

Лабораторная работа №1

**Установка и конфигурация операционной системы на виртуальную
машину**

Дарья Эдуардовна Ибатулина

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	7
4	Выполнение лабораторной работы	8
5	Выводы	20
6	Ответы на контрольные вопросы	21
	Список литературы	23

Список иллюстраций

4.1	Загрузка Linux Rocky	8
4.2	Интерфейс VirtualBox	9
4.3	Папка для виртуальных машин	9
4.4	Создание новой машины	10
4.5	Настройка памяти и числа процессоров	10
4.6	Создание нового виртуального жёсткого диска	11
4.7	Подключение образа диска	11
4.8	Запуск установки Linux Rocky	12
4.9	Выбор языка установки	12
4.10	Отключение kdump	13
4.11	Выбор программ	13
4.12	Установка пароля root	13
4.13	Выбор устройств установки ОС	14
4.14	Настройка сети и имени узла	14
4.15	Обзор параметров	15
4.16	Завершение установки	15
4.17	Интерфейс Linux Rocky	16
4.18	Подключение образа диска дополнений гостевой ОС	16
4.19	Вывод команды dmesg	17
4.20	Версия ядра Linux	17
4.21	Частота процессора	17
4.22	Модуль процессора	18
4.23	Объём доступной оперативной памяти	18
4.24	Тип обнаруженного гипервизора	18
4.25	Тип файловой системы корневого раздела	19
4.26	Последовательность монтирования файловых систем	19

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

Установить на виртуальную машину VirtualBox дистрибутив Linux Rocky.

3 Теоретическое введение

В данном разделе я представляю краткое определение понятия *виртуальные машины* и рассказываю о том, что такое Rocky Linux.

Rocky Linux — дистрибутив Linux, разработанный Rocky Enterprise Software Foundation. Предполагается, что это будет полный бинарно-совместимый выпуск, использующий исходный код операционной системы Red Hat Enterprise Linux (RHEL). Цель проекта — предоставить корпоративную операционную систему производственного уровня, поддерживаемую сообществом. Rocky Linux, наряду с Red Hat Enterprise Linux и SUSE Linux Enterprise, стала популярной для использования в корпоративных операционных системах.

Виртуальные машины (ВМ) — это такие абстрактные вычислительные экземпляры, созданные программой, работающей на другой машине, и которые физически не существуют, но работают, как настоящий компьютер. ВМ работает на программном обеспечении, а не на оборудовании. Так сказать, программное обеспечение создаёт «воображаемый» компьютер, который ведёт себя как физический. Другими словами, пользователь создаёт компьютер внутри другого компьютера. Несколько ВМ могут работать параллельно на одном базовом устройстве отдельно друг от друга.

4 Выполнение лабораторной работы

Для начала переходим на официальный сайт Linux Rocky и устанавливаем образ операционной системы на компьютер (рис. [4.1]).

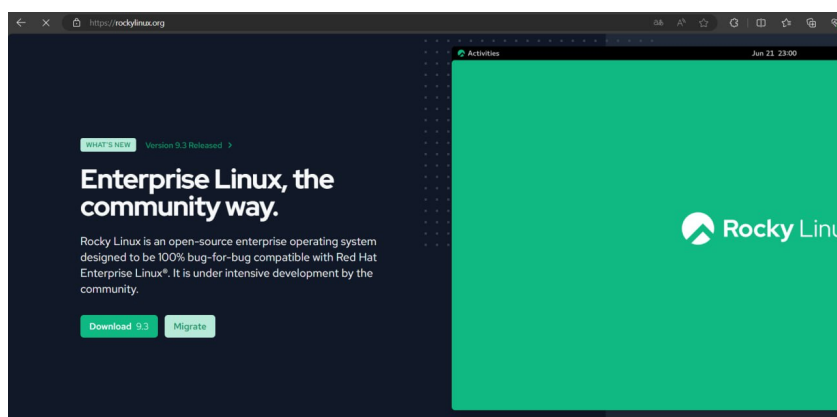


Рис. 4.1: Загрузка Linux Rocky

Для выполнения лабораторной работы понадобится VirtualBox. В программе можно создавать виртуальные машины. Она осталась у меня с прошлого года (рис. [4.2]).

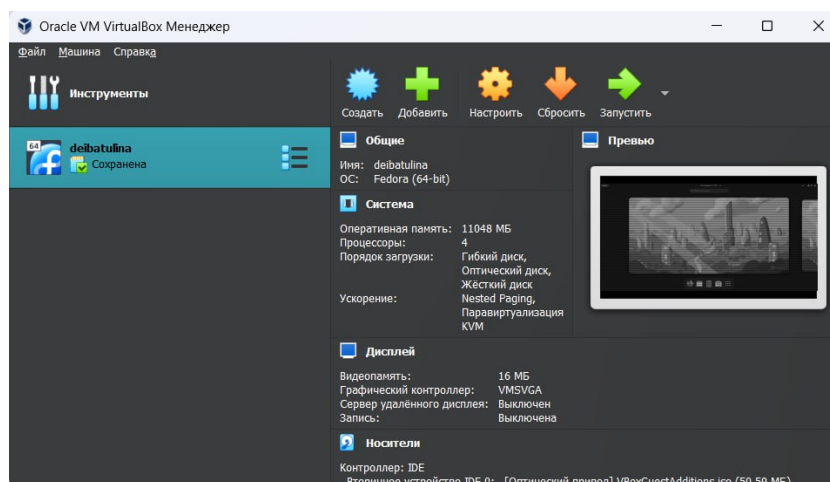


Рис. 4.2: Интерфейс VirtualBox

Теперь выбираем папку для виртуальных машин. Поскольку я выполняю работу на своей технике, то и каталог могу выбрать на своё усмотрение (рис. [4.3]).

