

Цель работы

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

Задание

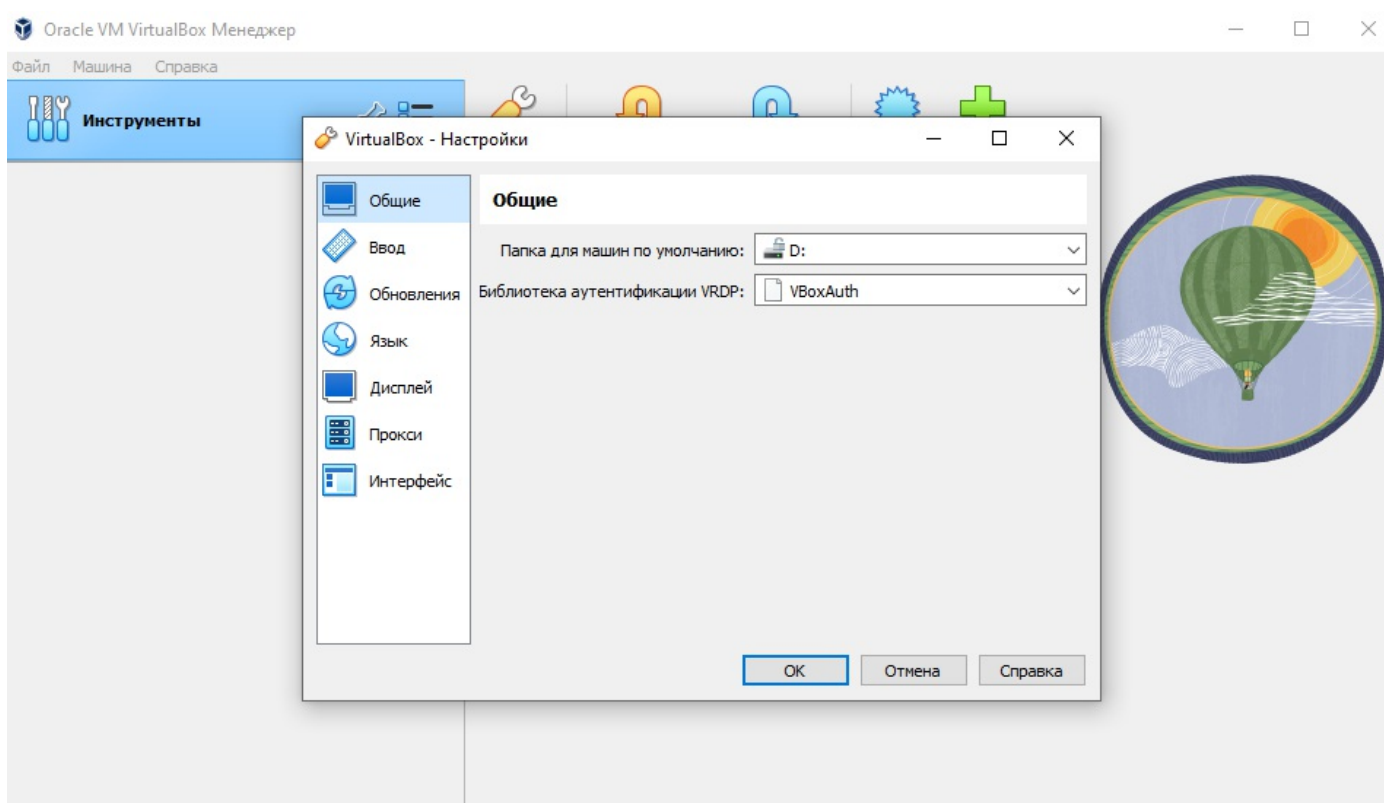
Лабораторная работа подразумевает установку на виртуальную машину VirtualBox (<https://www.virtualbox.org/>) операционной системы Linux (дистрибутив Rocky (<https://rockylinux.org/>) или CentOS (<https://www.centos.org/>)). Выполнение работы возможно как в дисплейном классе факультета физико-математических и естественных наук РУДН, так и дома. Описание выполнения работы приведено для дисплейного класса со следующими характеристиками: – Intel Core i3-550 3.2 GHz, 4 GB оперативной памяти, 20 GB свободного места на жёстком диске; – ОС Linux Gentoo (<http://www.gentoo.ru/>); – VirtualBox верс. 6.1 или старше; – каталог с образами ОС для работающих в дисплейном классе: [/afs/dk.sci.pfu.edu.ru/common/files/iso/](https://afs.dk.sci.pfu.edu.ru/common/files/iso/).

