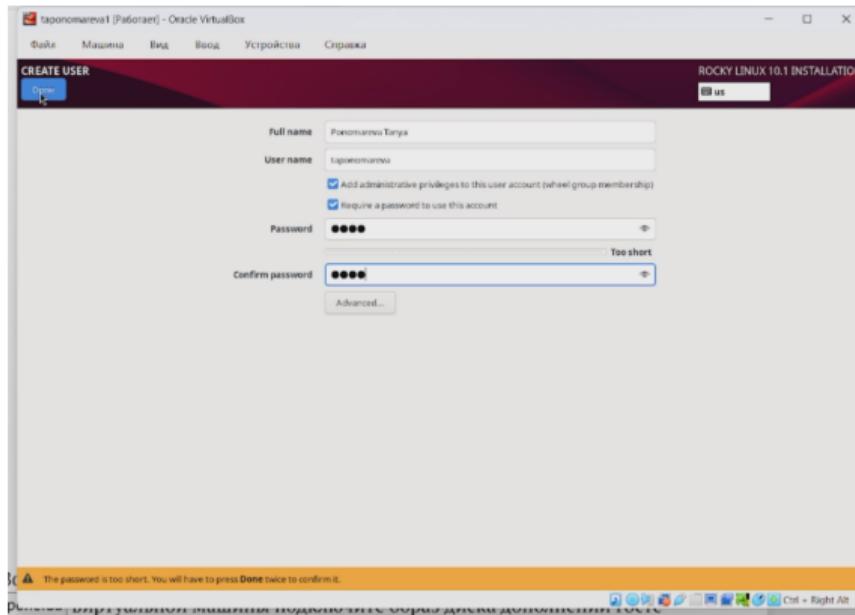


Рисунок 11: Системный пользователь и пароль

Запуск системы

Запускаем установку системы (рис. 12).

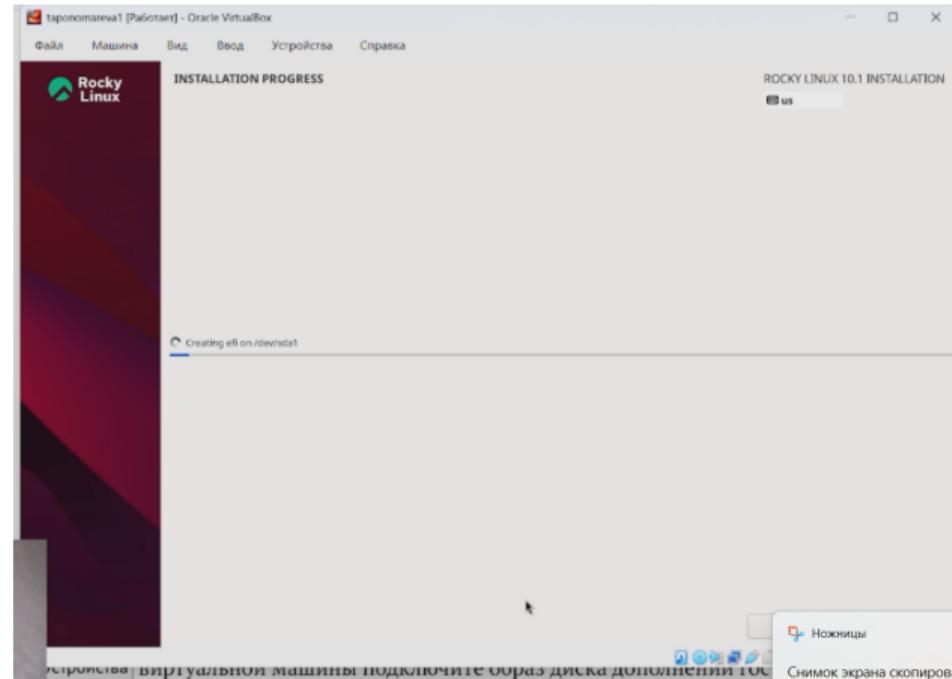


Рисунок 12: Начало установки

Конец установки Rocky Linux

Rocky Linux был установлен (рис. 13).