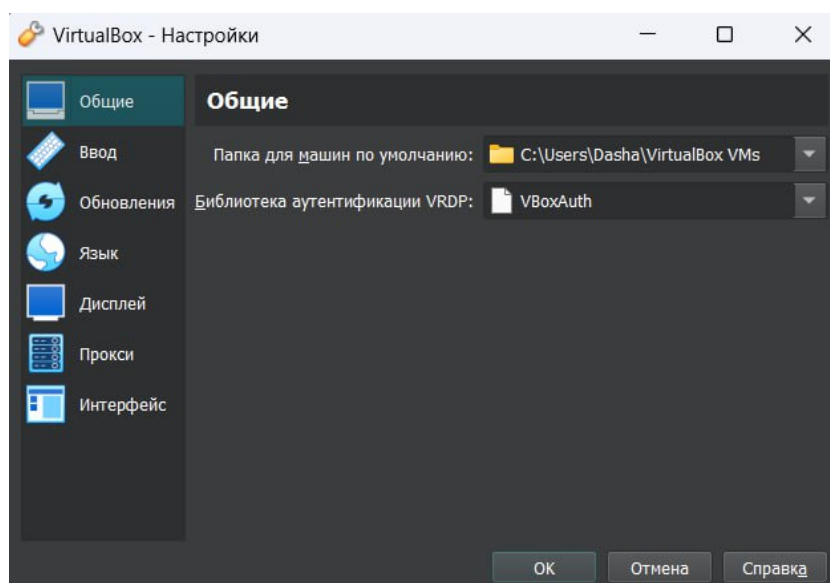


Рис. 4.3: Папка для виртуальных машин

Создаём новую машину. Даём ей имя (как логин в дисплейном классе по соглашению об именовании), выбираем папку для виртуальной машины, тип и версию ОС (рис. [4.4]).

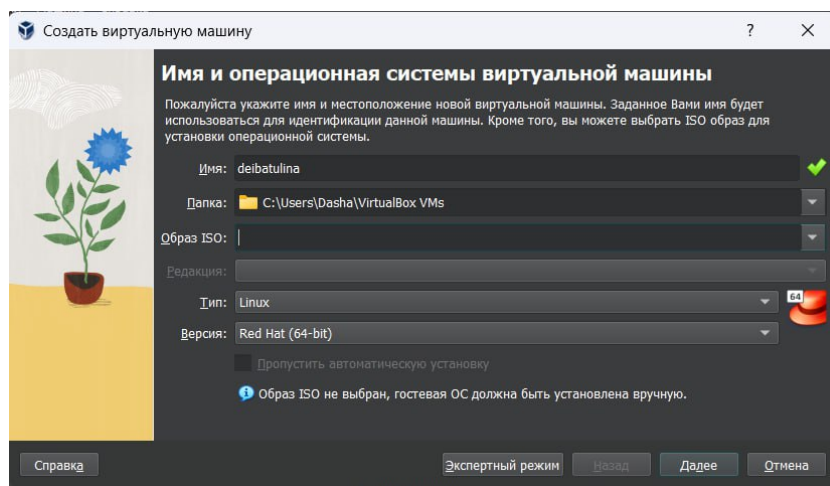


Рис. 4.4: Создание новой машины

Выбираем объём основной памяти и число процессоров (я выбрала почти максимально возможные значения, чтобы виртуальная машина не тормозила, и с ней было легко и комфортно работать) (рис. [4.5]).

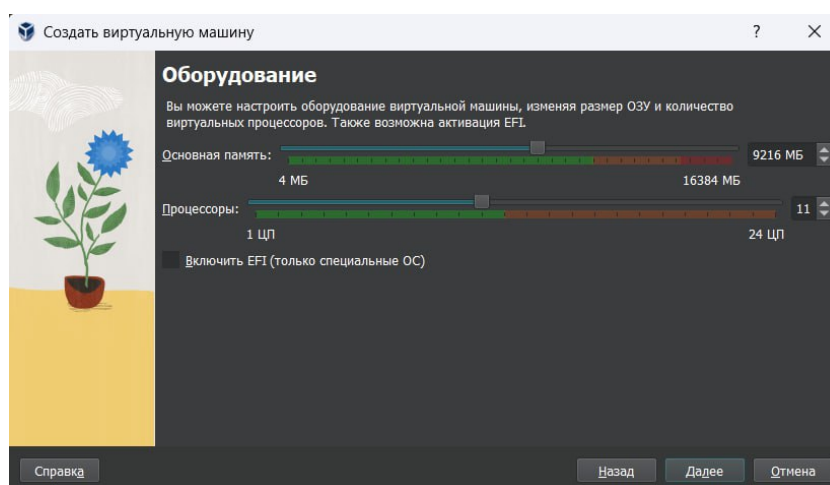


Рис. 4.5: Настройка памяти и числа процессоров

Затем создаём новый виртуальный жёсткий диск и задаём его размер (рис. [4.6]).