Выполнение лабораторной работы

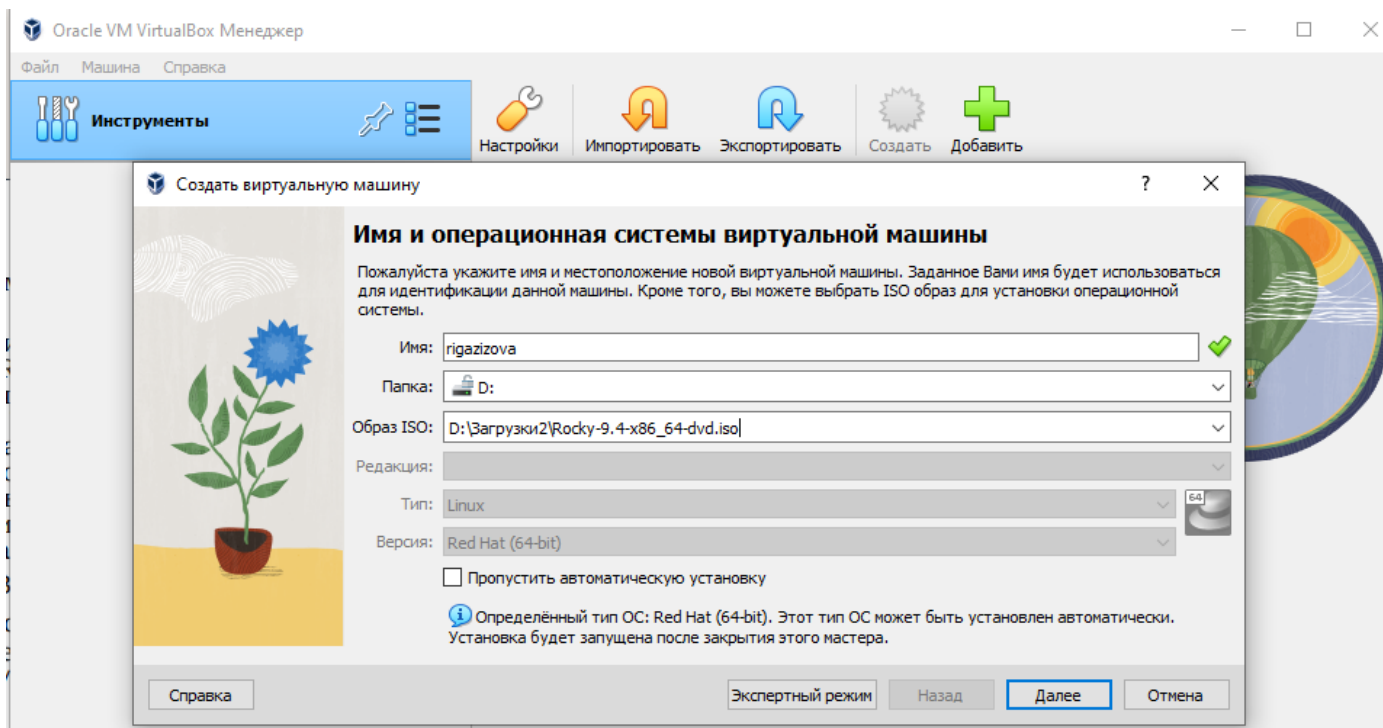
Для начала скачиваем VirtualBox



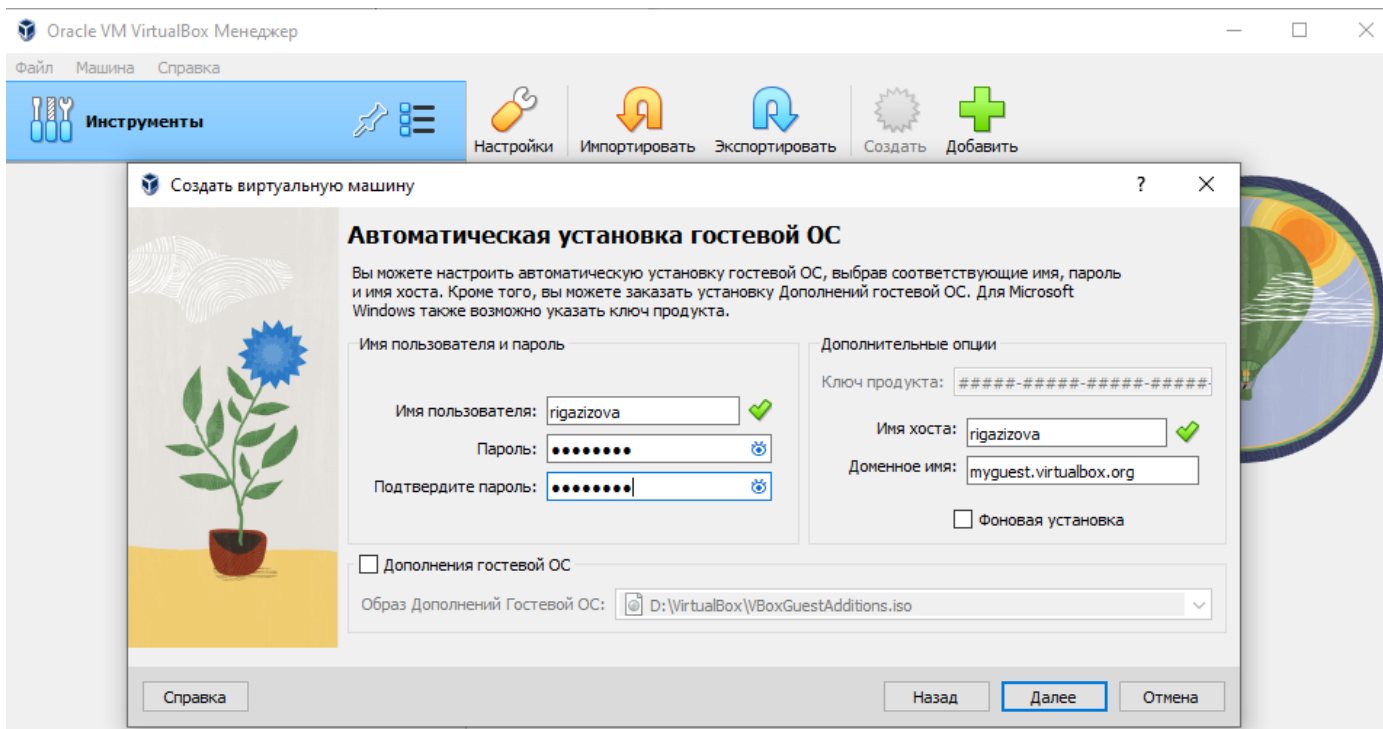
Создаем виртуальную машину



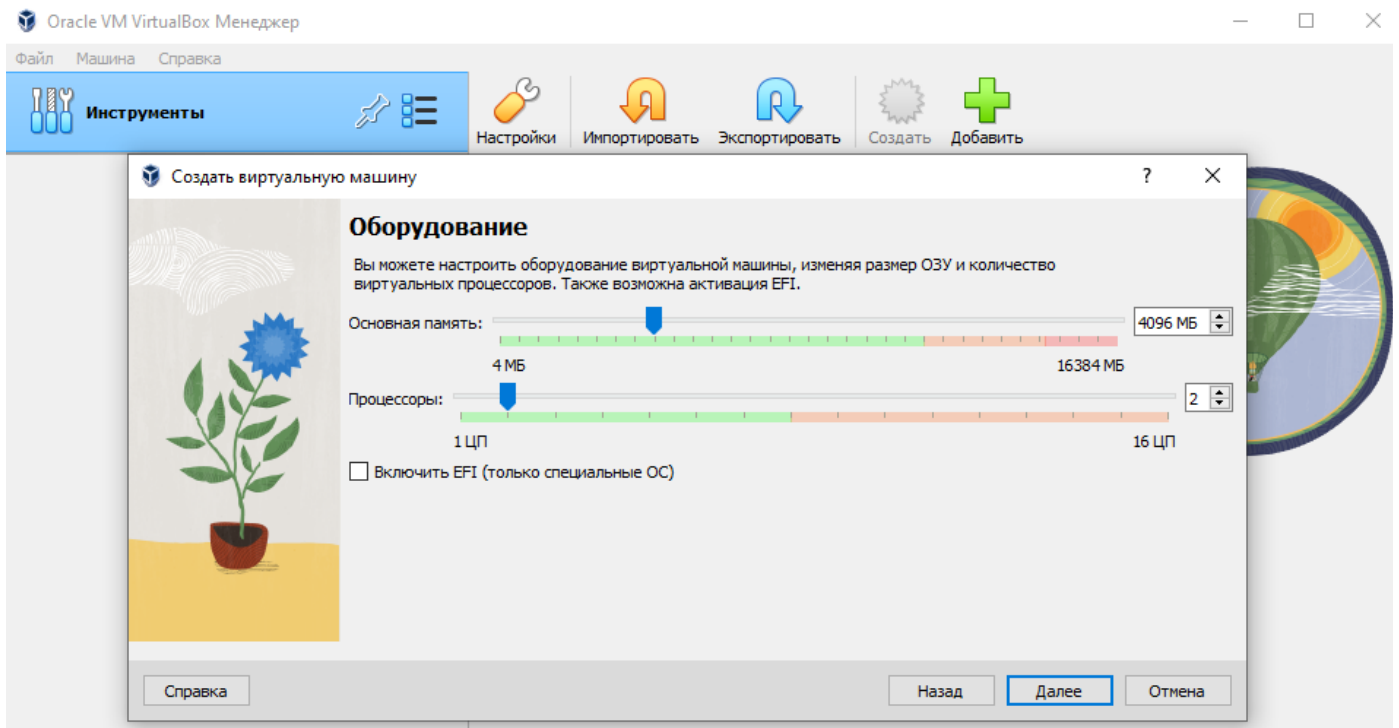
Указываем имя виртуальной машины (rigazizova), тип операционной системы — Linux, RedHat



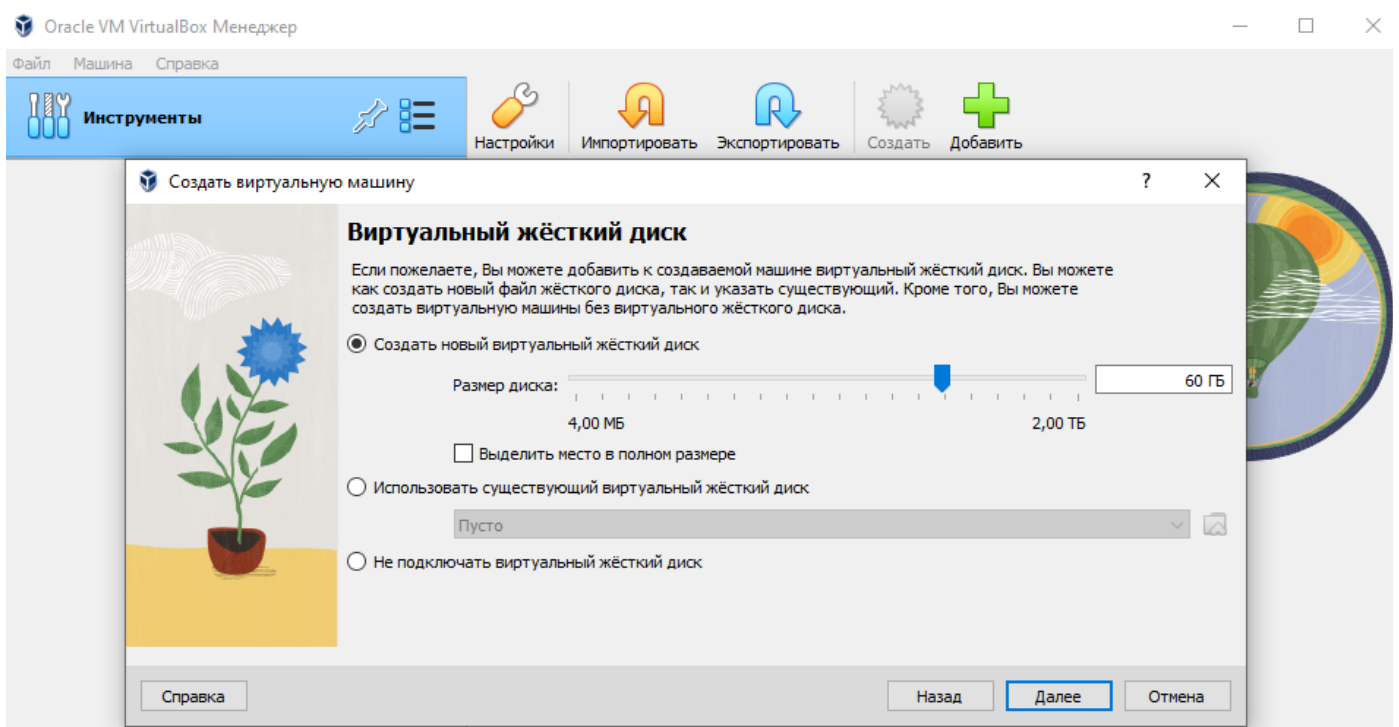
Так же для гостевой ОС вводим имя пользователя и хоста и их пароли



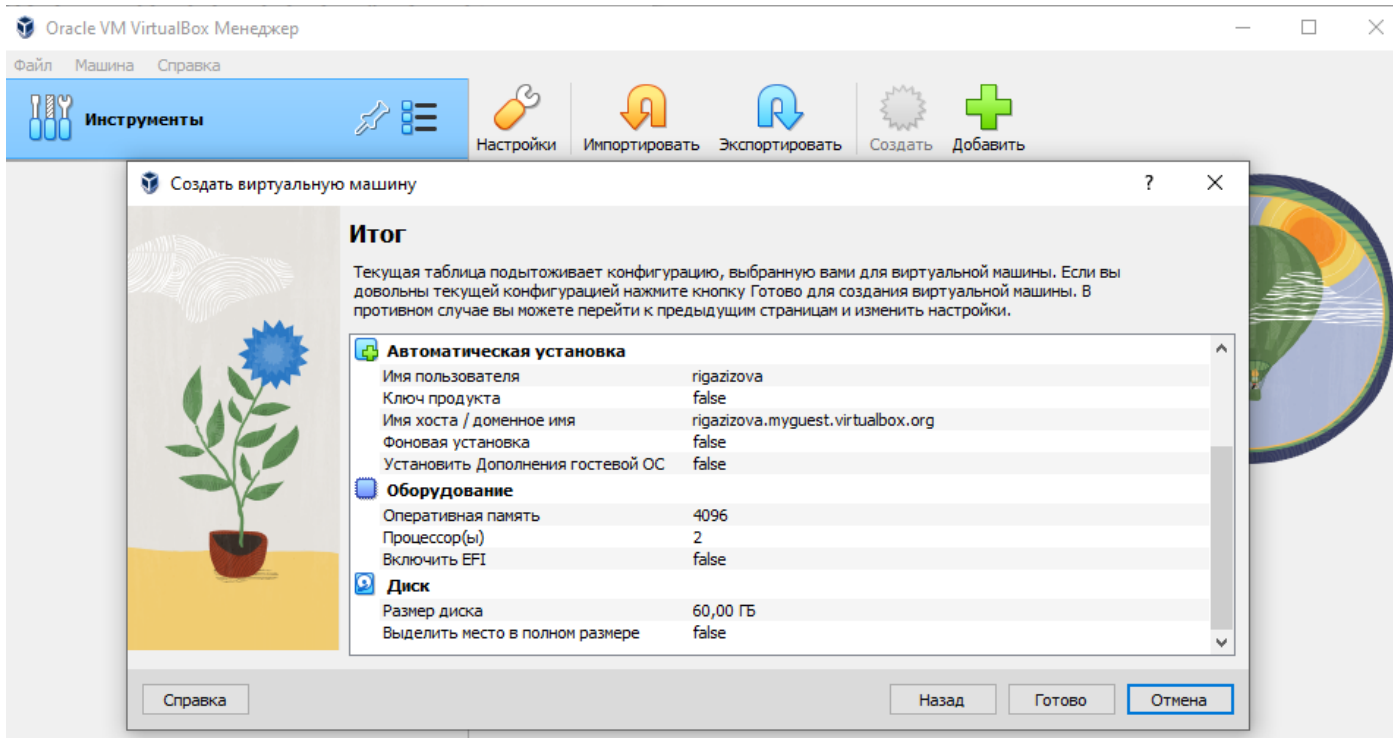
Указываем размер основной памяти виртуальной машины 4096 МБ и выделяем 2 процессора



