

Отчёт по лабораторной работе

Дисциплина: Основы информационной безопасности

Набережных Дарина Денисовна

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
2.1	Создание виртуальной машины и выбор ОС	6
2.2	Домашнее задание	14
3	Выводы	16
4	Контрольные вопросы	17

Список иллюстраций

2.1	Выбор расположения виртуальной машины	6
2.2	Выбор количества оперативной памяти	7
2.3	Создание виртуального жесткого диска	7
2.4	Формат жесткого диска	8
2.5	Тип хранения	9
2.6	Объем жесткого диска	9
2.7	Выбор оптического диска	10
2.8	Запуск виртуальной машины	10
2.9	Выбор языка системы	11
2.10	Отключение KDUMP	11
2.11	Подключение сетевого соединения	12
2.12	Установка пароля	13
2.13	Создание пользователя	13
2.14	Принятие условий лицензии	14
2.15	Установка имени хоста	14
2.16	Применение команды dsmeg	15

Список таблиц

1 Цель работы

Научиться создавать виртуальную машину и устанавливать на нее ОС Linux, а так же настроить необходимые для дальнейшей работы сервисы.

2 Выполнение лабораторной работы

2.1 Создание виртуальной машины и выбор ОС

Создадим виртуальную машину, используя программу Oracle VM VirtualBox. В начале устанавливаем расположение виртуальной машины и выбираем тип (рис. 2.1)

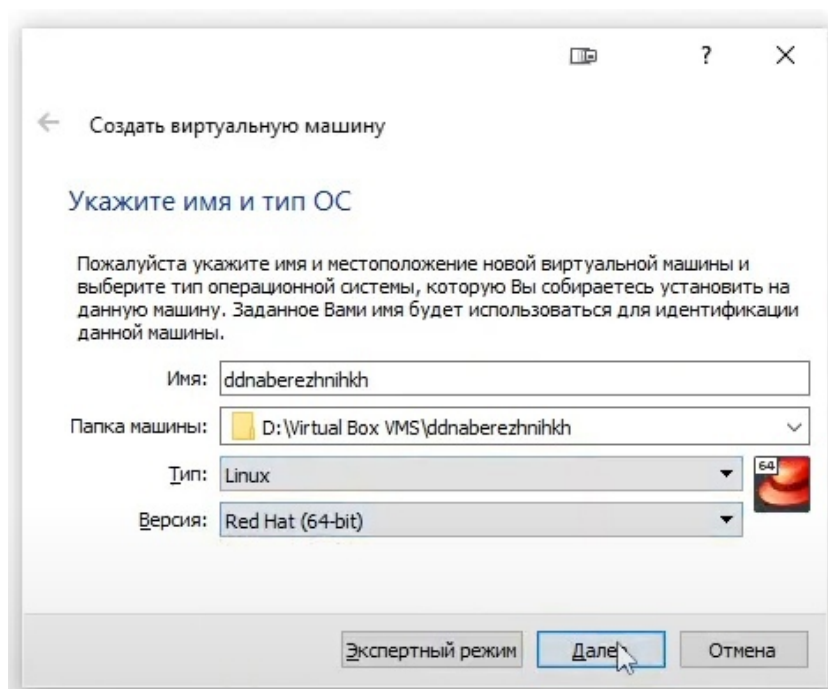


Рис. 2.1: Выбор расположения виртуальной машины

Зададим необходимое количество оперативной памяти - пусть это будет 12288 МБ (12 ГБ) (рис. 2.2)