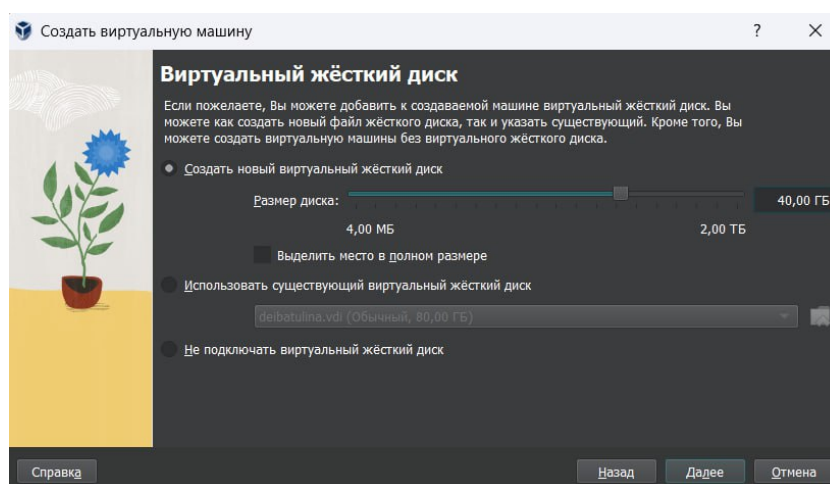


Рис. 4.6: Создание нового виртуального жёсткого диска

В пункте “Носители” выбираем нужный нам образ диска (это тот, что я скачала в самом начале работы) (рис. [4.7]).

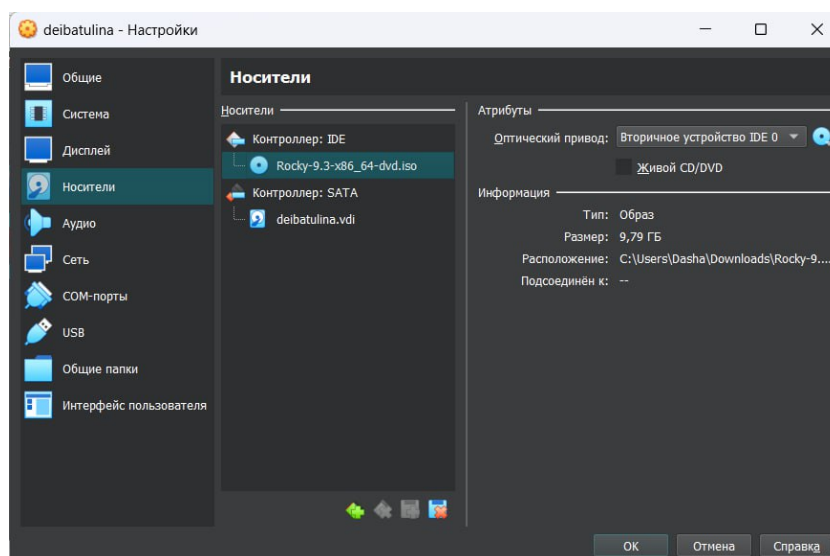


Рис. 4.7: Подключение образа диска

Запускается процесс установки Linux Rocky (рис. [4.8]).

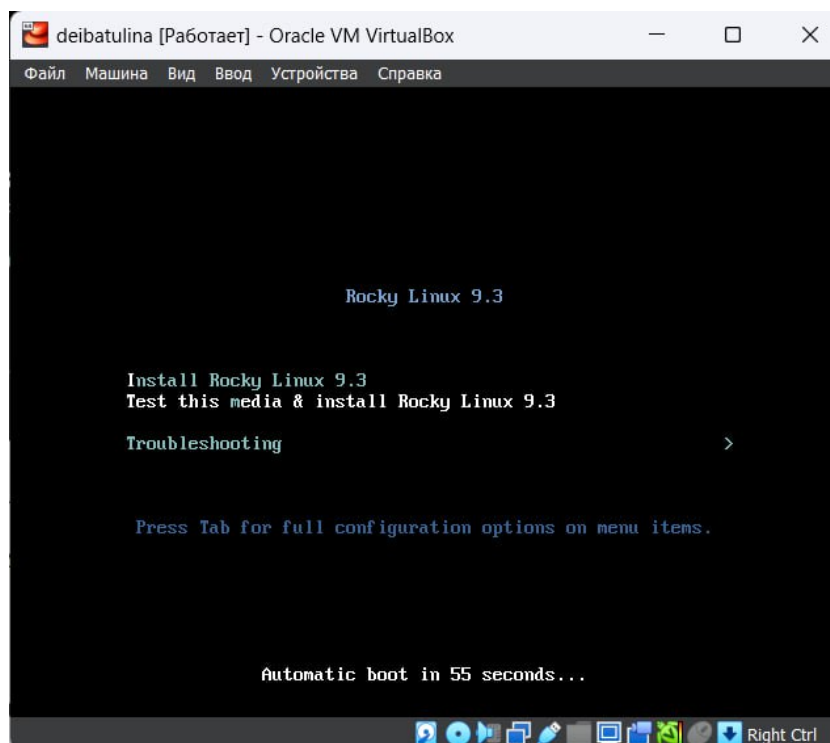


Рис. 4.8: Запуск установки Linux Rocky

Теперь выбираем язык, который будет использоваться в процессе установки (Русский) (рис. [4.9]).