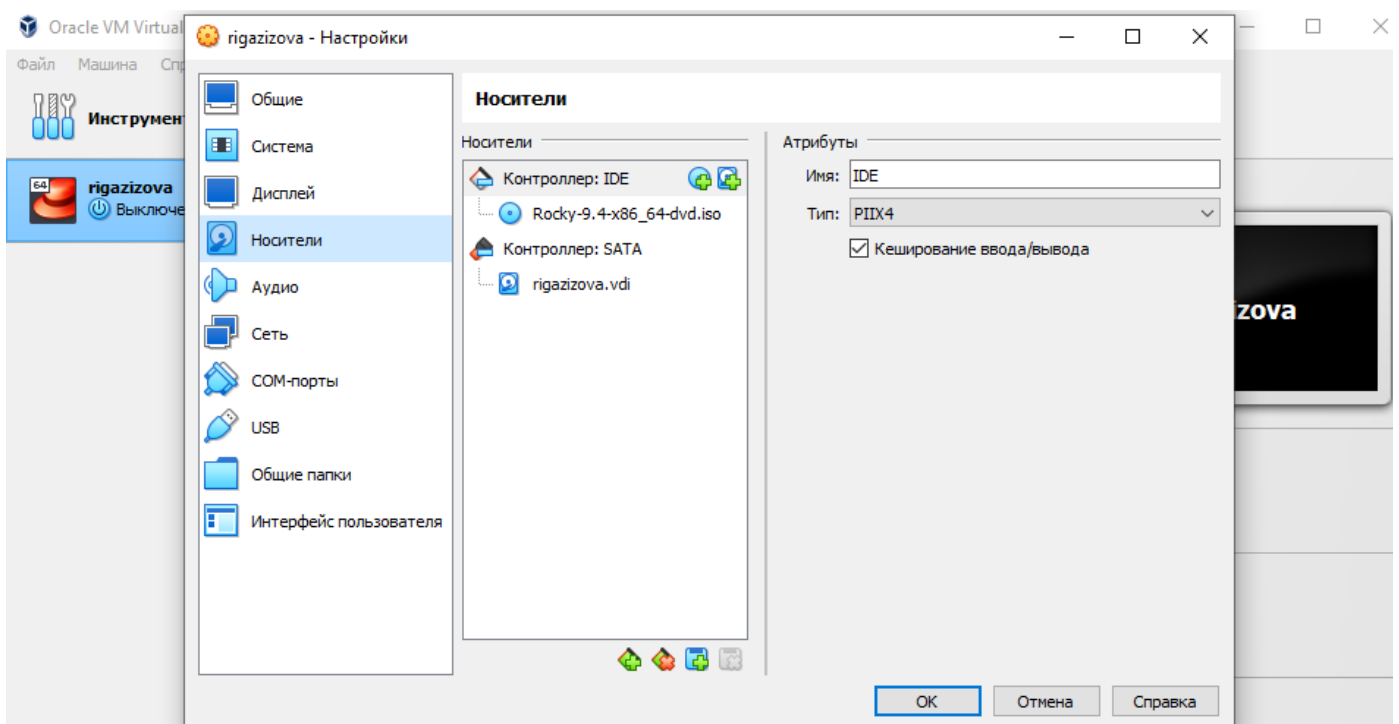
Создаем новый виртуальный диск с 60 ГБ памяти



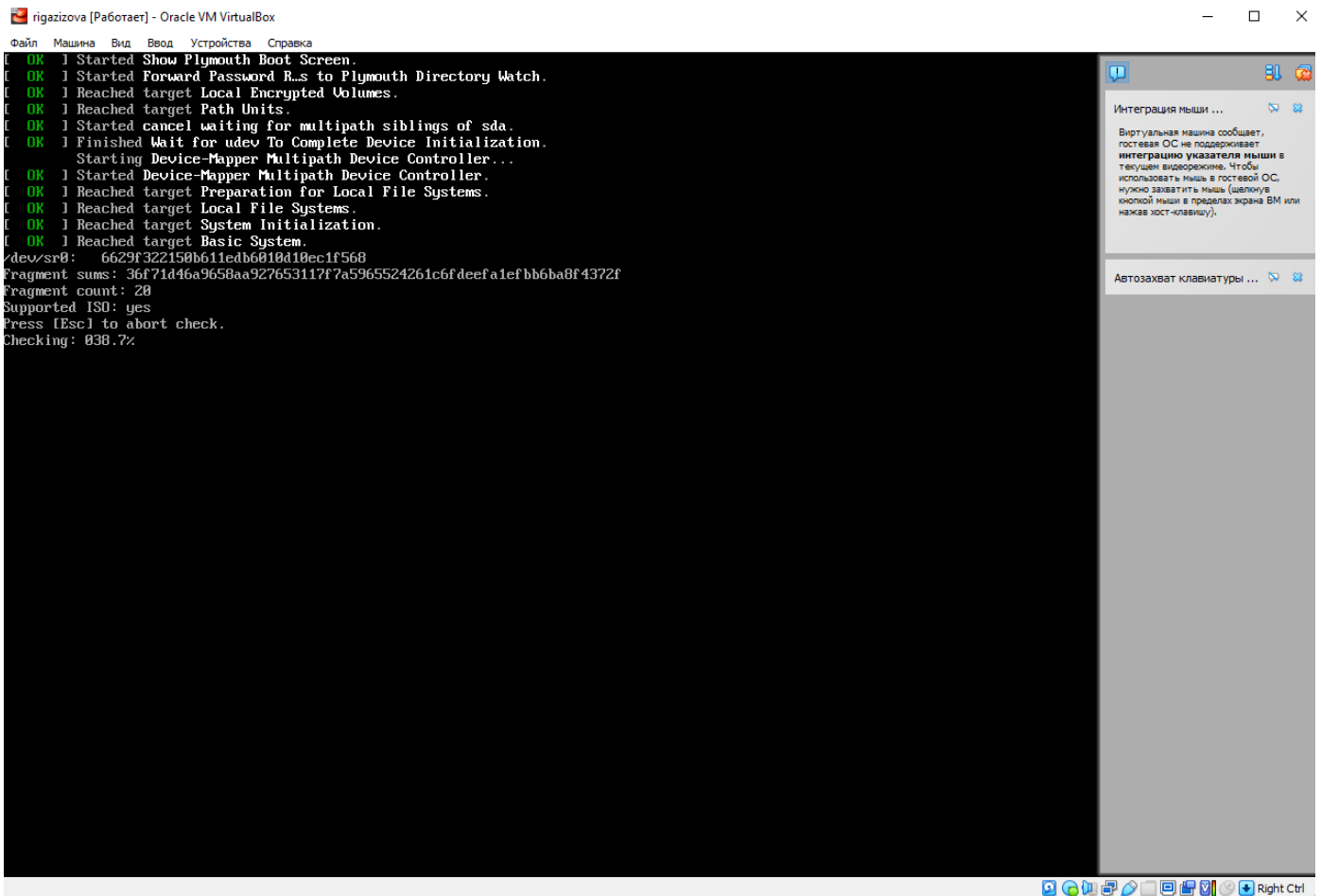
Получаем таблицу итоговых характеристик машины