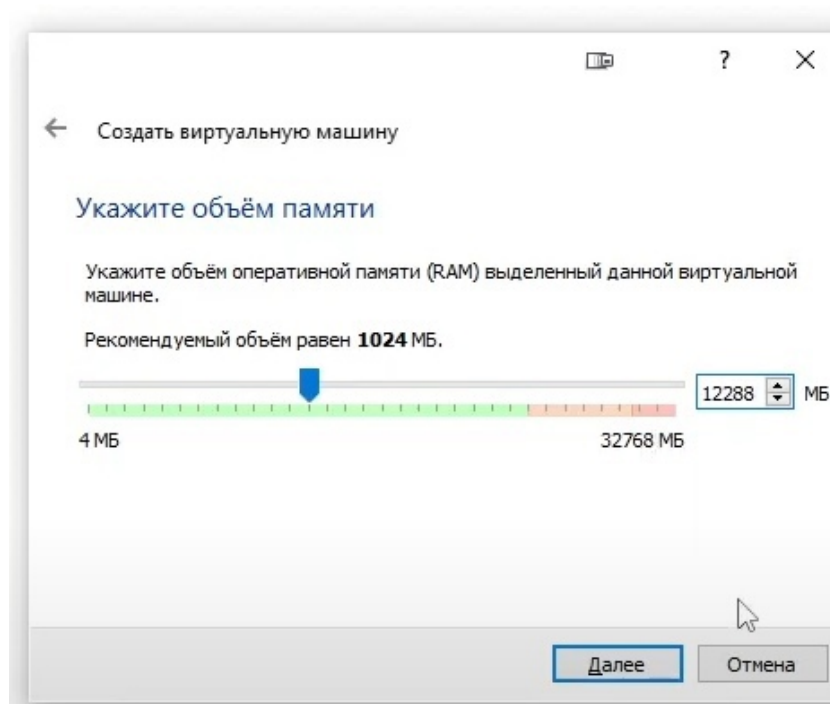


Рис. 2.2: Выбор количества оперативной памяти

Создадим новый виртуальный жесткий диск (рис. 2.3)

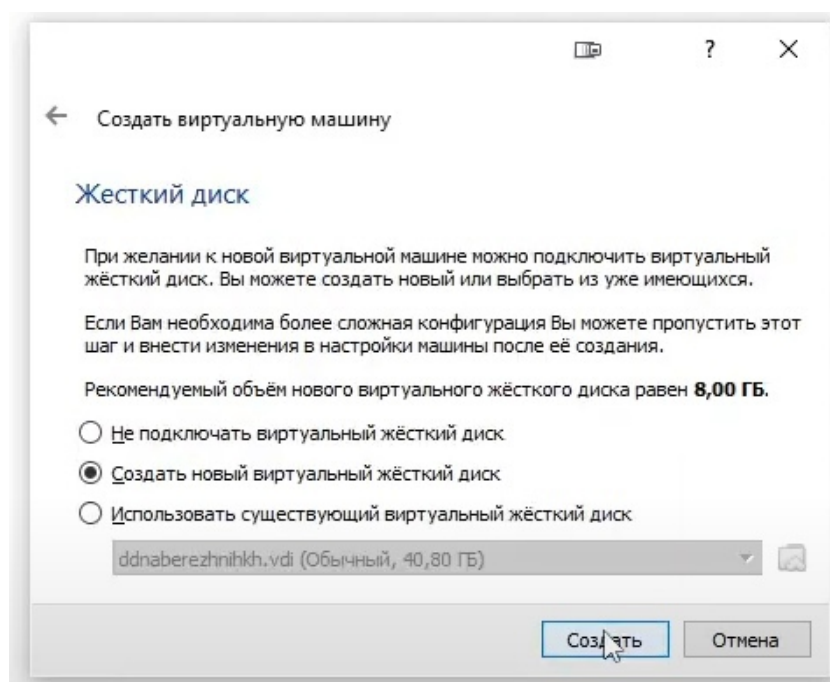


Рис. 2.3: Создание виртуального жесткого диска

Выбираем VDI как тип жесткого диска (рис. 2.4)

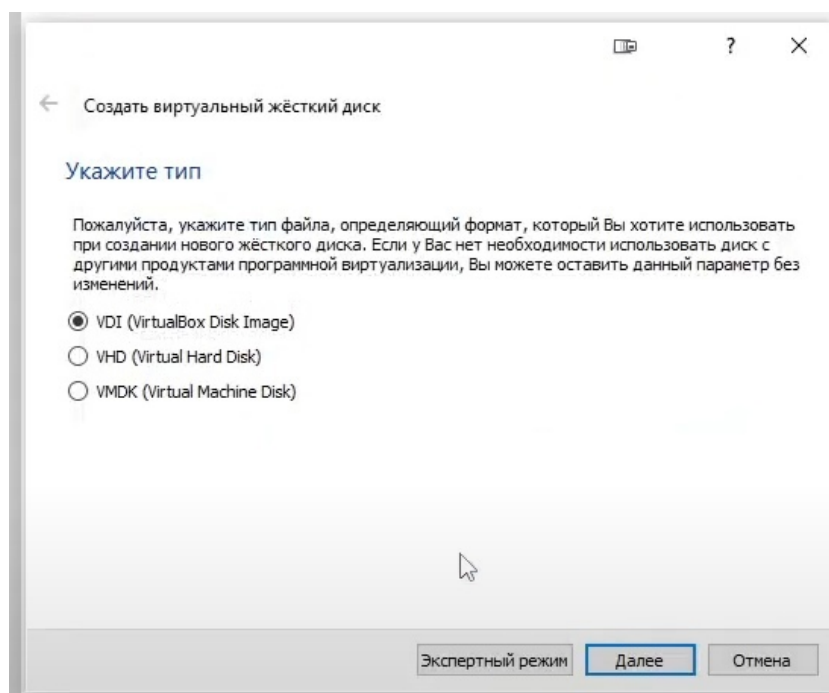


Рис. 2.4: Формат жесткого диска

Указываем формат хранения - динамический жесткий диск (рис. 2.5)