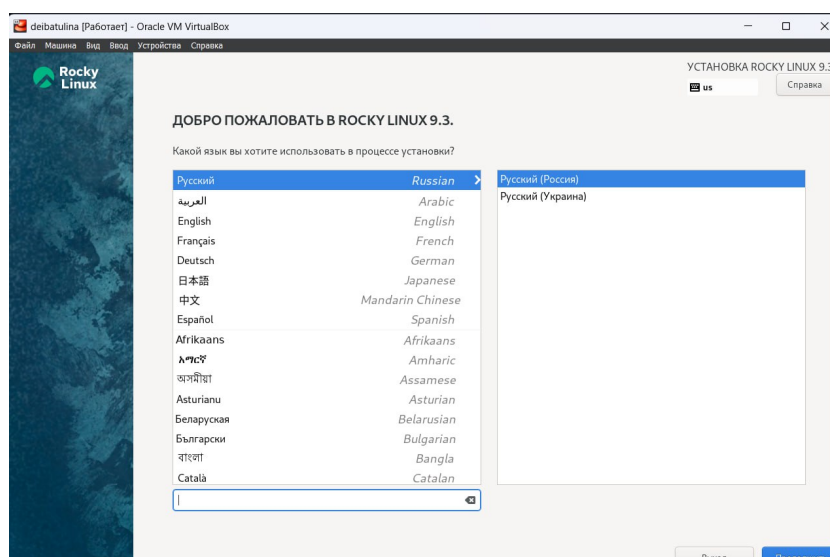


Рис. 4.9: Выбор языка установки

Отключаем kdump (рис. [4.10]).

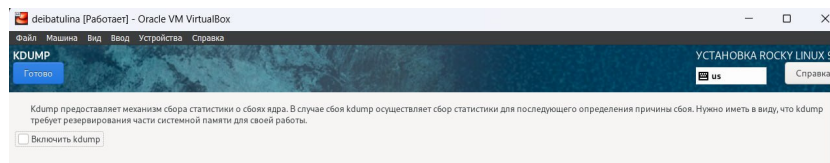


Рис. 4.10: Отключение kdump

Далее выбираем программы (Development Tools/Средства разработки) и в графе “Базовое окружение” выбираем “Сервер с GUI” (рис. [4.11]).

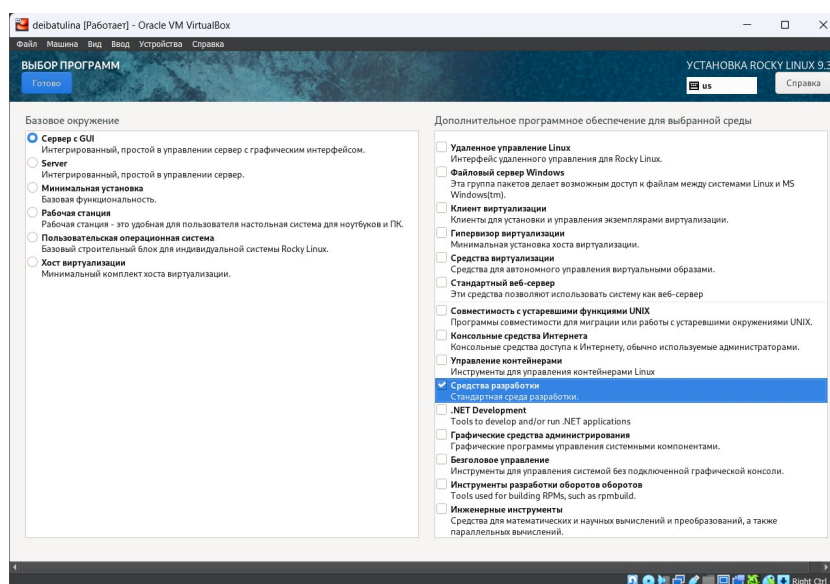


Рис. 4.11: Выбор программ

Задаём пароль root-администратора (рис. [4.12]).

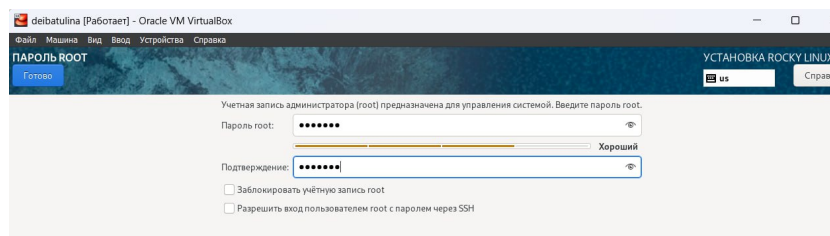


Рис. 4.12: Установка пароля root

Выберем устройства для установки операционной системы (рис. [4.13]).