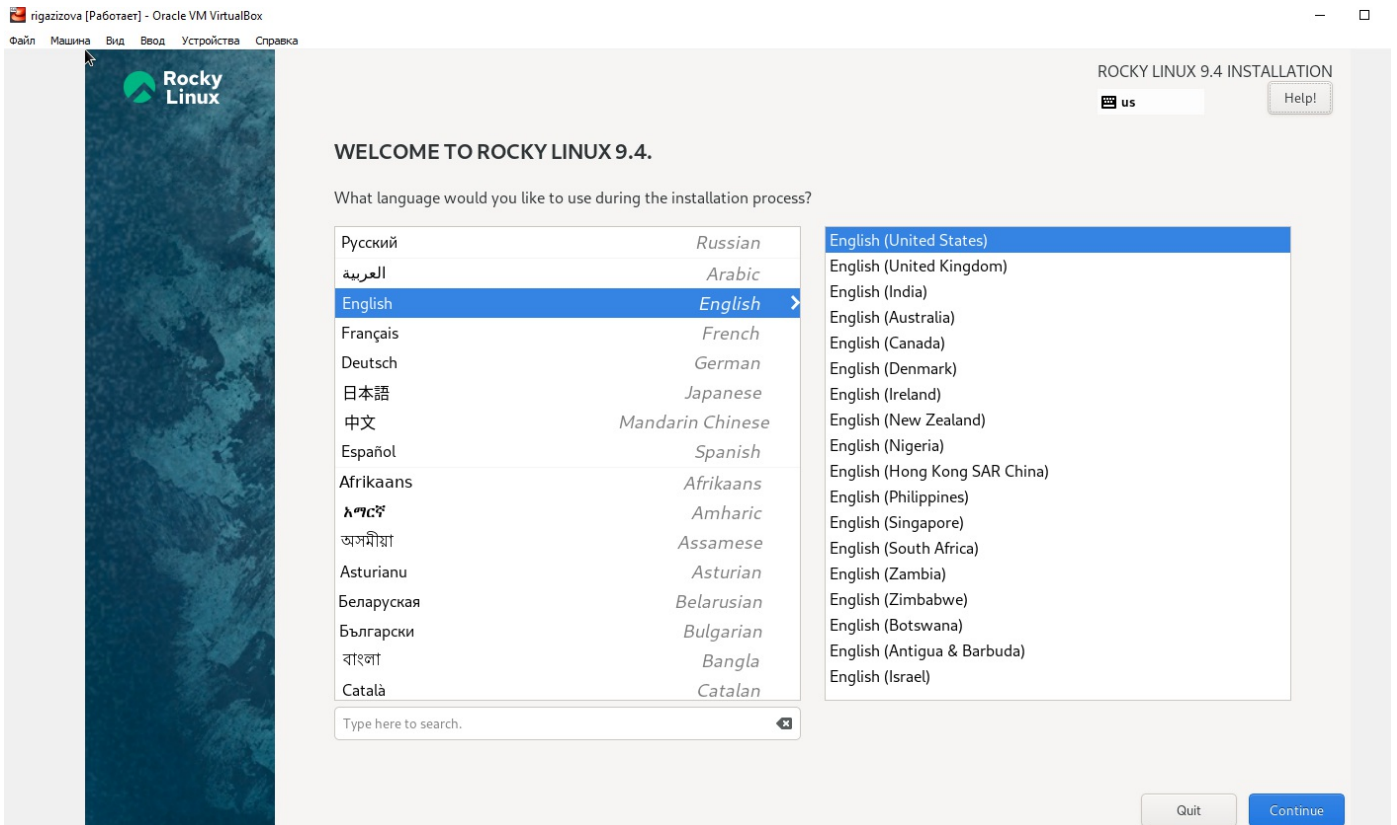
Добавляем новый привод оптических дисков и выбираем заранее скаченный образ операционной системы Rocky



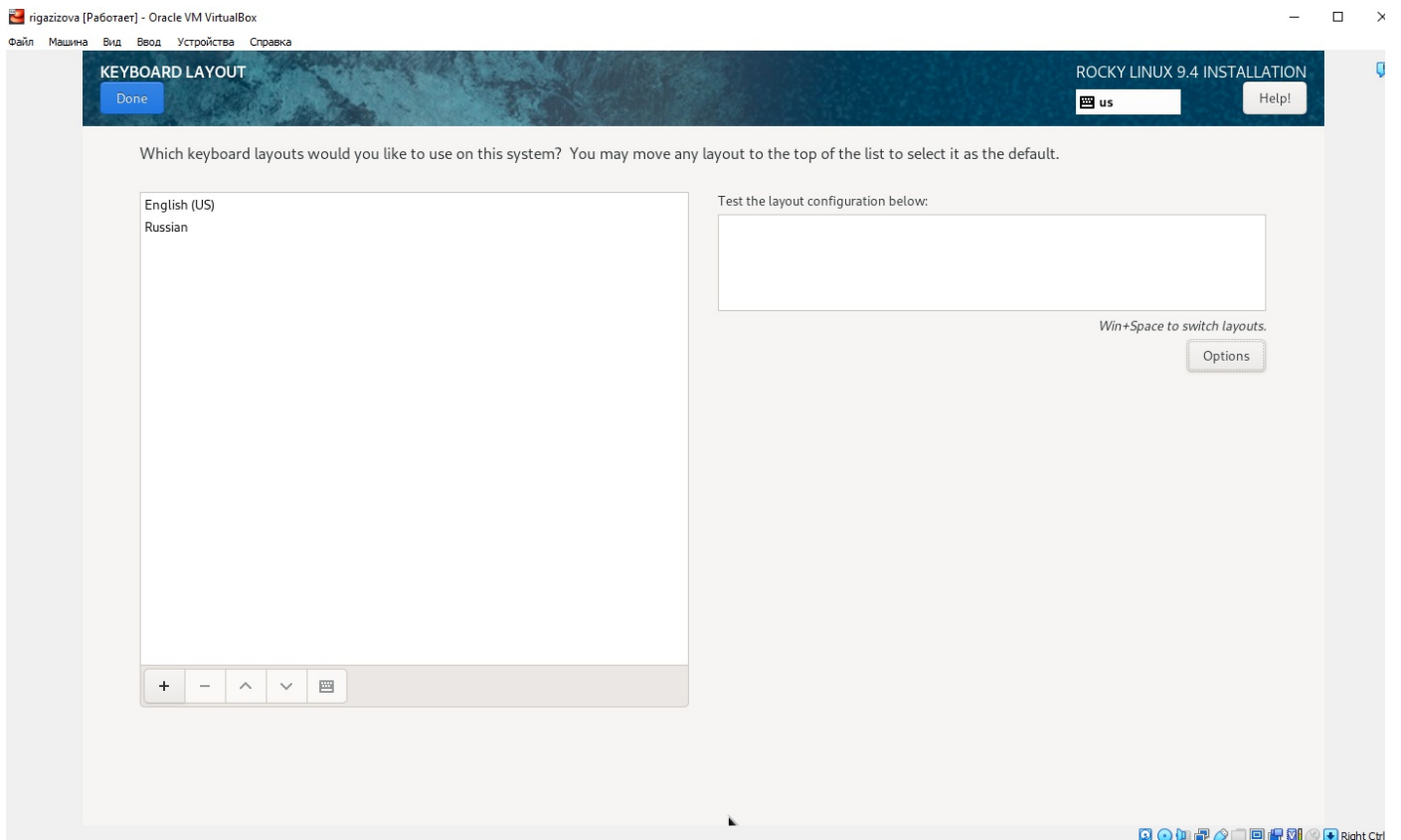
Начинается установка операционной системы



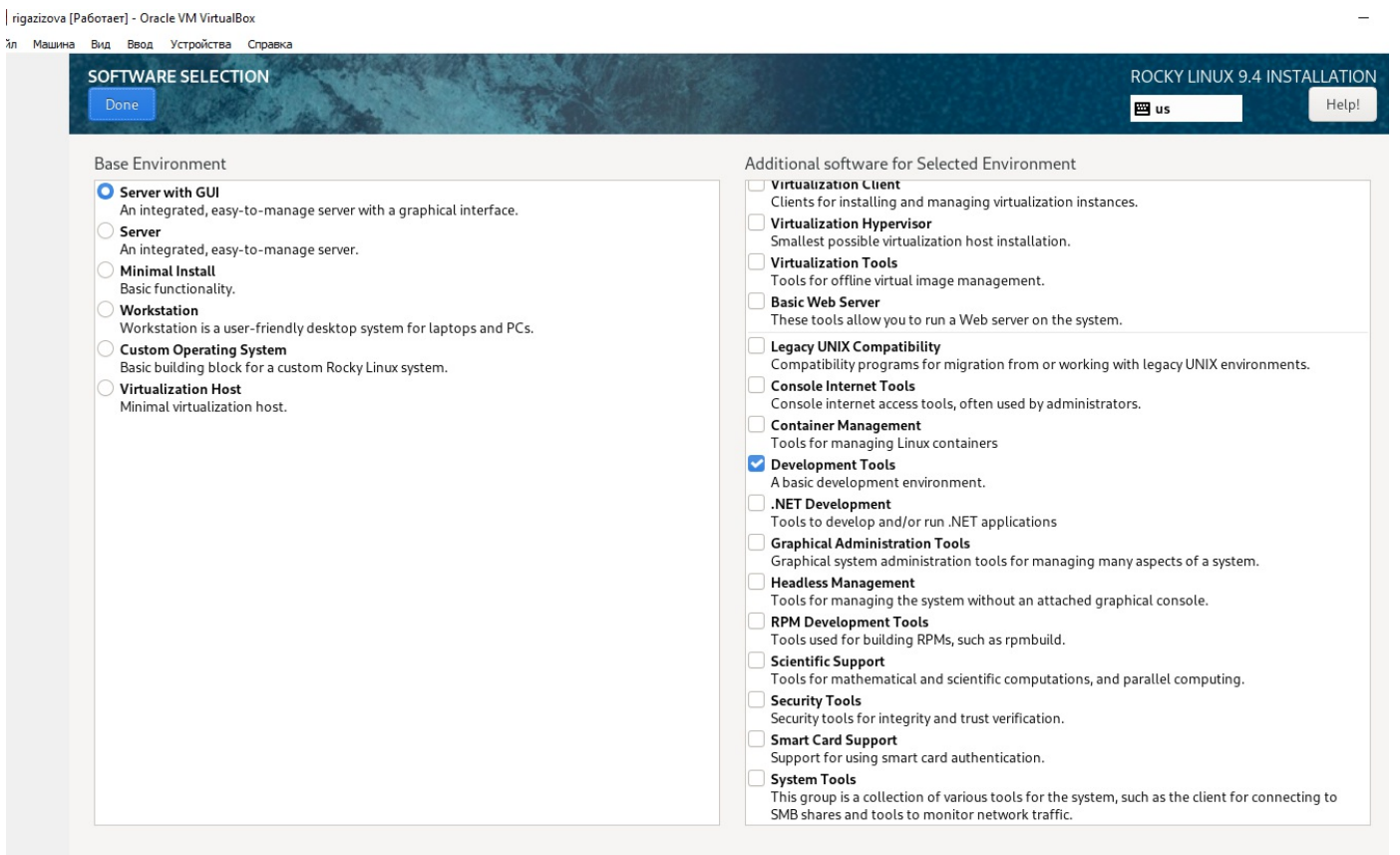
В настройках выбираем английский язык как основной