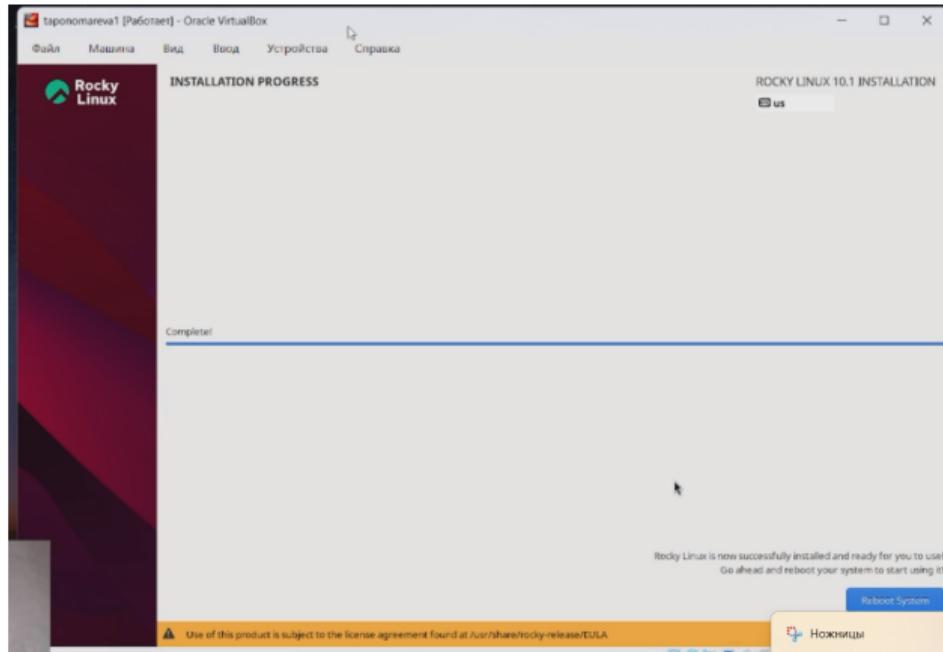


Рисунок 13: Конец установки

Установка графического интерфейса

Устанавливаем графический интерфейс (рис. 14).

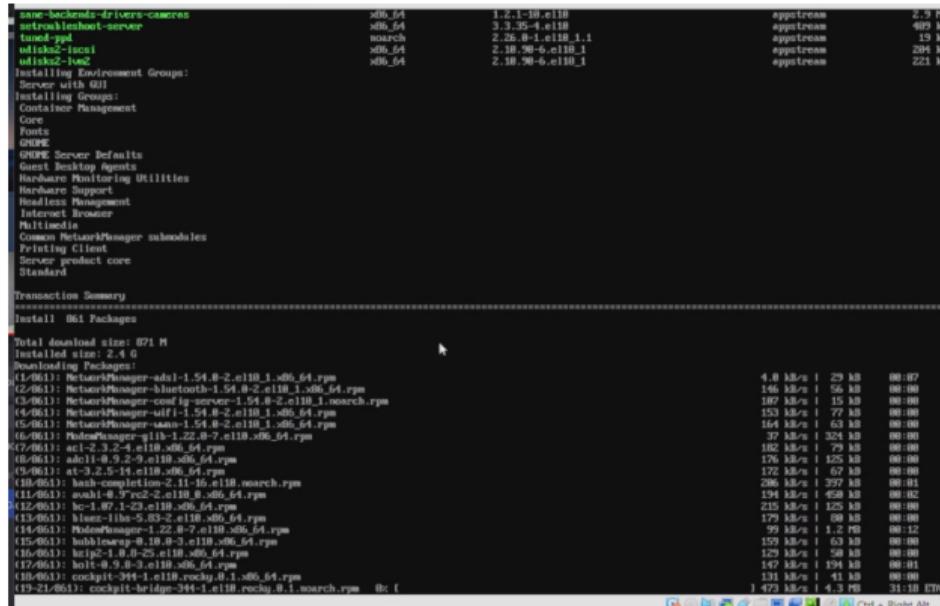


Рисунок 14: Начало установки графического интерфейса

Рабочий стол

Графический интерфейс был установлен (рис. 15).