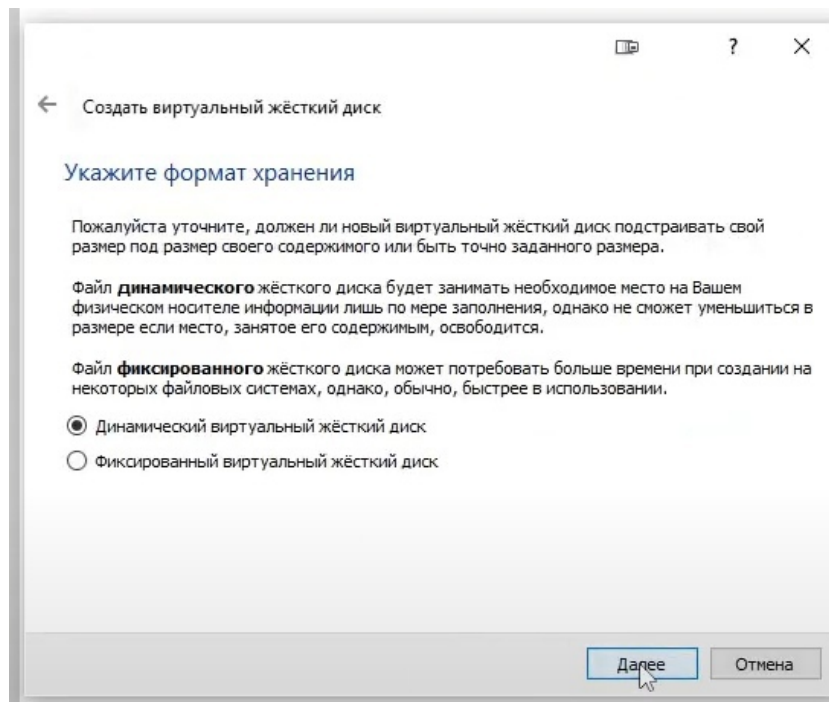


Рис. 2.5: Тип хранения

Выбираем объем жесткого диска - 40 ГБ (рис. 2.6)

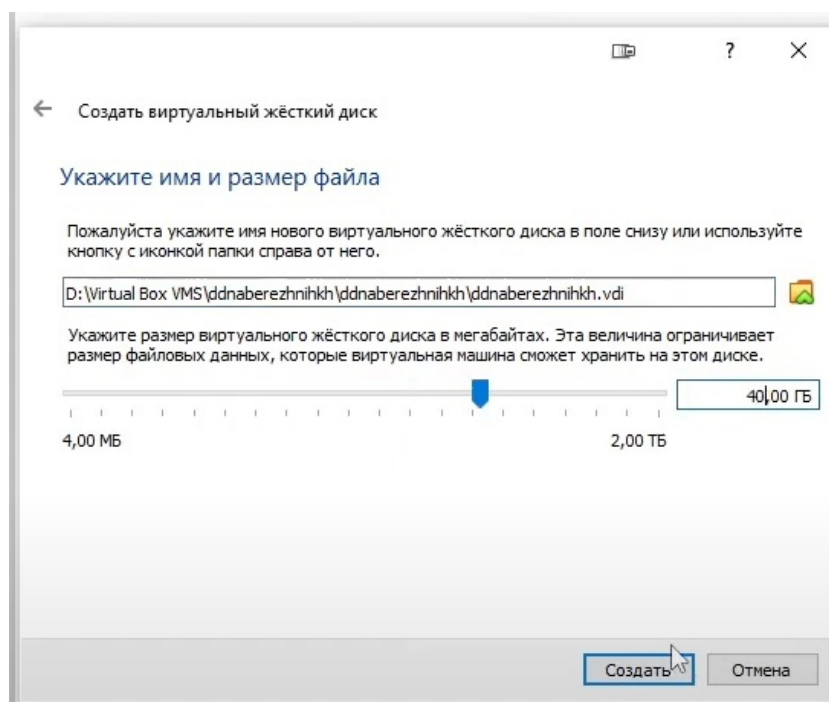


Рис. 2.6: Объем жесткого диска

Переходим в окно настроек виртуальной машины и в разделе “Носители” выбираем оптический диск (рис. 2.7)