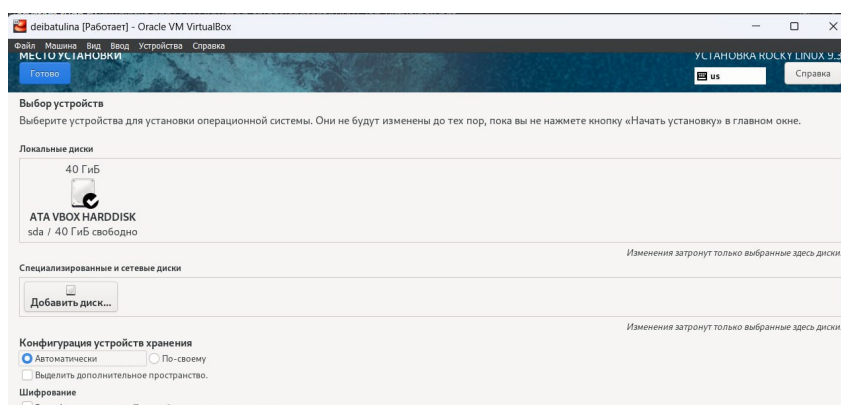


Рис. 4.13: Выбор устройств установки ОС

Настраиваем сеть и даём имя узлу в формате user.localdomain (где user - это логин в дисплейном классе) (рис. [4.14]).

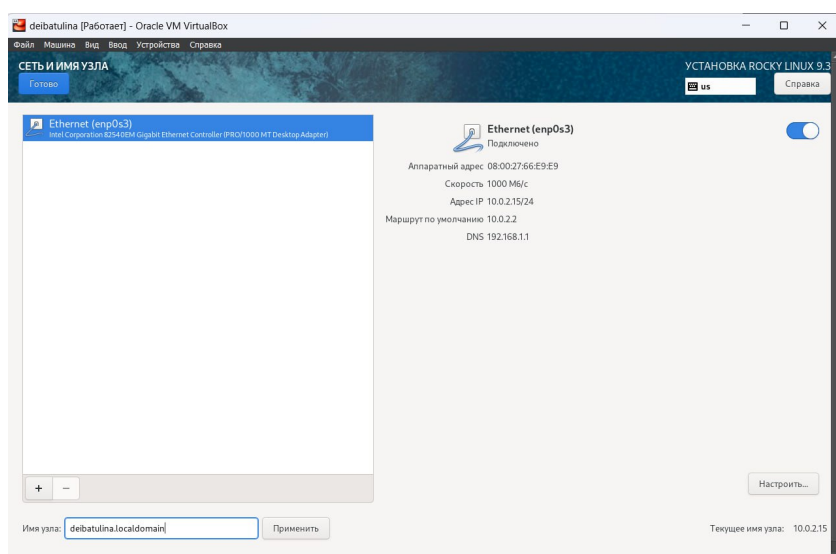


Рис. 4.14: Настройка сети и имени узла

Далее проверяем, все ли параметры установлены так, как прописано в руководстве к выполнению лабораторной работы (рис. [4.15]).

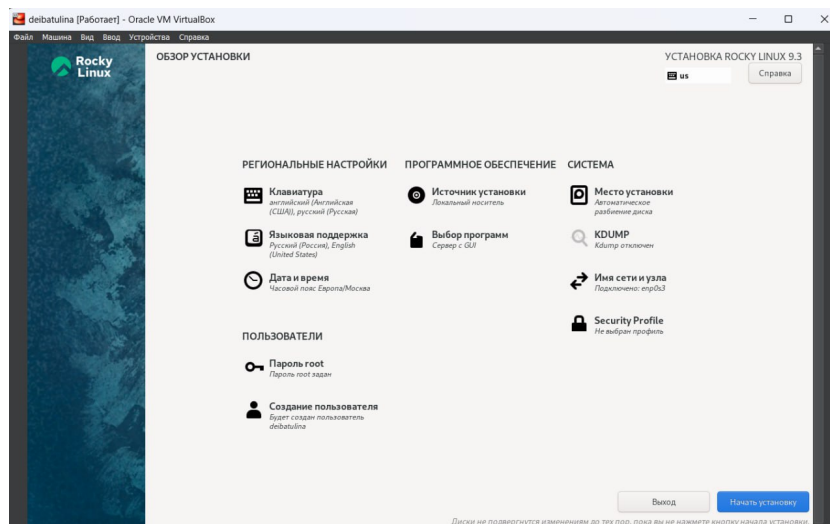


Рис. 4.15: Обзор параметров

Затем появляется окно с установкой. И видим, что система установлена (рис. [4.16]).

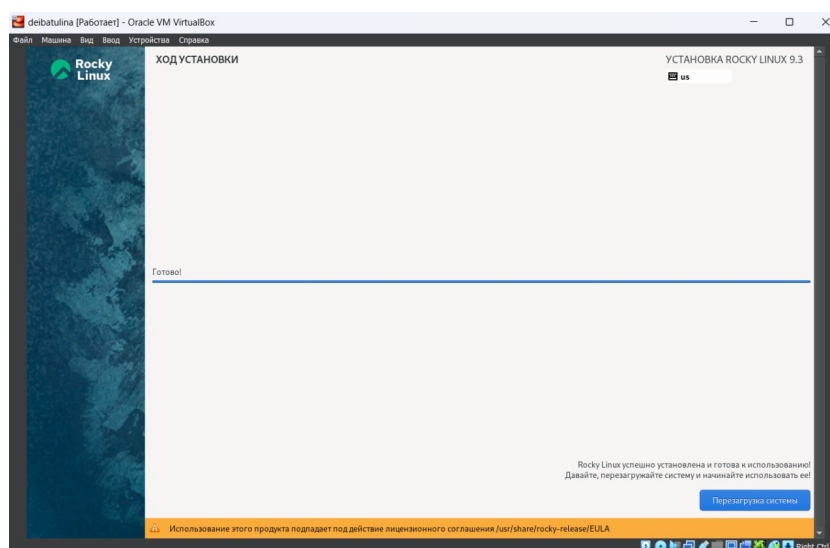


Рис. 4.16: Завершение установки

Перезагружаем систему и видим интерфейс Linux Rocky. Очень похож на Fedora, в которой мы работали в прошлом году (рис. [4.17]).