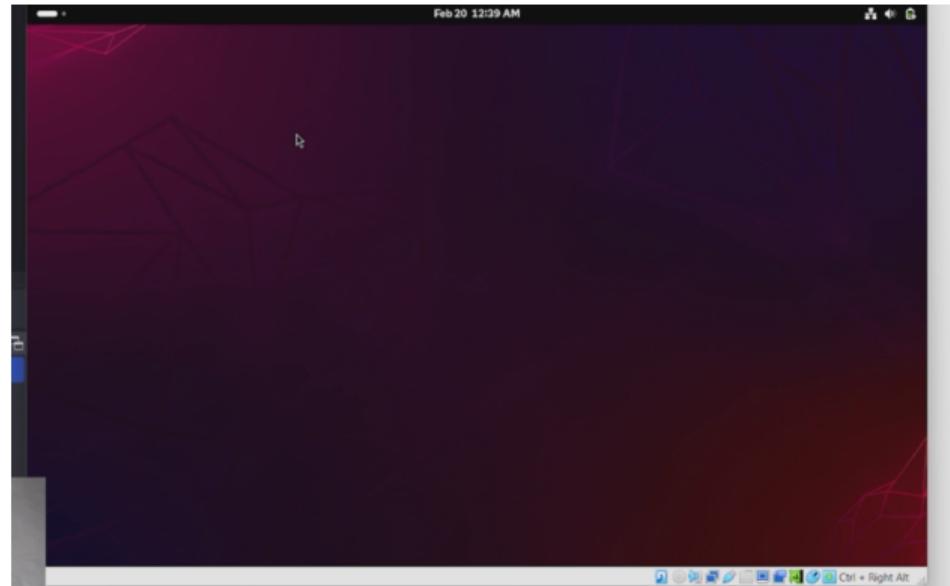


Рисунок 15: Рабочий стол

Подключение образа гостевой ОС

Затем подключаем образ гостевой ОС и изымаем диск из виртуального привода (рис. 16).

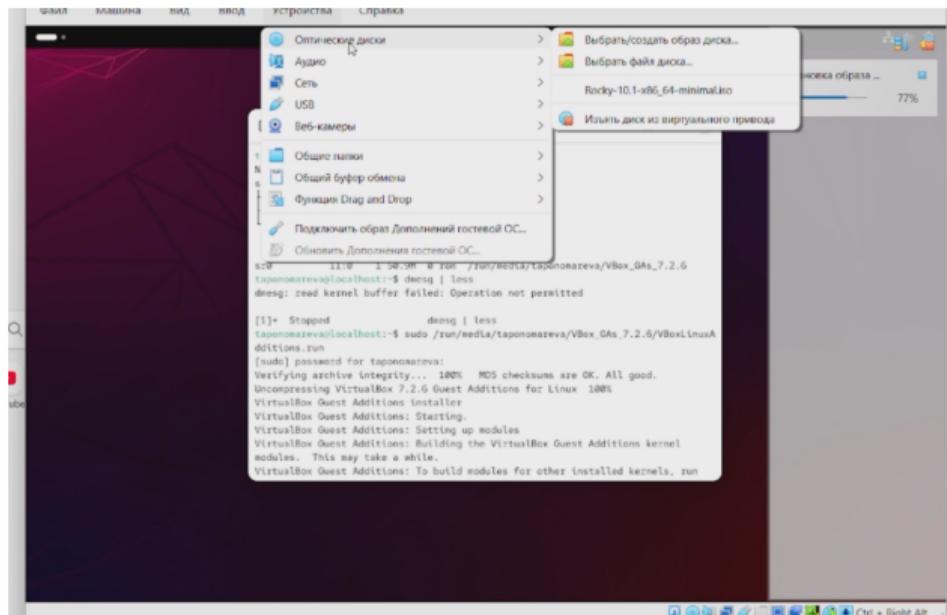


Рисунок 16: Подключение образа гостевой ОС и изъятие диска из виртуального привода

Настройка мыши

Настраиваем, чтобы мышь могла выходить за пределы окна VirtualBox (рис. 17).

```
taponomareva@localhost:~$ sudo /run/media/taponomareva/VBox_GAs_7.2.6/VBoxLinuxA
dditions.run
[sudo] password for taponomareva:
Verifying archive integrity... 100%   MD5 checksums are OK. All good.
Uncompressing VirtualBox 7.2.6 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
```

Рисунок 17: Настройка мыши

Замена localhost

Меняем localhost на имя пользователя(рис. 18).

```
taponomareva@localhost:~$ sudo hostnamectl set-hostname taponomareva  
taponomareva@localhost:~$ reboot
```