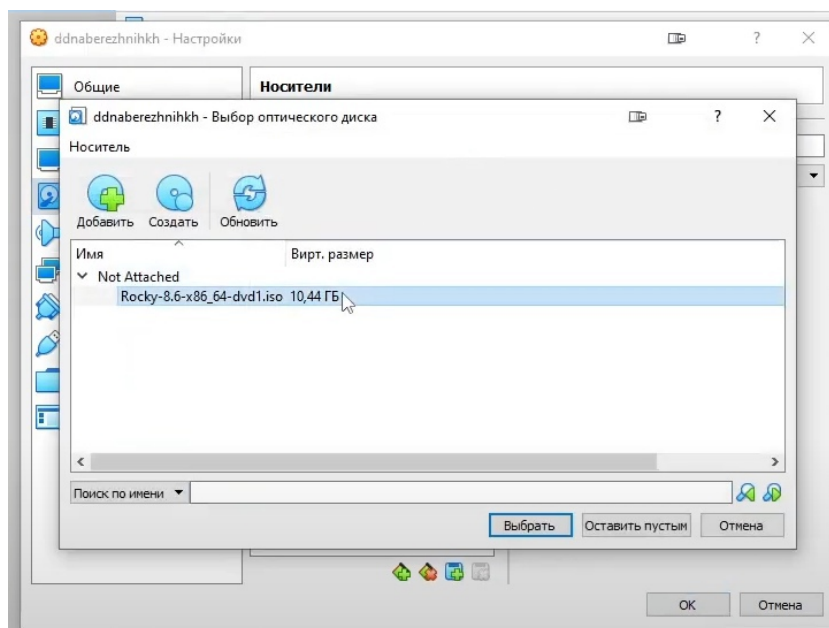


Рис. 2.7: Выбор оптического диска

Запускаем виртуальную машину (рис. 2.8)

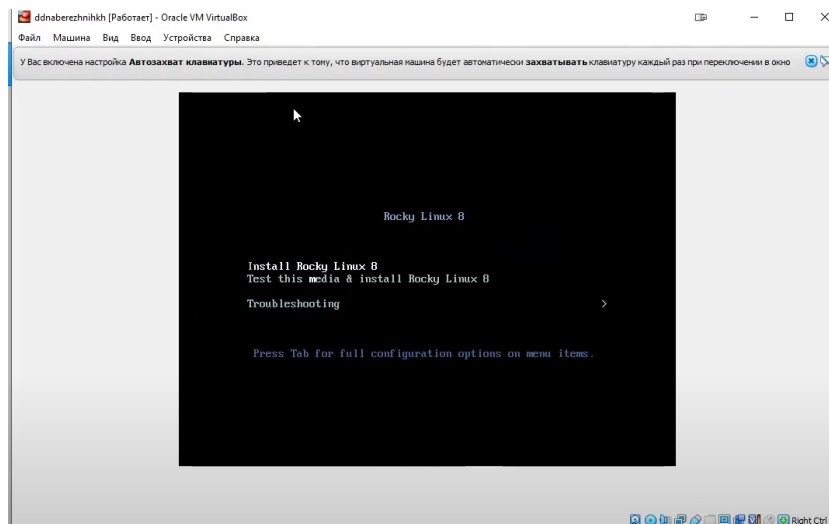


Рис. 2.8: Запуск виртуальной машины

Выбираем язык, я оставила английский (рис. 2.9)