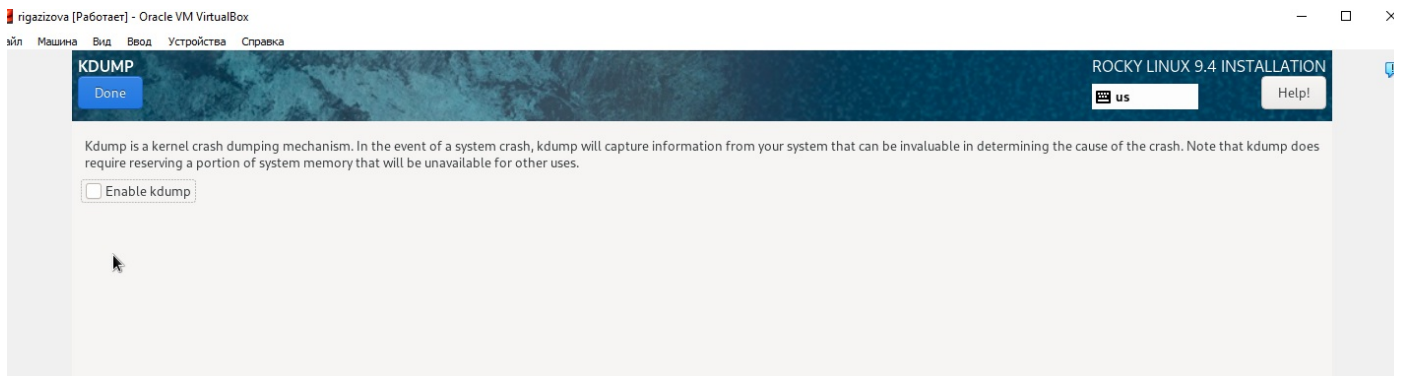
Добавляем к английскому русский и ставим комбинацию Win+Space для смены языка



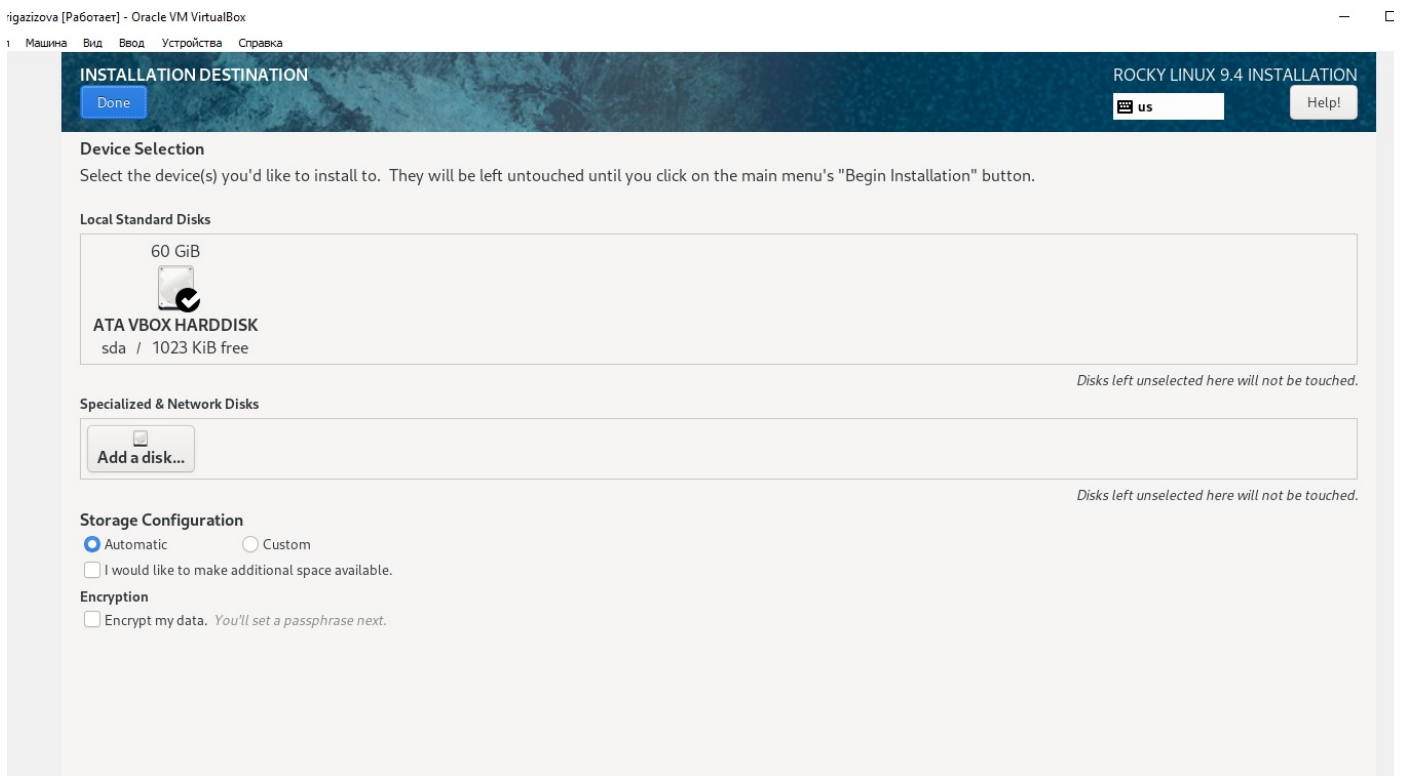
В разделе выбора программ указываем в качестве базового окружения Server with GUI , а в качестве дополнения — Development Tools