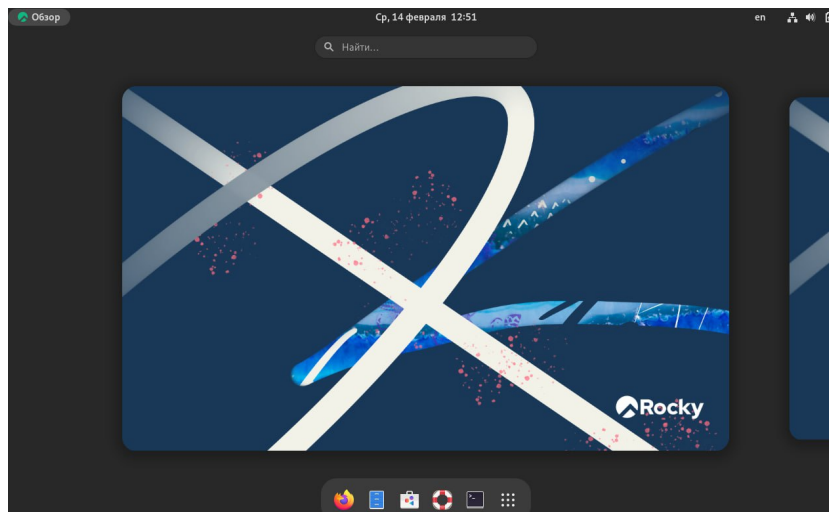


Рис. 4.17: Интерфейс Linux Rocky

Подключаем образ диска дополнений гостевой ОС (рис. [4.18]).

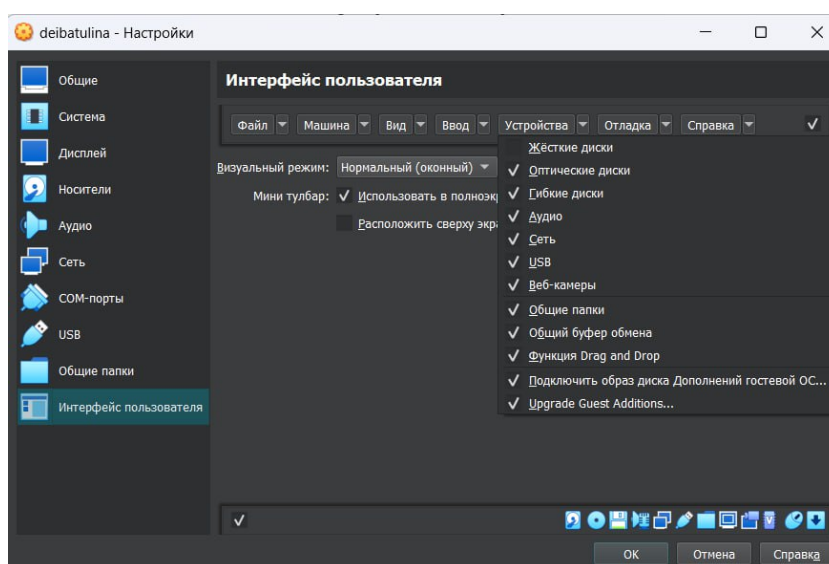


Рис. 4.18: Подключение образа диска дополнений гостевой ОС

Приступаем к выполнению заданий лабораторной работы:

Дождитесь загрузки графического окружения и откройте терминал. В окне терминала проанализируйте последовательность загрузки системы, выполнив команду `dmesg`. Можно просто посмотреть вывод этой команды (рис. [4.19]).


```
deibatulina@deibatulina:~$ dmesg
[ 0.000000] Linux version 5.14.0-362.18.1.el9_3.x86_64 (mockbuild@iad1-prod-b
uild001.bld.equ.rockylinux.org) (gcc (GCC) 11.4.1 20230605 (Red Hat 11.4.1-2), G
NU ld version 2.35.2-42.el9) #1 SMP PREEMPT_DYNAMIC Wed Jan 24 23:11:18 UTC 2024
[ 0.000000] The list of certified hardware and cloud instances for Enterprise
Linux 9 can be viewed at the Red Hat Ecosystem Catalog, https://catalog.redhat.
com.
[ 0.000000] Command line: BOOT_IMAGE=(hd0,msdos1)/vmlinuz-5.14.0-362.18.1.el9
_3.x86_64 root=/dev/mapper/rl_deibatulina-root ro resume=/dev/mapper/rl_deibatul
ina-swap rd.lvm.lv=rl_deibatulina/root rd.lvm.lv=rl_deibatulina/swap rhgb quiet
crashkernel=1G-4G:192M,4G-64G:256M,64G-:512M
[ 0.000000] x86/fpu: x87 FPU will use FXSAVE
[ 0.000000] signal: max sigframe size: 1440
[ 0.000000] BIOS-provided physical RAM map:
[ 0.000000] BIOS-e820: [mem 0x0000000000000000-0x00000000000009fbff] usable
[ 0.000000] BIOS-e820: [mem 0x00000000000009fc00-0x00000000000009ffff] reserved
[ 0.000000] BIOS-e820: [mem 0x0000000000000f0000-0x0000000000000fffff] reserved
[ 0.000000] BIOS-e820: [mem 0x000000000001000000-0x000000000000dfffff] usable
[ 0.000000] BIOS-e820: [mem 0x00000000dffff0000-0x00000000dffffffffff] ACPI data
[ 0.000000] BIOS-e820: [mem 0x00000000fec00000-0x00000000fec00fff] reserved
[ 0.000000] BIOS-e820: [mem 0x00000000fee00000-0x00000000fee00fff] reserved
[ 0.000000] BIOS-e820: [mem 0x00000000fffc0000-0x00000000ffffffff] reserved
[ 0.000000] BIOS-e820: [mem 0x0000000100000000-0x0000000025ffffffffff] usable
[ 0.000000] NX (Execute Disable) protection: active
[ 0.000000] SMBIOS 2.5 present.
[ 0.000000] DMI: innotek GmbH VirtualBox/VirtualBox, BIOS VirtualBox 12/01/20
06
[ 0.000000] Hypervisor detected: KVM
[ 0.000000] kvm-clock: Using msrs 4b564d01 and 4b564d00
[ 0.000003] kvm-clock: using sched offset of 4119805536497 cycles
[ 0.000007] clocksource: kvm-clock: mask: 0xffffffffffffffff max_cycles: 0x1c
d42e4dffb, max_idle_ns: 881590591483 ns
```

Рис. 4.19: Вывод команды dmesg

Получим следующую информацию (используя grep для поиска определённой информации): 1. Версия ядра Linux (Linux version) (рис. [4.20]).

```
deibatulina@deibatulina ~]$ dmesg | grep -i "Linux version"
[ 0.000000] Linux version 5.14.0-362.18.1.el9_3.x86_64 (mockbuild@iad1-prod-b
uild001.bld.equ.rockylinux.org) (gcc (GCC) 11.4.1 20230605 (Red Hat 11.4.1-2), G
NU ld version 2.35.2-42.el9) #1 SMP PREEMPT_DYNAMIC Wed Jan 24 23:11:18 UTC 2024
deibatulina@deibatulina ~]$
```

Рис. 4.20: Версия ядра Linux

2. Частота процессора (Detected Mhz processor) (рис. [4.21]).

```
deibatulina@deibatulina ~]$ dmesg | grep -i "processor"
[ 0.000010] tsc: Detected 3110.398 MHz processor
[ 1.399782] smpboot: Total of 11 processors activated (68428.75 BogoMIPS)
[ 3.308941] ACPI: Added _OSI(Processor Device)
[ 3.308942] ACPI: Added _OSI(Processor Aggregator Device)
deibatulina@deibatulina ~]$
```

Рис. 4.21: Частота процессора

3. Модель процессора (CPU0) (рис. [4.22]).

```
[deibatulina@deibatulina ~]$ dmesg | grep -i "CPU0"
[ 0.610910] smpboot: CPU0: 12th Gen Intel(R) Core(TM) i5-12500H (family: 0x6,
model: 0x9a, stepping: 0x3)
[deibatulina@deibatulina ~]$
```

Рис. 4.22: Модуль процессора

4. Объем доступной оперативной памяти (Memory available) (рис. [4.23]).

```
[ 0.408276] Memory: 3355304K/9436728K available (16384K kernel code, 5596K rw
data, 11460K rodata, 3824K init, 18412K bss, 622992K reserved, 0K cma-reserved)
[ 0.501210] Freeing SMP alternatives memory: 36K
[ 3.297662] x86/mm: Memory block size: 128MB
[ 8.107385] Non-volatile memory driver v1.3
[ 8.998580] Freeing initrd memory: 55248K
[ 10.320761] Freeing unused decrypted memory: 2036K
[ 10.322323] Freeing unused kernel image (initmem) memory: 3824K
[ 10.335831] Freeing unused kernel image (rodata/data gap) memory: 828K
[ 50.384924] vmwgfx 0000:00:02.0: [drm] Legacy memory limits: VRAM = 16384 kB,
FIFO = 2048 kB, surface = 507904 kB
[ 50.384929] vmwgfx 0000:00:02.0: [drm] Maximum display memory size is 16384 k
iB
[deibatulina@deibatulina ~]$
```

Рис. 4.23: Объём доступной оперативной памяти

5. Тип обнаруженного гипервизора (Hypervisor detected) (рис. [4.24]).

```
[deibatulina@deibatulina ~]$ dmesg | grep -i "Hypervisor detected"
[ 0.000000] Hypervisor detected: KVM
[deibatulina@deibatulina ~]$
```

Рис. 4.24: Тип обнаруженного гипервизора

6. Тип файловой системы корневого раздела (рис. [4.25]).

```
[deibatulina@deibatulina ~]$ df -hT
```