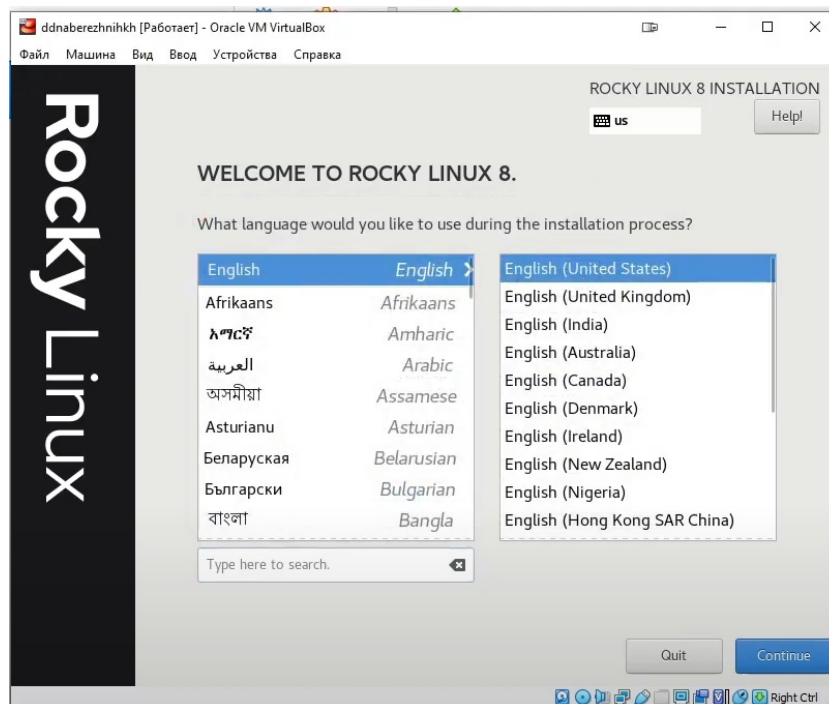


Рис. 2.9: Выбор языка системы

Отключаем KDUMP (рис. 2.10)

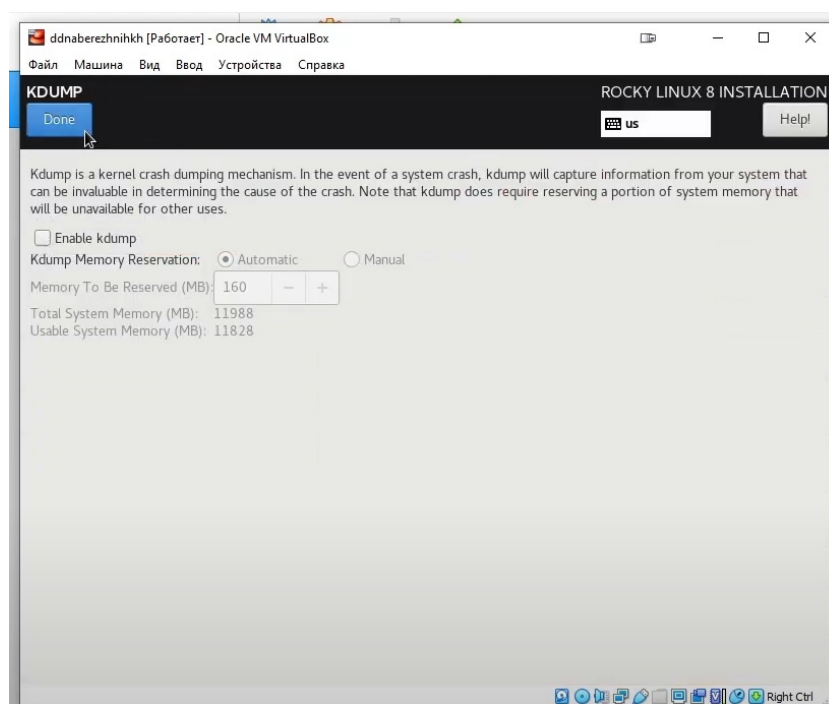


Рис. 2.10: Отключение KDUMP

Подключаем сетевое соединение (рис. 2.11)

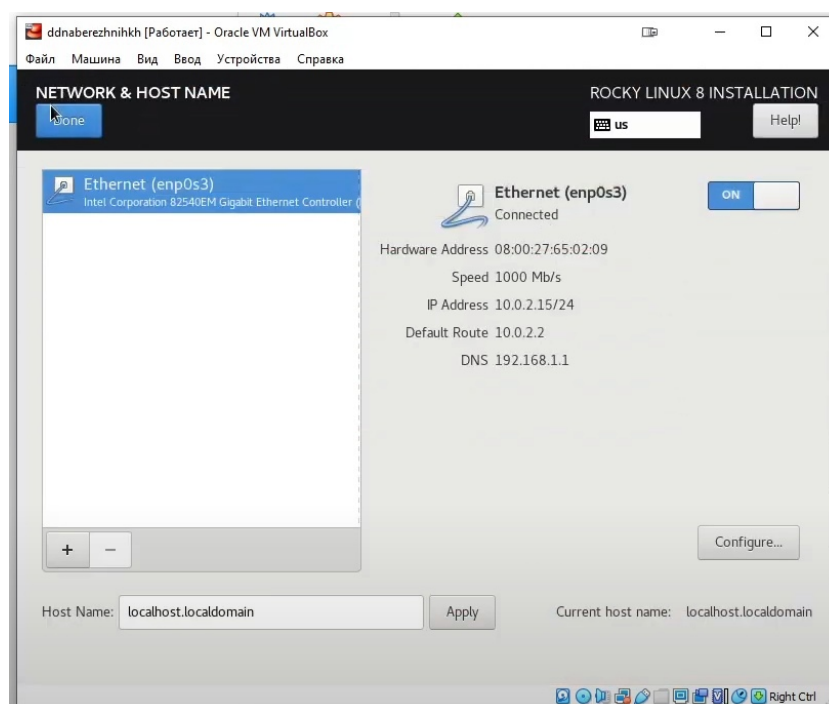


Рис. 2.11: Подключение сетевого соединения

Устанавливаем пароль для root и администратора (рис. 2.12)