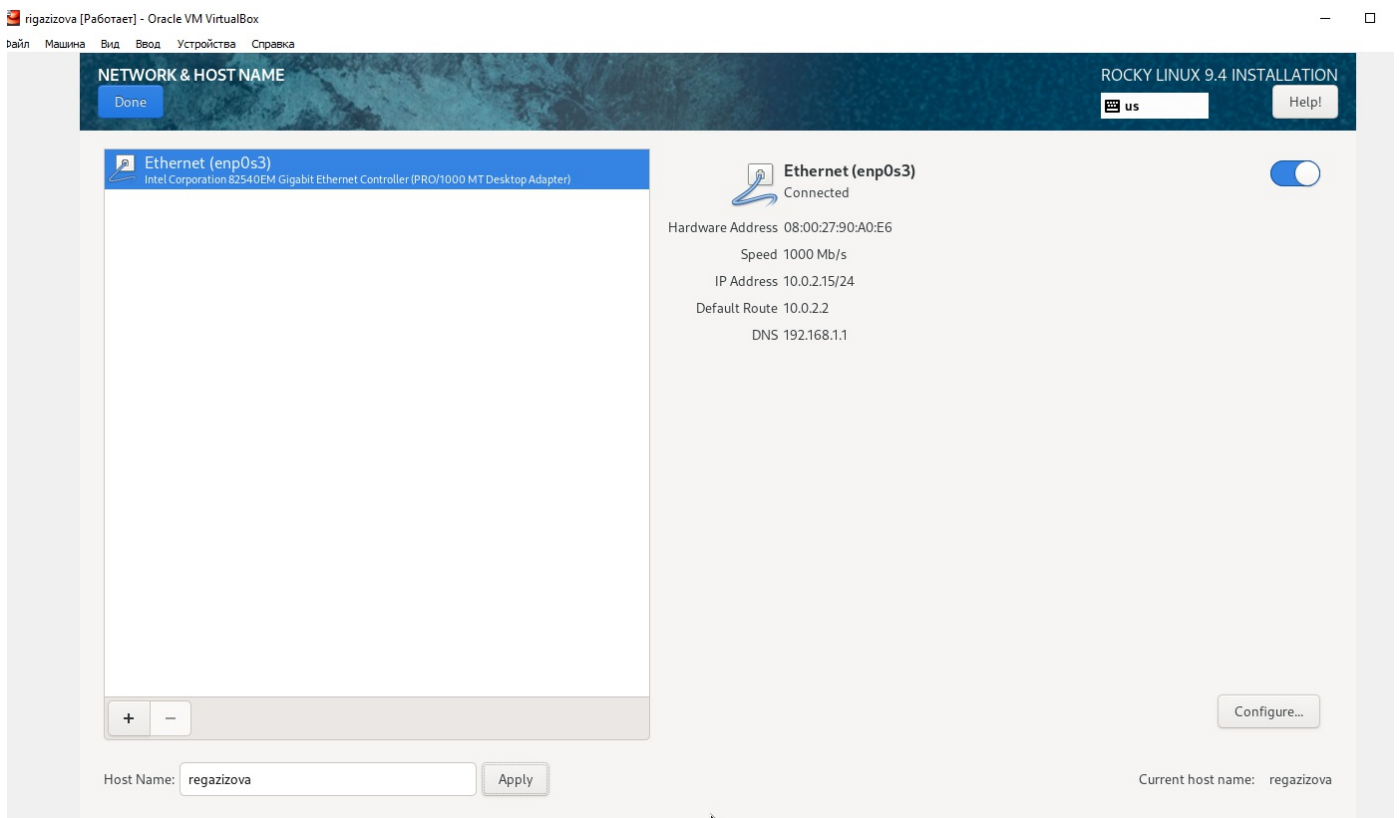
Отключаем KDUMP



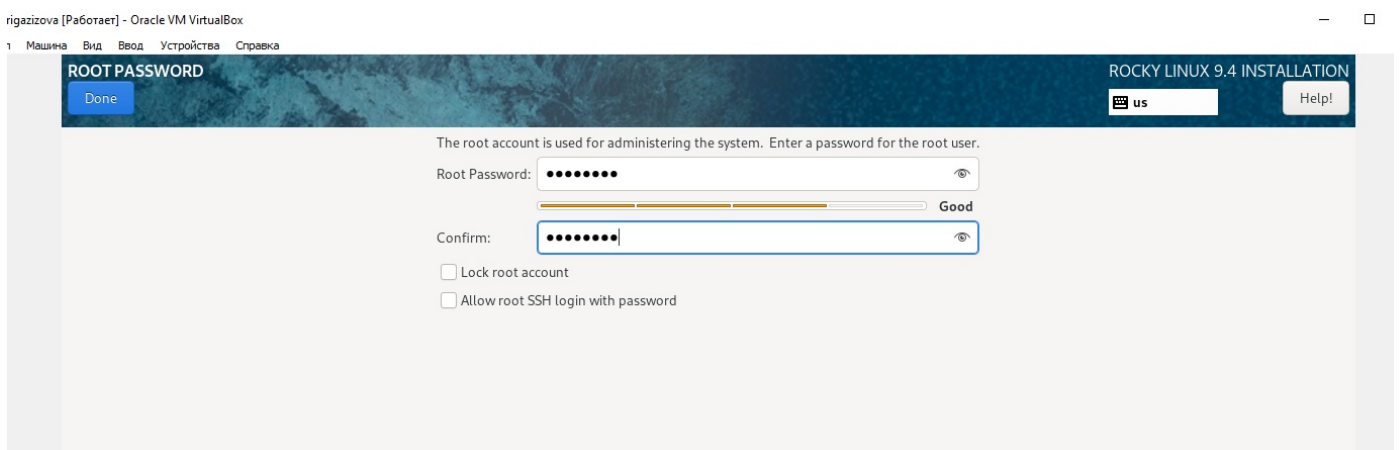
Место установки ОС оставляем неизменным



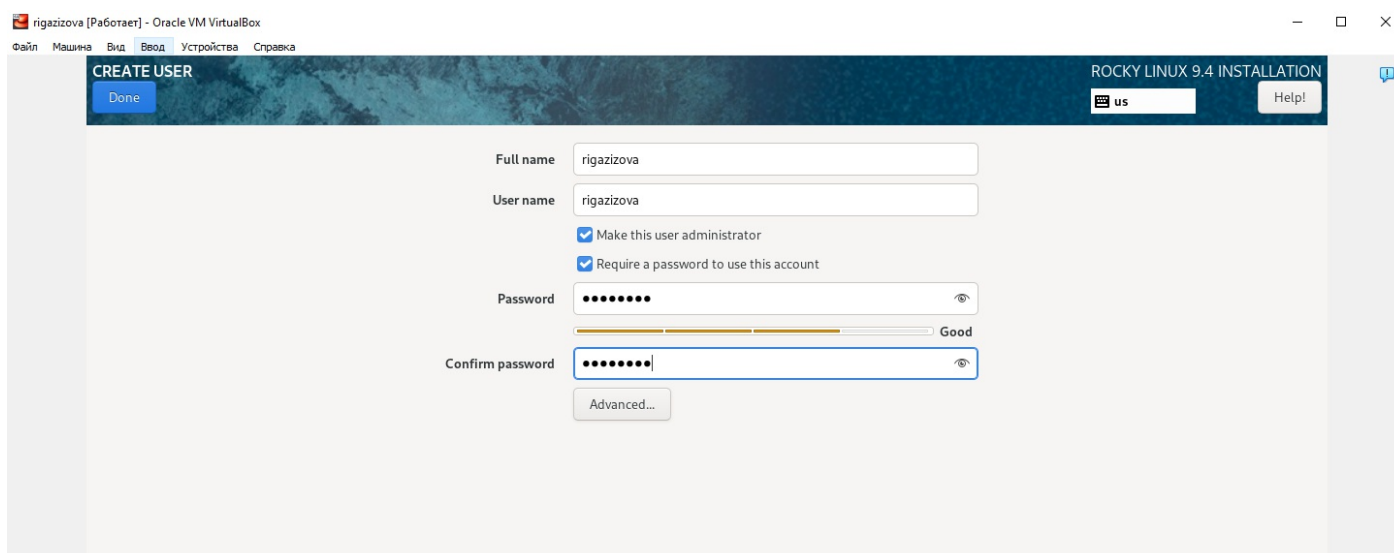
Включаем сетевое соединение и в качестве имени узла указываем rigazizova.localdomain