Файловая система	Тип	Размер	Использовано	Дост	Использовано%
Смонтировано в					
devtmpfs	devtmpfs	4,0M	0	4,0M	0%
/dev					
tmpfs	tmpfs	4,3G	0	4,3G	0%
/dev/shm					
tmpfs	tmpfs	1,7G	9,3M	1,7G	1%
/run					
/dev/mapper/rl_deibatulina-root	xfs	35G	5,2G	30G	15%
/					
/dev/sda1	xfs	960M	374M	587M	39%
/boot					
tmpfs	tmpfs	867M	120K	867M	1%
/run/user/1000					
/dev/sr0	iso9660	51M	51M	0	100%
/run/media/deibatulina/VBox_GAs_7.0.6					

```
[deibatulina@deibatulina ~]$
```

Рис. 4.25: Тип файловой системы корневого раздела

7. Последовательность монтирования файловых систем (рис. [4.26]).

```
[deibatulina@deibatulina ~]$ mount
proc on /proc type proc (rw,nosuid,nodev,noexec,relatime)
sysfs on /sys type sysfs (rw,nosuid,nodev,noexec,relatime,seclabel)
devtmpfs on /dev type devtmpfs (rw,nosuid,seclabel,size=4096k,nr_inodes=1101721,mode=755,inode64)
securityfs on /sys/kernel/security type securityfs (rw,nosuid,nodev,noexec,relatime)
tmpfs on /dev/shm type tmpfs (rw,nosuid,nodev,seclabel,inode64)
devpts on /dev/pts type devpts (rw,nosuid,noexec,relatime,seclabel,gid=5,mode=620,ptmxmode=000)
tmpfs on /run type tmpfs (rw,nosuid,nodev,seclabel,size=1775144k,nr_inodes=819200,mode=755,inode64)
cgroup2 on /sys/fs/cgroup type cgroup2 (rw,nosuid,nodev,noexec,relatime,seclabel,nsdelegate,memory_recursiveprot)
pstore on /sys/fs/pstore type pstore (rw,nosuid,nodev,noexec,relatime,seclabel)
bpf on /sys/fs/bpf type bpf (rw,nosuid,nodev,noexec,relatime,mode=700)
/dev/mapper/rl_deibatulina-root on / type xfs (rw,relatime,seclabel,attr2,inode64,logbufs=8,logbsize=32k,noquota)
selinuxfs on /sys/fs/selinux type selinuxfs (rw,nosuid,noexec,relatime)
systemd-1 on /proc/sys/fs/binfmt_misc type autofs (rw,relatime,fd=29,pgrp=1,timeout=0,minproto=5,maxproto=5,direct,pipe_ino=20645)
mqueue on /dev/mqueue type mqueue (rw,nosuid,nodev,noexec,relatime,seclabel)
tracefs on /sys/kernel/tracing type tracefs (rw,nosuid,nodev,noexec,relatime,seclabel)
debugfs on /sys/kernel/debug type debugfs (rw,nosuid,nodev,noexec,relatime,seclabel)
hugetlbfs on /dev/hugepages type hugetlbfs (rw,relatime,seclabel,pagesize=2M)
fusectl on /sys/fs/fuse/connections type fusectl (rw,nosuid,nodev,noexec,relatime)
```

Рис. 4.26: Последовательность монтирования файловых систем

5 Выводы

В данной лабораторной работе я попрактиковалась в установке операционной системы Linux Rocky на виртуальную машину, настроила необходимые для работы сервисы и выполнила задания (познакомилась с командой `dmesg` и той информацией об операционной системе, которую можно получить с помощью данной команды).

6 Ответы на контрольные вопросы

1. Какую информацию содержит учётная запись пользователя?

Учетная запись пользователя содержит информацию по авторизации - учётные данные. Это идентификатор для подключения к системе. То есть, это:

Системное имя - должно быть уникальным, содержит только латинские знаки. Уникальный идентификатор пользователя в системе, содержит число. Полное имя - ФИО пользователя.

2. Укажите команды терминала и приведите примеры: – для получения справки по команде (help); – для перемещения по файловой системе (cd); – для просмотра содержимого каталога (ls); – для определения объёма каталога (du + имя каталога); – для создания / удаления каталогов / файлов (rm - для удаления файлов, rmdir - для удаления каталогов, mkdir - для создания каталогов, touch - для создания файлов); – для задания определённых прав на файл / каталог (chmod); – для просмотра истории команд (history).
3. Что такое файловая система? Приведите примеры с краткой характеристикой.

Это способ хранения, организации и именования данных на различных носителях.

Примеры:

FAT32 - файловая система, в которой пространство разделено на три части: область служебных структур, указатели в виде таблиц и область хранения файлов;

ext4 - система, которая используется в основном в ОС на Linux. Журналируемая файловая система, в последней версии максимальный размер файла - 16Гб.

4. Как посмотреть, какие файловые системы подмонтированы в ОС?

Для этого нужно ввести в терминал команду `df` - это покажет список всех файловых систем по номерам устройств, размеры и данные о памяти. Но при этом можно посмотреть в свойствах папок все эти данные вручную.

5. Как удалить зависший процесс?

- Команда `killall` - остановит все процессы, которые есть в данный момент.
- Команда `kill + id-процесса` поможет удалить один конкретный процесс. (Чтобы узнать `id` нужно написать в терминале команду `ps`).

Список литературы

1. Руководство к лабораторной работе №1 “Установка и конфигурация операционной системы на виртуальную машину”, Д.С. Кулябов, А.В. Королькова, М.Н. Геворкян.