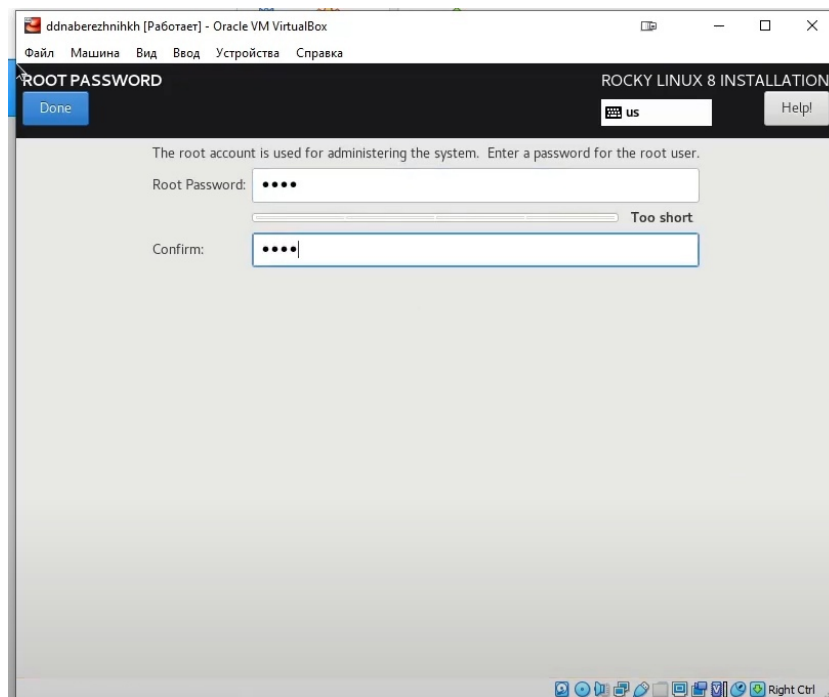


Рис. 2.12: Установка пароля

Создаем пользователя с правами администратора (рис. 2.13)

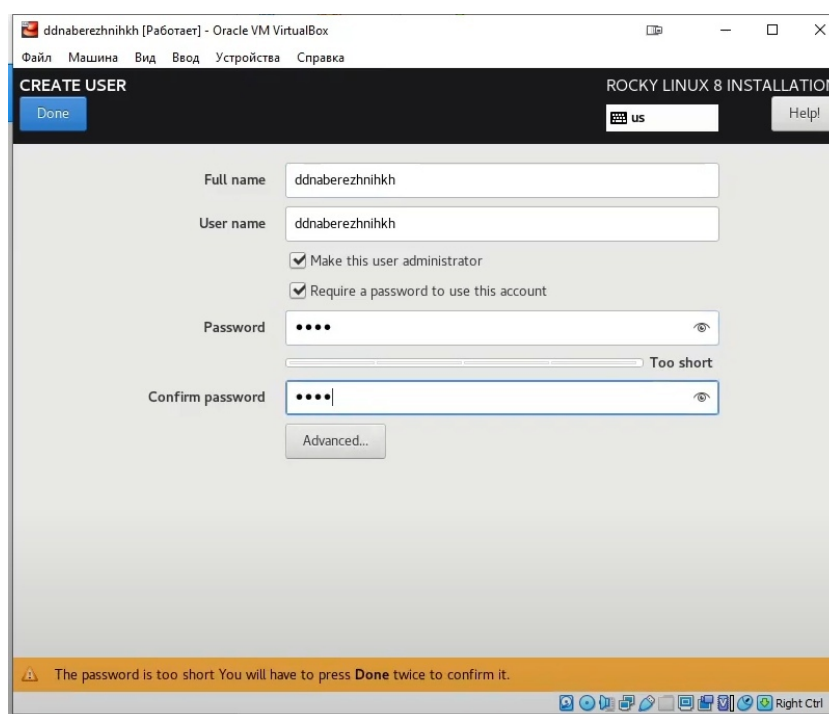


Рис. 2.13: Создание пользователя

Запускаем установку ОС, после завершения установки перезапускаем систему и принимаем условия лицензии (рис. 2.14)