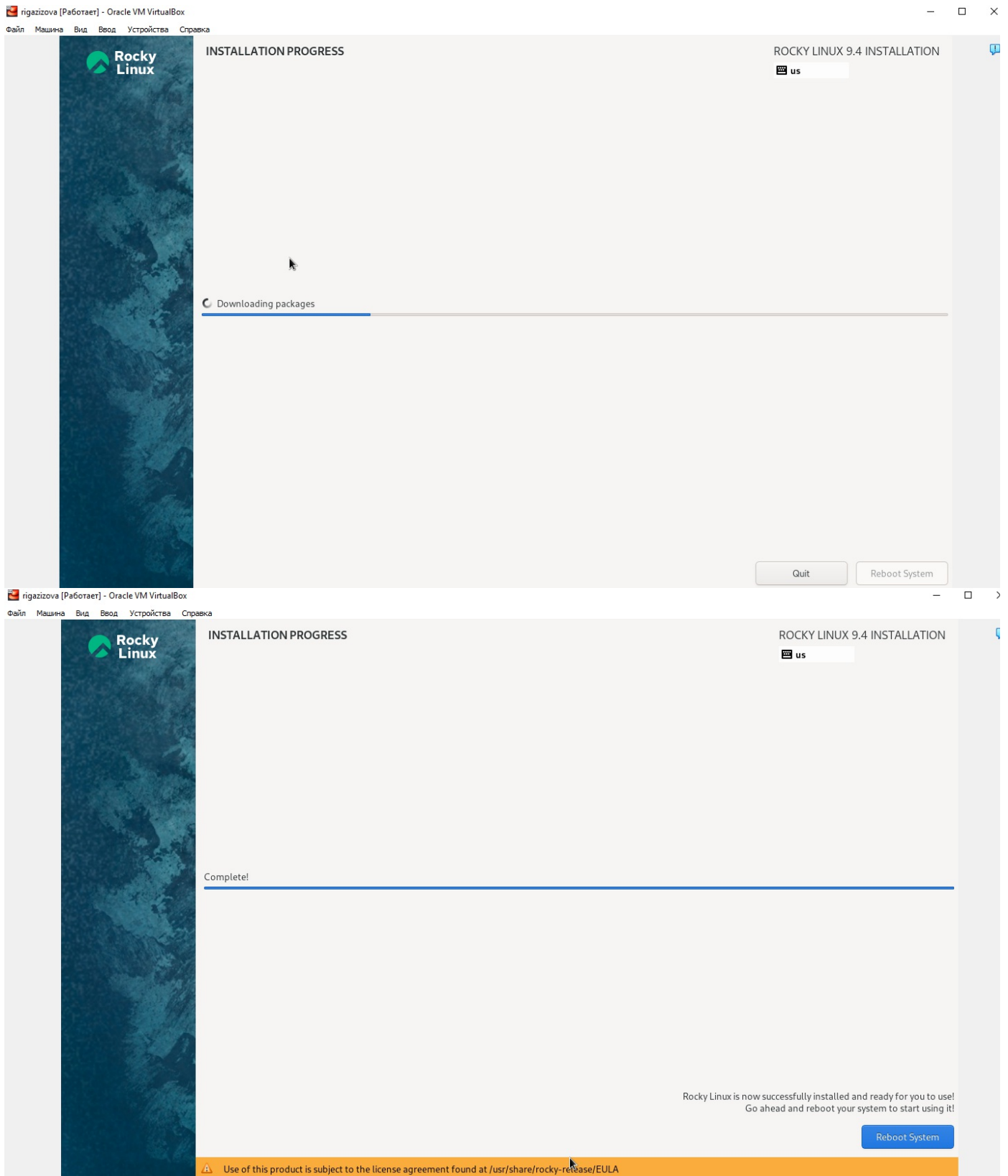
Устанавливаем пароль от root



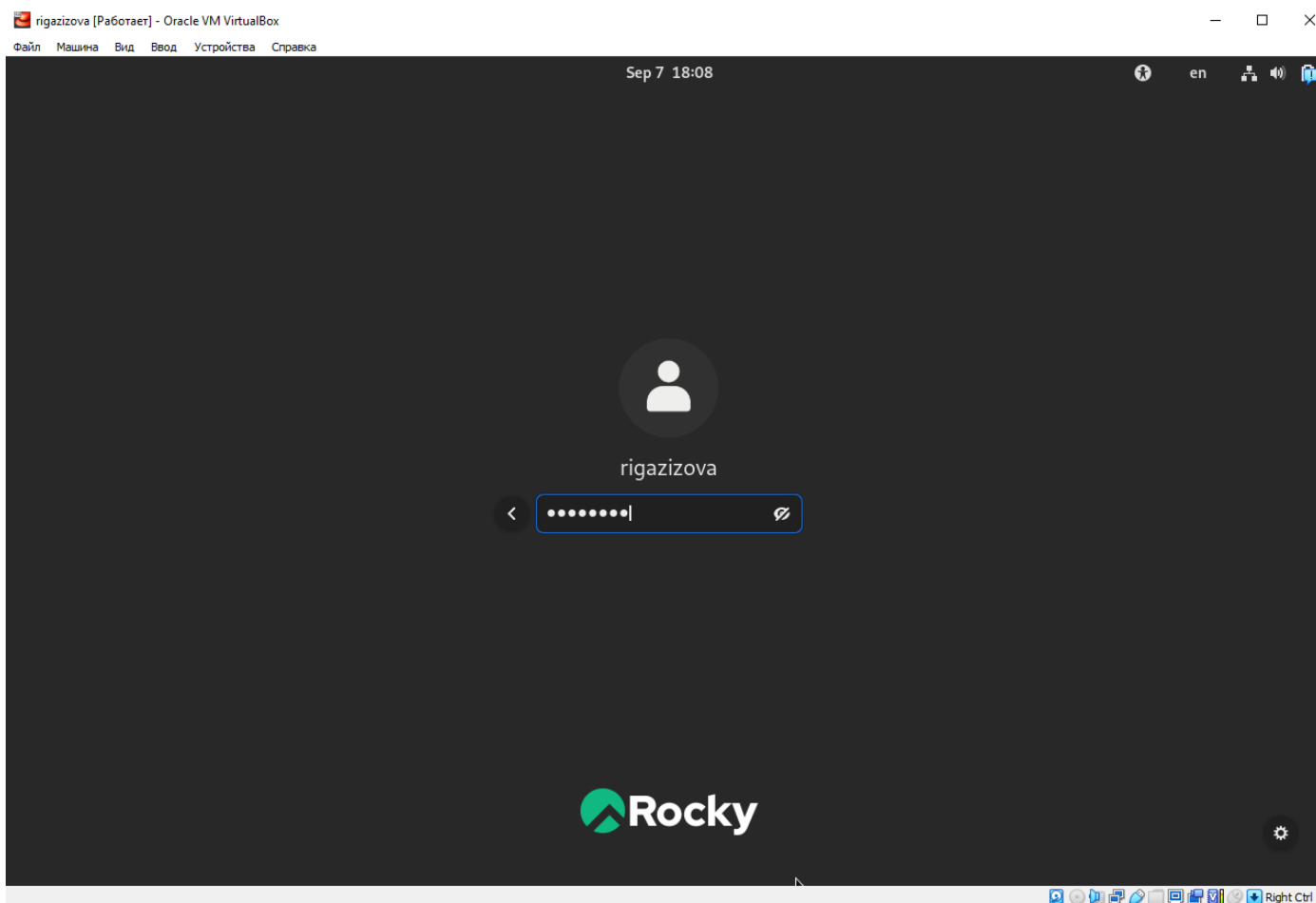
Создаем пользователя и пароль для него



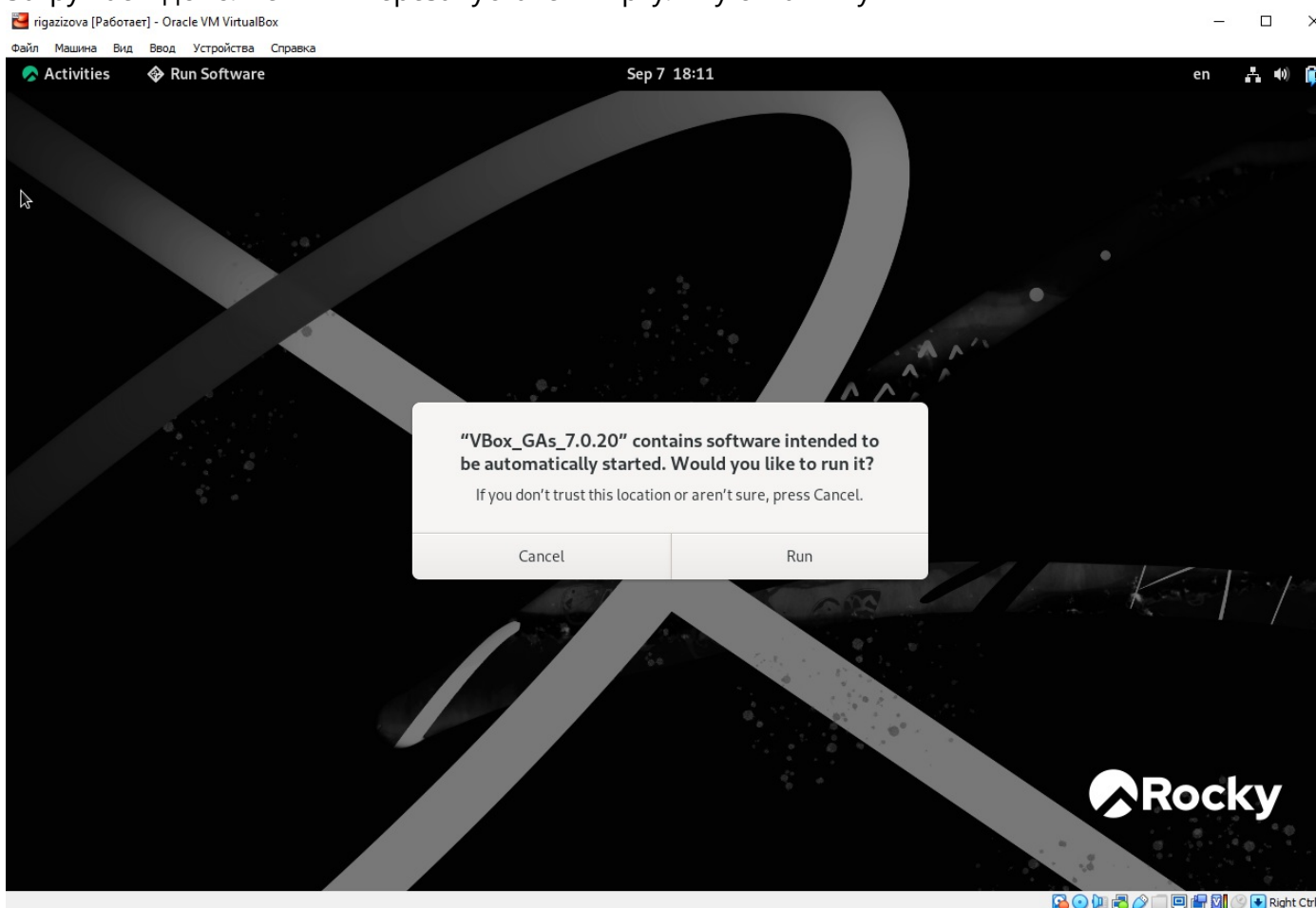
Завершаем установку ОС

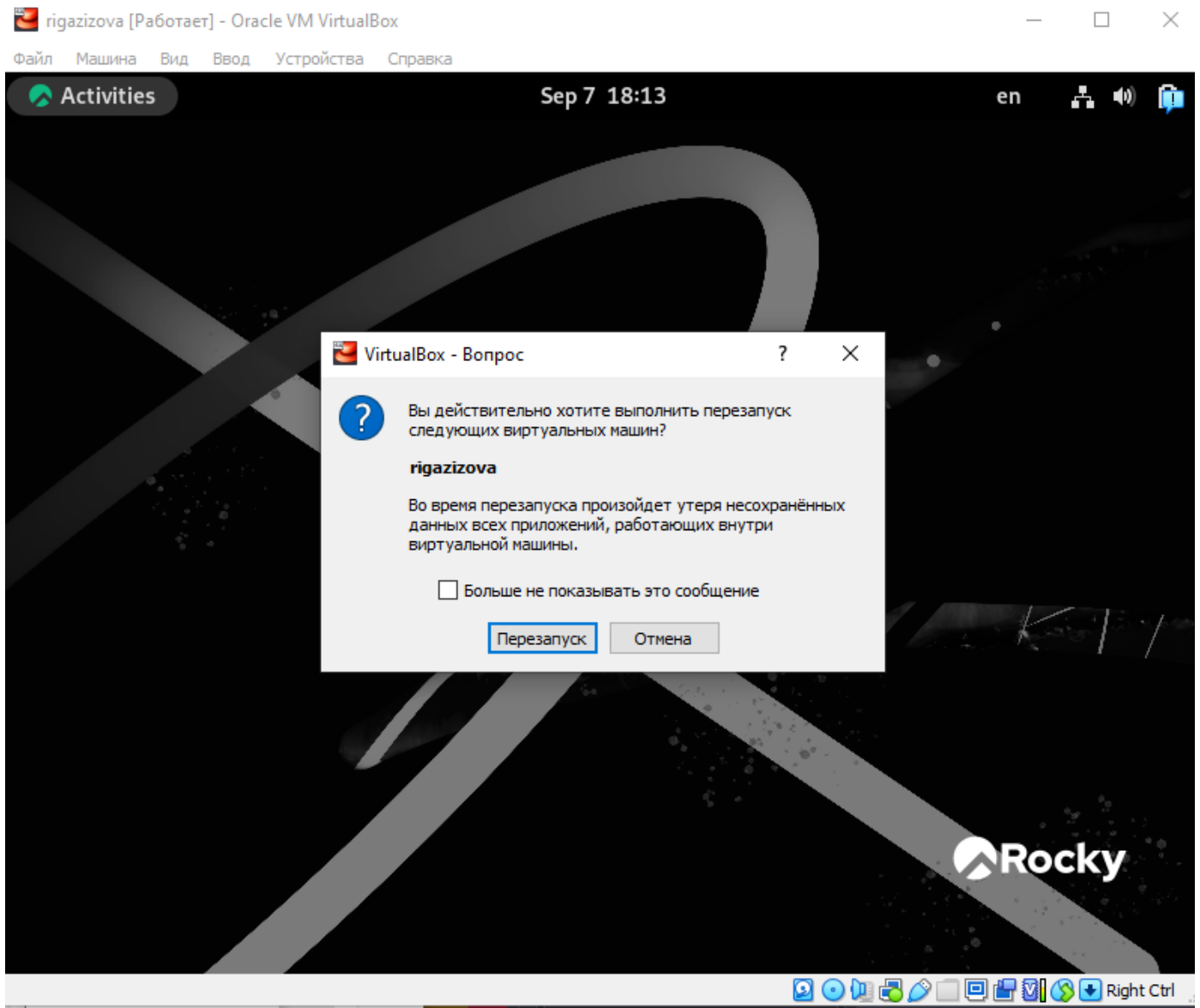


Входим в ОС под заданной нами при установке учётной записью



Загружаем дополнения и перезапускаем виртуальную машину





Домашнее задание

С помощью команд `dmesg | grep -i "то, что ищем"` Получили 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.

```
rigazizova@regazizova:~  
[rigazizova@regazizova ~]$ dmesg | grep "Linux version"  
[ 0.000000] Linux version 5.14.0-427.13.1.el9_4.x86_64 (mockbuild@iad1-prod-build001.bld.equ.rockylinux.org) (gcc (GCC) 11.4.1 20231218 (Red Hat 11.4.1-3), GNU  
ld version 2.35.2-43.el9) #1 SMP PREEMPT_DYNAMIC Wed May 1 19:11:28 UTC 2024  
[rigazizova@regazizova ~]$ dmesg | grep "Detected Mhz processor"  
[rigazizova@regazizova ~]$ dmesg | grep -i "Detected Mhz processor"  
[rigazizova@regazizova ~]$ dmesg | grep -i "Linux version"  
[ 0.000000] Linux version 5.14.0-427.13.1.el9_4.x86_64 (mockbuild@iad1-prod-build001.bld.equ.rockylinux.org) (gcc (GCC) 11.4.1 20231218 (Red Hat 11.4.1-3), GNU  
ld version 2.35.2-43.el9) #1 SMP PREEMPT_DYNAMIC Wed May 1 19:11:28 UTC 2024  
[rigazizova@regazizova ~]$ dmesg | grep -i "CPU0"  
[ 0.174735] smpboot: CPU0: 12th Gen Intel(R) Core(TM) i5-12450H (family: 0x6, model: 0x9a, stepping: 0x3)  
[rigazizova@regazizova ~]$ dmesg | grep -i "Memory available"  
[rigazizova@regazizova ~]$ dmesg | grep -i "Hypervisor detected"  
[ 0.000000] Hypervisor detected: KVM  
[rigazizova@regazizova ~]$ dmesg  
[ 0.000000] Linux version 5.14.0-427.13.1.el9_4.x86_64 (mockbuild@iad1-prod-build001.bld.equ.rockylinux.org) (gcc (GCC) 11.4.1 20231218 (Red Hat 11.4.1-3), GNU  
ld version 2.35.2-43.el9) #1 SMP PREEMPT_DYNAMIC Wed May 1 19:11:28 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=5.14.0-427.13.1.el9_4.x86_64 root=/dev/mapper/rl-root ro resume=/dev/mapper/rl-swap rd.lvm.lv=rl/root  
rd.lvm.lv=rl/swap rhgb quiet  
[ 0.000000] x86/fpu: Supporting XSAVE feature 0x001: 'x87 floating point registers'  
[ 0.000000] x86/fpu: Supporting XSAVE feature 0x002: 'SSE registers'  
[ 0.000000] x86/fpu: Supporting XSAVE feature 0x004: 'AVX registers'  
[ 0.000000] x86/fpu: xstate_offset[2]: 576, xstate_sizes[2]: 256  
[ 0.000000] x86/fpu: Enabled xstate features 0x7, context size is 832 bytes, using 'standard' format.  
[ 0.000000] signal: max sigframe size: 1776  
[ 0.000000] BIOS-provided physical RAM map:  
[ 0.000000] BIOS-e820: [mem 0x0000000000000000-0x000000000009fbff] usable  
[ 0.000000] BIOS-e820: [mem 0x000000000009fc00-0x000000000009ffff] reserved  
[ 0.000000] BIOS-e820: [mem 0x00000000000a0000-0x00000000000fffff] reserved  
[ 0.000000] BIOS-e820: [mem 0x0000000000100000-0x0000000000dfffff] usable  
[ 0.000000] BIOS-e820: [mem 0x0000000000dffff000-0x0000000000dfffff] ACPI data  
[ 0.000000] BIOS-e820: [mem 0x000000000fec00000-0x000000000fec00fff] reserved  
[ 0.000000] BIOS-e820: [mem 0x000000000fee00000-0x000000000fee00fff] reserved  
[ 0.000000] BIOS-e820: [mem 0x000000000fffc0000-0x000000000ffffff] reserved  
[ 0.000000] BIOS-e820: [mem 0x00000000100000000-0x000000001ffffff] usable  
[ 0.000000] NX (Execute Disable) protection: active  
[ 0.000000] SMBIOS 2.5 present.  
[ 0.000000] DMI: innotek GmbH VirtualBox/VirtualBox, BIOS VirtualBox 12/01/2006
```

Контрольные вопросы

1. Учетная запись пользователя - это необходимая для системы информация о пользователе, которая хранится в специальных файлах. Вся информация о пользователе обычно хранится в файлах /etc/passwd и /etc/group. Учетная запись пользователя содержит: имя пользователя, идентификационный номер группы, идентификационный номер пользователя, пароль, полное имя, домашний каталог, начальную оболочку.
2. Команды терминала:
 -
 - 1. Для получения справки по команде: man команда. Например, команда "man ls" выведет справку о команде "ls".