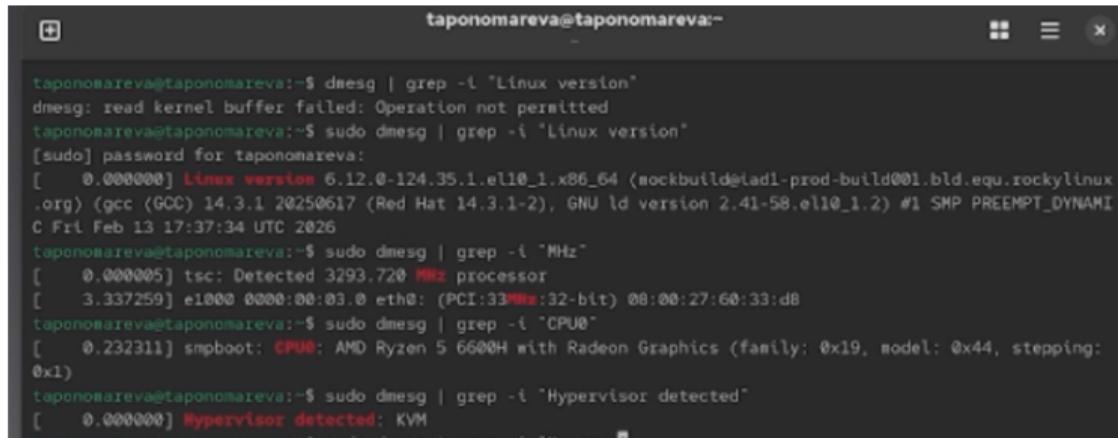
Рисунок 18: Замена localhost на taponomareva

Раздел 5

Выполнение задания часть 1

Выполнение задания часть 1

Выполняем задания из лабораторной работы по компонентам поставленной системы, т.е узнаем версию ядра, частоту процессора, модель процессора, тип обнаруженного гипервизора (рис. 19).



The screenshot shows a terminal window titled "taponomareva@taponomareva:~". The user runs several commands to gather system information:

```
taponomareva@taponomareva:~$ dmesg | grep -i "Linux version"
dmesg: read kernel buffer failed: Operation not permitted
taponomareva@taponomareva:~$ sudo dmesg | grep -i "Linux version"
[sudo] password for taponomareva:
[    0.000000] Linux version 6.12.0-124.35.1.el8_1.x86_64 (mockbuild@iadi-prod-build001.bld.equ.rockylinux.org) (gcc (GCC) 14.3.1 20250617 (Red Hat 14.3.1-2), GNU ld version 2.41-58.el8_1.2) #1 SMP PREEMPT_DYNAMIC
C Fri Feb 13 17:37:34 UTC 2026
taponomareva@taponomareva:~$ sudo dmesg | grep -i "MHz"
[    0.000005] tsc: Detected 3293.720 MHz processor
[  3.337259] e1000 0000:00:00:03.0 eth0: (PCI:33MHz:32-bit) 00:00:27:60:33:d8
taponomareva@taponomareva:~$ sudo dmesg | grep -i "CPU0"
[  0.232311] smpboot: CPU0: AMD Ryzen 5 6600H with Radeon Graphics (family: 0x19, model: 0x44, stepping: 0x1)
taponomareva@taponomareva:~$ sudo dmesg | grep -i "Hypervisor detected"
[    0.000000] Hypervisor detected: KVM
```

Рисунок 19: Выполнение задания часть 1

Выполнение задания часть 2

Смотрим, какой объем оперативной памяти доступен (рис. 20).

```
taponovareva@taponovareva:~$ sudo dmesg | grep -i "Memory"
[ 0.074091] PM: hibernation: Registered nosave memory: [mem 0x00000000-0x00000fff]
[ 0.074092] PM: hibernation: Registered nosave memory: [mem 0x000a0000-0x000fffff]
[ 0.074093] PM: hibernation: Registered nosave memory: [mem 0x7cf5000-0x7cfe5fff]
[ 0.074094] PM: hibernation: Registered nosave memory: [mem 0x7daa3000-0x7dbb6fff]
[ 0.074094] PM: hibernation: Registered nosave memory: [mem 0x7e5ed000-0x7e96cff]
[ 0.074095] PM: hibernation: Registered nosave memory: [mem 0x7e96d000-0x7e97eff]
[ 0.074095] PM: hibernation: Registered nosave memory: [mem 0x7e97f000-0x7e9fefff]
[ 0.074096] PM: hibernation: Registered nosave memory: [mem 0x7f125000-0x7f128fff]
[ 0.074096] PM: hibernation: Registered nosave memory: [mem 0x7f129000-0x7f12aff]
[ 0.123367] Freeing SMP alternatives memory: 40K
[ 0.244680] Memory: 1633920K/2077572K available (18432K kernel code, 5805K rwdta, 14280K rodata, 4348K
init, 6684K bss, 409504K reserved, 0K cma-reserved)
[ 0.831664] Freeing initrd memory: 52076K
[ 1.253270] Freeing unused decrypted memory: 2028K
[ 1.255426] Freeing unused kernel image (initmem) memory: 4348K
[ 1.258132] Freeing unused kernel image (rodata/data gap) memory: 56K
```