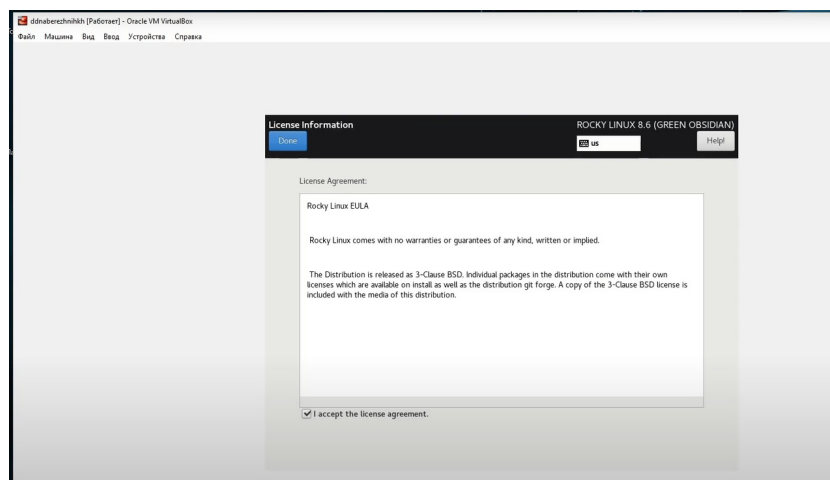


Рис. 2.14: Принятие условий лицензии

Открываем консоль и устанавливаем имя хоста (рис. 2.15)

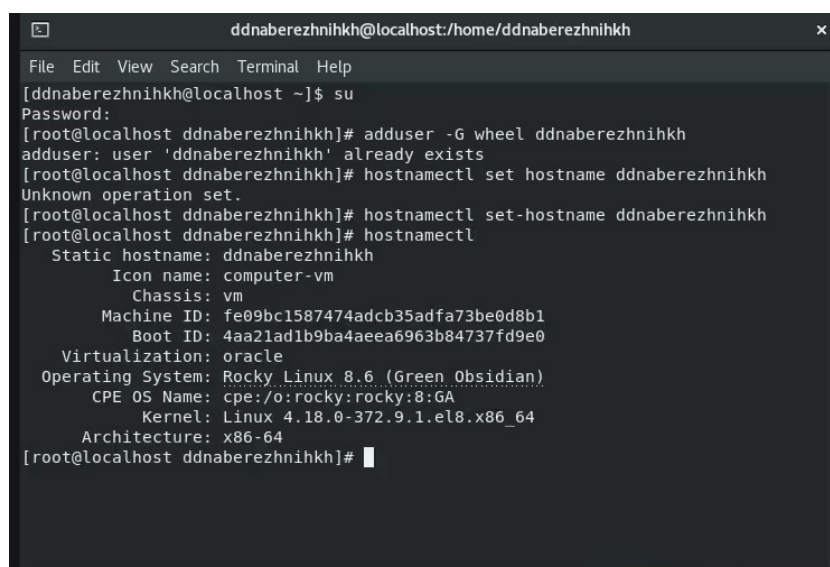


Рис. 2.15: Установка имени хоста

2.2 Домашнее задание

С помощью команды `dsmeq` узнала данные системы (рис. 2.16)

```
root@ddnaberezhnikh:~  
File Edit View Search Terminal Help  
[ddnaberezhnikh@ddnaberezhnikh ~]$ hostnamectl  
bash: hostnamectl: command not found...  
[ddnaberezhnikh@ddnaberezhnikh ~]$ clear  
[ddnaberezhnikh@ddnaberezhnikh ~]$ su -  
Password:  
[root@ddnaberezhnikh ~]# dmesg | grep -i 'Linux version'  
[ 0.000000] Linux version 4.18.0-372.9.1.el8.x86_64 (mockbuild@dal1-prod-buil  
der001.bld.equ.rockylinux.org) (gcc version 8.5.0 20210514 (Red Hat 8.5.0-10) (G  
CC)) #1 SMP Tue May 10 14:48:47 UTC 2022  
[root@ddnaberezhnikh ~]# dmesg | grep -i 'Mhz'  
[ 0.000000] tsc: Detected 2903.998 MHz processor  
[ 2.402792] e1000 0000:00:03:0 eth0: (PCI:33MHz:32-bit) 08:00:27:65:02:09  
[root@ddnaberezhnikh ~]# dmesg | grep -i 'CPU0'  
[ 0.110073] smpboot: CPU0: Intel(R) Core(TM) i5-10400F CPU @ 2.90GHz (family:  
0x6, model: 0xa5, stepping: 0x3)  
[root@ddnaberezhnikh ~]# dmesg | grep -i 'available'  
[ 0.000000] Zeroed struct page in unavailable ranges: 114 pages  
[ 0.000000] [mem 0xe0000000-0xfebfffff] available for PCI devices  
[ 0.000000] Memory: 3682312K/12582456K available (12293K kernel code, 5865K r  
wdata, 8292K rodata, 2520K init, 14348K bss, 367444K reserved, 0K cma-reserved)  
[ 1.915774] [TTM] Zone kernel: Available graphics memory: 6137858 KiB  
[ 1.915776] [TTM] Zone dma32: Available graphics memory: 2097152 KiB  
[ 9.193470] bridge: filtering via arp/ip/ip6tables is no longer available by  
default. Update your scripts to load br_netfilter if you need this.  
[root@ddnaberezhnikh ~]# dmesg | grep -i 'hypervisor'  
[ 0.000000] Hypervisor detected: KVM  
[ 1.915877] [drm] Max dedicated hypervisor surface memory is 507904 kiB  
[root@ddnaberezhnikh ~]# dmesg | grep -i 'filesystem'  
[ 2.664230] XFS (dm-0): Mounting V5 Filesystem  
[ 4.754553] XFS (sda1): Mounting V5 Filesystem  
[root@ddnaberezhnikh ~]# dmesg | grep -i 'mount'  
[ 0.003828] Mount-cache hash table entries: 32768 (order: 6, 262144 bytes, vm  
alloc)  
[ 0.003853] Mountpoint-cache hash table entries: 32768 (order: 6, 262144 byte  
s, vmalloc)  
[ 2.664230] XFS (dm-0): Mounting V5 Filesystem  
[ 2.678484] XFS (dm-0): Ending clean mount  
[ 4.754553] XFS (sda1): Mounting V5 Filesystem  
[ 4.942969] XFS (sda1): Ending clean mount  
[root@ddnaberezhnikh ~]#
```