 -
 - 2. Для перемещения по файловой системе: cd путь. Например, команда "cd newdir" осуществляет переход в каталог newdir.
 -
 - 3. Для просмотра содержимого каталога: ls опции путь. Например, команда "ls -a ~/newdir" отобразит имена скрытых файлов в каталоге newdir.
 -
 - 4. Для определения объёма каталога: du опция путь. Например, команда "du -k ~/newdir" выведет размер каталога newdir в килобайтах.
 -
 - 5. Для создания / удаления каталогов / файлов: mkdir опции путь / rmdir опции путь / rm

опции путь. Например, команда "mkdir -p ~/newdir1/newdir2" создаст иерархическую цепочку подкаталогов, создав каталоги newdir1 и newdir2; команда "rmdir -v ~/newdir" удалит каталог newdir; команда "rm -r ~/newdir" так же удалит каталог newdir. 29

-
- 6. Для задания определённых прав на файл / каталог: chmod опции путь. Например, команда "chmod g+r ~/text.txt" даст группе право на чтение файла text.txt.
-
- 7. Для просмотра истории команд: history опции. Например, команда "history 5" покажет список последних 5 команд.
- 3. Файловая система имеет два значения: с одной стороны - это архитектура хранения битов на жёстком диске, с другой - это организация каталогов в соответствии с идеологией Linux. Файловая система - это архитектура хранения данных в системе, хранение данных в оперативной памяти и доступа к конфигурации ядра. В физическом смысле файловая система Linux представляет собой пространство раздела диска, разбитое на блоки фиксированного размера. Их размер кратен размеру сектора: 1024, 2048, 4096 или 8120 байт. Примеры файловых систем:
-
- 1. XFS рассчитана на файлы большого размера, поддерживает диски до 2 терабайт. Преимущества: высокая скорость работы с большими файлами, отложенное выделение места, увеличение разделов на лету, незначительный размер служебной информации. Недостатки: невозможность уменьшения размера, сложность восстановления данных и риск потери файлов при аварийном отключении питания.
-
- 2. Ext2, Ext3, Ext4 или Extended Filesystem - стандартная файловая система, первоначально разработанная еще для Minix. Содержит максимальное количество функций и является наиболее стабильной в связи с редкими изменениями кодовой базы. Начиная с ext3 в системе используется функция журналирования. Сегодня версия ext4 присутствует во всех дистрибутивах Linux.
-
- 3. JFS или Journaled File System разработана в IBM в качестве альтернативы для файловых систем ext. Сейчас используется там, где необходима высокая стабильность и минимальное потребление ресурсов (в первую очередь в многопроцессорных компьютерах). В журнале хранятся только метаданные, что позволяет восстанавливать старые версии файлов после сбоев.
- 4. Команда "findmnt" или "findmnt --all" будет отображать все подмонтированные файловые системы или искать файловую систему.
- 5. Команда "kill -сигнал pid_процесса" позволяет удалить зависший процесс, где PID -

уникальный идентификатор процесса. Сигналы могут быть следующие:

- 1. SIGINT - самый безобидный сигнал завершения, означает Interrupt. Он отправляется процессу, запущенному из терминала с помощью сочетания клавиш Ctrl+C. Процесс правильно завершает все свои действия и возвращает управление
- 2. SIGQUIT - сигнал, который отправляется с помощью сочетания клавиш, программе, запущенной в терминале. Он сообщает ей, что нужно завершиться, и программа может выполнить корректное завершение или проигнорировать сигнал. В отличие от предыдущего, она генерирует дампы памяти. Сочетание клавиш Ctrl+Q
- 3. SIGHUP - сообщает процессу, что соединение с управляющим терминалом разорвано, отправляется, в основном, системой при разрыве соединения с интернетом.
- 4. SIGKILL - тоже немедленно завершает процесс, но, в отличие от предыдущего варианта, он не передается самому процессу, а обрабатывается ядром. Поэтому ресурсы и дочерние процессы остаются запущенными

Выводы

Приобрели практические навыки установки операционной системы на виртуальную машину и настройки минимально необходимых для дальнейшей работы сервисов.