Рисунок 20: Выполнение задания часть 2

Выполнение задания часть 3

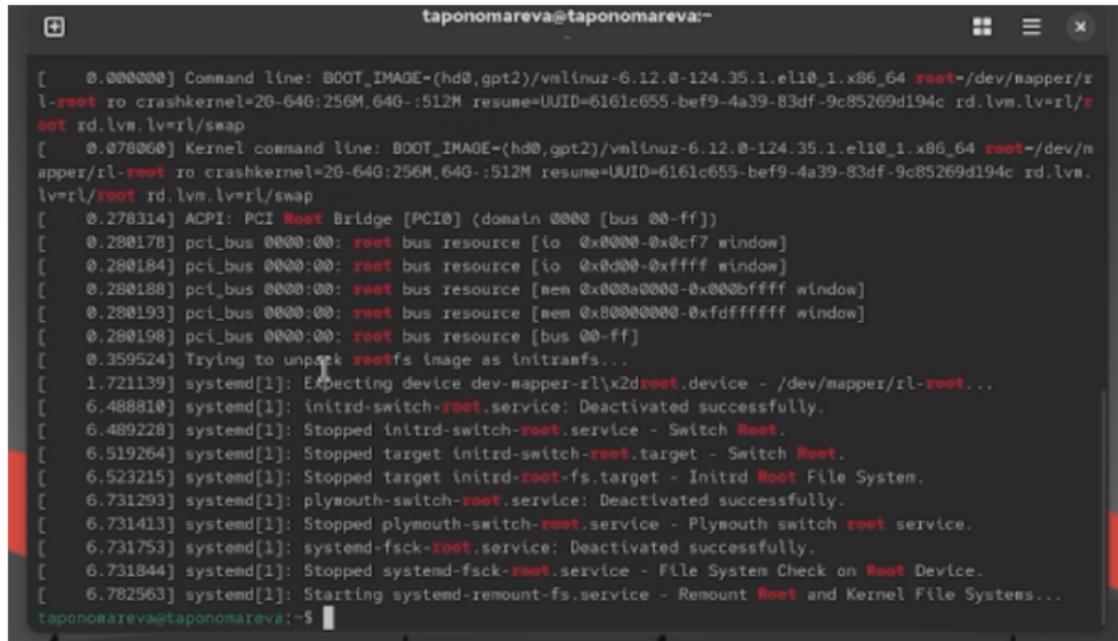
Далее, смотрим на тип файловой системы корневого раздела (рис. 21).

```
tarponareva@tarponareva:~$ sudo dmesg | grep -i "root"
[    0.000000] Command line: BOOT_IMAGE=(hd0,gpt2)/vmlinuz-6.12.0-124.35.1.el10_1.x86_64 root=/dev/mapper/r
l-root ro crashkernel=2G-64G:256M,64G-:512M resume=UUID=6161c655-beff9-4a39-83df-9c85269d194c rd.lvm.lv=rl/r
oot rd.lvm.lv=rl/swap
[    0.078060] Kernel command line: BOOT_IMAGE=(hd0,gpt2)/vmlinuz-6.12.0-124.35.1.el10_1.x86_64 root=/dev/m
apper/rl-root ro crashkernel=2G-64G:256M,64G-:512M resume=UUID=6161c655-beff9-4a39-83df-9c85269d194c rd.lvm.
lv=rl/root rd.lvm.lv=rl/swap
[    0.278314] ACPI: PCI Root Bridge [PCI0] (domain 0000 [bus 00-ff])
[    0.280178] pci_bus 0000:00: root bus resource [io 0x0000-0x0cf7 window]
[    0.280184] pci_bus 0000:00: root bus resource [io 0xd00-0xffff window]
[    0.280188] pci_bus 0000:00: root bus resource [mem 0x00a0000-0x000bffff window]
[    0.280193] pci_bus 0000:00: root bus resource [mem 0x80000000-0xfdffff window]
[    0.280198] pci_bus 0000:00: root bus resource [bus 00-ff]
[    0.359524] Trying to unpack rootfs image as initramfs...
[    1.721139] systemd[1]: Expecting device dev-mapper-rl\x2droot.device - /dev/mapper/rl-root...
[    6.488810] systemd[1]: initrd-switch-root.service: Deactivated successfully.
```