Рис. 2.16: Применение команды dmesg

1. Версия ядра — 4.18.0-372.9.1.el8.x86_64
2. Частота процессора — 2903.998
3. Модель процессора — Intel(R) Core(TM) i5-10400F
4. Объем оперативной памяти — 3682312 Кбайт
5. Тип обнаруживаемого гипервизора — KVM
6. Тип файловой системы — XFS
7. Последовательность монтирования файловых систем можно найти через `dmesg | grep -i 'mount'`

3 Выводы

Познакомилась с процессом создания виртуальной машины, а так же установила на нее ОС Linux, после этого настроила для дальнейшей работы

4 Контрольные вопросы

1. Когда пользователь регистрируется в системе (проходит процедуру авторизации, например, вводя системное имя и пароль), он идентифицируется с учётной записью, в которой система хранит информацию о каждом пользователе: его системное имя и некоторые другие сведения, необходимые для работы с ним.
2. Команды терминала:
 1. Команда `man` для получения справки по команде
 2. Команда `cd` используется для перемещения по файловой системе
 3. Команда `ls` используется для просмотра содержимого каталога.
 4. Команда `du` отображает объем дисковой памяти (в килобайтах), занятой указанными файлами или каталогами.
 5. Команда `mkdir` используется для создания каталогов.
 6. Команда `rm` используется для удаления файлов и/или каталогов.
 7. Команда `chmod` (`change mode` – сменить режим) предназначена для изменения прав доступа к файлам и каталогам.
 8. Для вывода на экран списка ранее выполненных команд используется команда `history`.
3. Файловая система - часть операционной системы, которая обеспечивает чтение и запись файлов на дисковых носителях информации. Файловая система устанавливает физическую и логическую структуру файлов, правила их создания и управления ими, а также сопутствующие данные файла и идентификацию. Каждый дистрибутив Linux позволяет использовать одну

из этих файловых систем:

1. Ext2;
 2. Ext3;
 3. Ext4;
 4. JFS;
 5. ReiserFS;
 6. XFS;
 7. Btrfs;
 8. ZFS; Все они включены в ядро и могут использоваться в качестве корневой файловой системы.
4. С помощью `findmnt`. Можно переключиться из представления дерева по умолчанию в представление списка с помощью `-l`, определить выходные столбцы с помощью `-o`, отфильтровать результаты по типу файловой системы с помощью `-t`. Эта команда используется для поиска монтированных устройств, а также может монтировать или размонтировать их при необходимости. Для просмотра всех примонтированных файловых систем использовать команду: `findmnt --all`
5. Найдем процесс: `ps -A | grep "имя_процесса"` Для завершения процесса нужно вызвать утилиту `kill` с параметром `"-9"`. В этом случае она просто убьет процесс без процедуры "завершитесь, пожалуйста" (123 - ID процесса). `sudo kill -9 123` Также можно использовать утилиту `killall`, когда необходимо убить дерево процессов. `sudo killall имя_процесса`