Рисунок 21: Выполнение задания часть 3.0

Выполнение задания часть 3.5

Далее, продолжаем смотреть на тип файловой системы корневого раздела (рис. 22).



The screenshot shows a terminal window with the title "taponomareva@taponomareva:~". The window displays a series of log messages from the kernel and systemd during the boot process. The messages are color-coded in red and black. Key text visible in the log includes:

- [0.000000] Command line: BOOT_IMAGE=(hd0,gpt2)/vmlinuz-6.12.0-124.35.1.el10_1.x86_64 root=/dev/mapper/r1-root ro crashkernel=2G-64G:256M,64G-:512M resume=UUID=6161c655-bef9-4a39-83df-9c85269d194c rd.lvm.lv=r1-root rd.lvm.lv=r1/swap
- [0.078060] Kernel command line: BOOT_IMAGE=(hd0,gpt2)/vmlinuz-6.12.0-124.35.1.el10_1.x86_64 root=/dev/mapper/r1-root ro crashkernel=2G-64G:256M,64G-:512M resume=UUID=6161c655-bef9-4a39-83df-9c85269d194c rd.lvm.lv=r1/root rd.lvm.lv=r1/swap
- [0.278314] ACPI: PCI Root Bridge [PCI0] (domain 0000 [bus 00-ff])
- [0.280178] pci_bus 0000:00: root bus resource [io 0x0000-0x0cf7 window]
- [0.280184] pci_bus 0000:00: root bus resource [io 0xd00-0xffff window]
- [0.280188] pci_bus 0000:00: root bus resource [mem 0x000a0000-0x000bffff window]
- [0.280193] pci_bus 0000:00: root bus resource [mem 0x80000000-0xfdffffff window]
- [0.280198] pci_bus 0000:00: root bus resource [bus 00-ff]
- [0.359524] Trying to unpack rootfs image as initramfs...
- [1.721139] systemd[1]: Expecting device dev-mapper-r1\x2droot.device - /dev/mapper/r1-root...
- [6.488810] systemd[1]: initrd-switch-root.service: Deactivated successfully.
- [6.489228] systemd[1]: Stopped initrd-switch-root.service - Switch Root.
- [6.519264] systemd[1]: Stopped target initrd-switch-root.target - Switch Root.
- [6.523215] systemd[1]: Stopped target initrd-root-fs.target - Initrd Root File System.
- [6.731293] systemd[1]: plymouth-switch-root.service: Deactivated successfully.
- [6.731413] systemd[1]: Stopped plymouth-switch-root.service - Plymouth switch root service.
- [6.731753] systemd[1]: systemd-fsck-root.service: Deactivated successfully.
- [6.731844] systemd[1]: Stopped systemd-fsck-root.service - File System Check on Root Device.
- [6.782563] systemd[1]: Starting systemd-remount-fs.service - Remount Root and Kernel File Systems...

Рисунок 22: Выполнение задания часть 3.5

Выполнение задания часть 4

Наконец, выводим последовательность монтирования файловых систем (рис. 23).

```
taponomareva@taponomareva:~$ sudo dmesg | grep -i "mounted"
[    6.857265] systemd[1]: Mounted dev-hugepages.mount - Huge Pages File System.
[    6.868524] systemd[1]: Mounted dev-mqueue.mount - POSIX Message Queue File System.
[    6.869952] systemd[1]: Mounted sys-kernel-debug.mount - Kernel Debug File System.
[    6.871457] systemd[1]: Mounted sys-kernel-tracing.mount - Kernel Trace File System.
taponomareva@taponomareva:~$
```

Рисунок 23: Выполнение задания часть 4

Раздел 6

Выводы

Выводы

При проведении лабораторной работы были приобретены практические навыки установки операционной системы на виртуальную машину, настройки минимально необходимых для дальнейшей работы сервисов.