Отчет по лабораторной работе №1

Дисциплина: Информационная безопасность

Выполнила: Губина Ольга Вячеславовна

Содержание

# 1 Цель работы

### 1.0.1 Цель работы:

Создание репозитория дисциплины на github.com в соответствие с требованиями по шаблону и соблюдением соглашения наименований, соглашения для путей к файлам; Создание виртуальной машины в соответствии с соглашением о наименовании типа RedHat для возможность осуществления дальнейшей работы на курсе.

# 2 Задание

1. Создать рабочую директорию и репозиторий дисциплины на github.com в соответствие с:

* шаблоном;
* соглашением наименований;

1. Создать виртуальную машину.

# 3 Теоретическое введение

В данной работе мы будем взаимодействовать с ***github.com***, где будем выгружать лабораторные работы.

**Git** — распределённая система управления версиями. Проект был создан *Линусом Торвальдсом* для управления разработкой ядра Linux, первая версия выпущена 7 апреля 2005 года. На сегодняшний день его поддерживает Джунио Хамано.

Для того, чтобы git узнал наше имя и электронную почту, необходимо выполнить следующие действия в командной строке:

git config --global user.name "Your Name" git config --global user.email "your\_email@whatever.com"

Однако, с прошлого года сохранились некоторые данные, поэтому в моем случае этого делать не пришлось.

Отчеты по лабораторным работам данной дисциплины будем писать с использованием ***markdown***.

**Markdown** — облегчённый язык разметки, созданный с целью обозначения форматирования в простом тексте, с максимальным сохранением его читаемости человеком, и пригодный для машинного преобразования в языки для продвинутых публикаций.

Теперь в табл. [1](#tbl:1) рассмотрим основные сведения и функции языка разметки.

Table 1: Описание некоторых функций языка разметки markdown

| Символьное обозначение в коде | Описание |
| --- | --- |
| # | Создание заголовка первого порядка (всего порядков шесть, кол-во символов # обозначаем номер порядка) |
| \*\* \*\* | Задает полужирное начертание текста |
| \* \* | Задает курсивное начертание текста |
| \*\*\* \*\*\* | Задает курсивное полужирное начертание текста |
| - или \* | Задает элемент неупорядоченного маркированного списка |
| ``` ``` | Задает выделенные блоки кода |

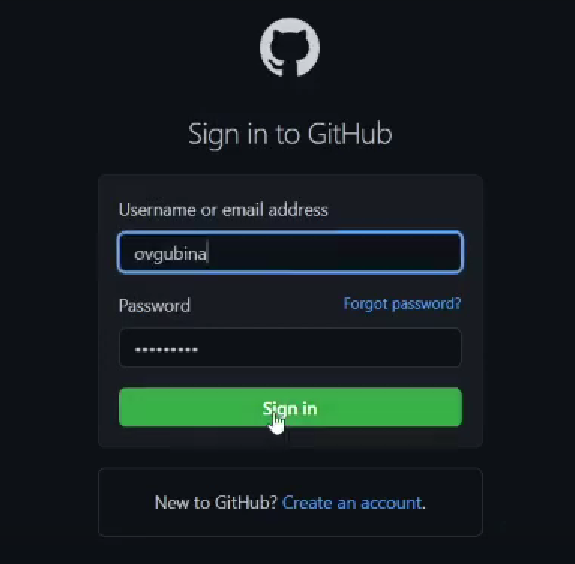
Более подробно о git и markdown см. в [1–4].

Также будем использовать дистрибутив Rocky Linux. **Rocky Linux** — это дистрибутив Linux, разработанный *Rocky Enterprise Software Foundation*. Предполагается, что это будет полный бинарно-совместимый выпуск, использующий исходный код операционной системы **Red Hat** Enterprise Linux.

Некоторые образы ISO, выпущенные Rocky Enterprise Software Foundation, не имеют прямых эквивалентов вышестоящего уровня. Они создаются для определенных целей, например, для предоставления живого загрузочного образа или для предоставления установочного носителя уменьшенного размера [5].

# 4 Выполнение лабораторной работы

1. Авторизуемся на github.com (рис. [??]).



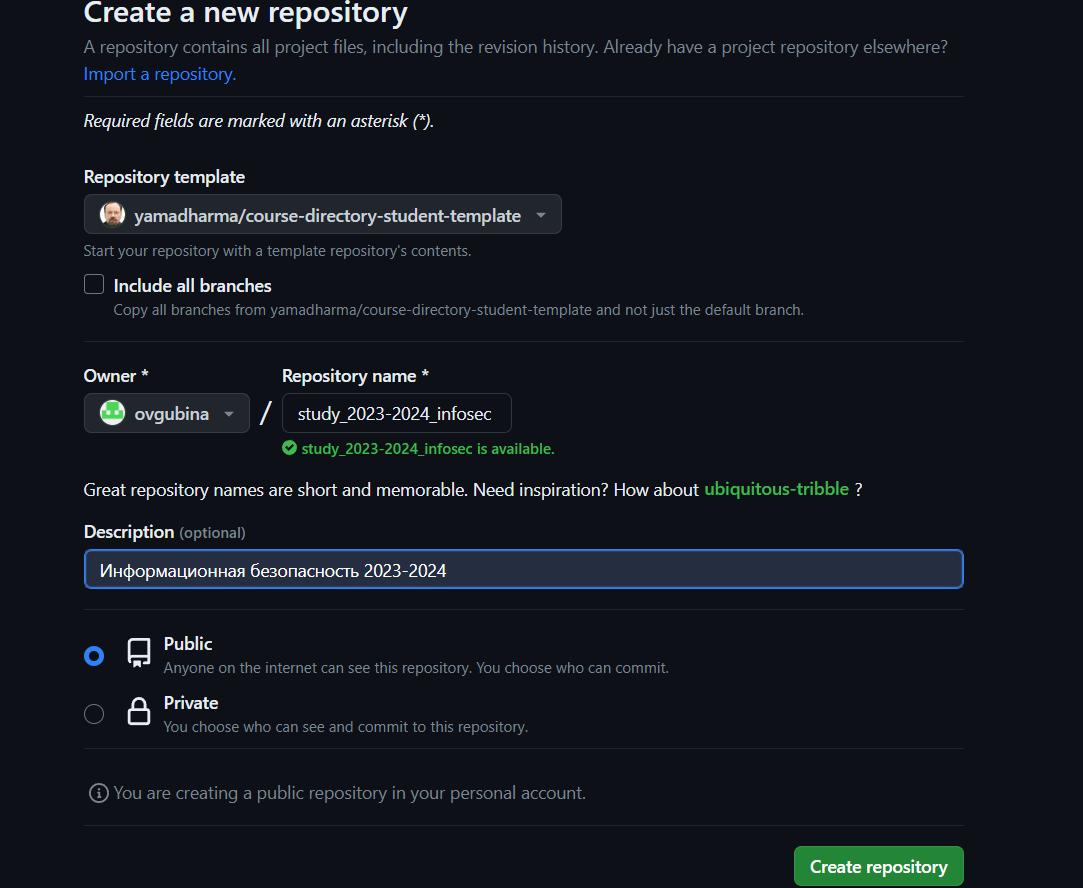
Авторизация на github.com

1. Далее нам необходимо создать каталог, где будут располагаться все лабораторные работы, в моем случае: /d/work/2023-2024/Information\_security(рис. ??). Полное название предмета написано на латиницей, посколько при попытке сделать мейк возникала ошибка, то же и с пробелом - вместо него нижнее подчеркивание.

Создание каталога work/2023-2024/Information_security

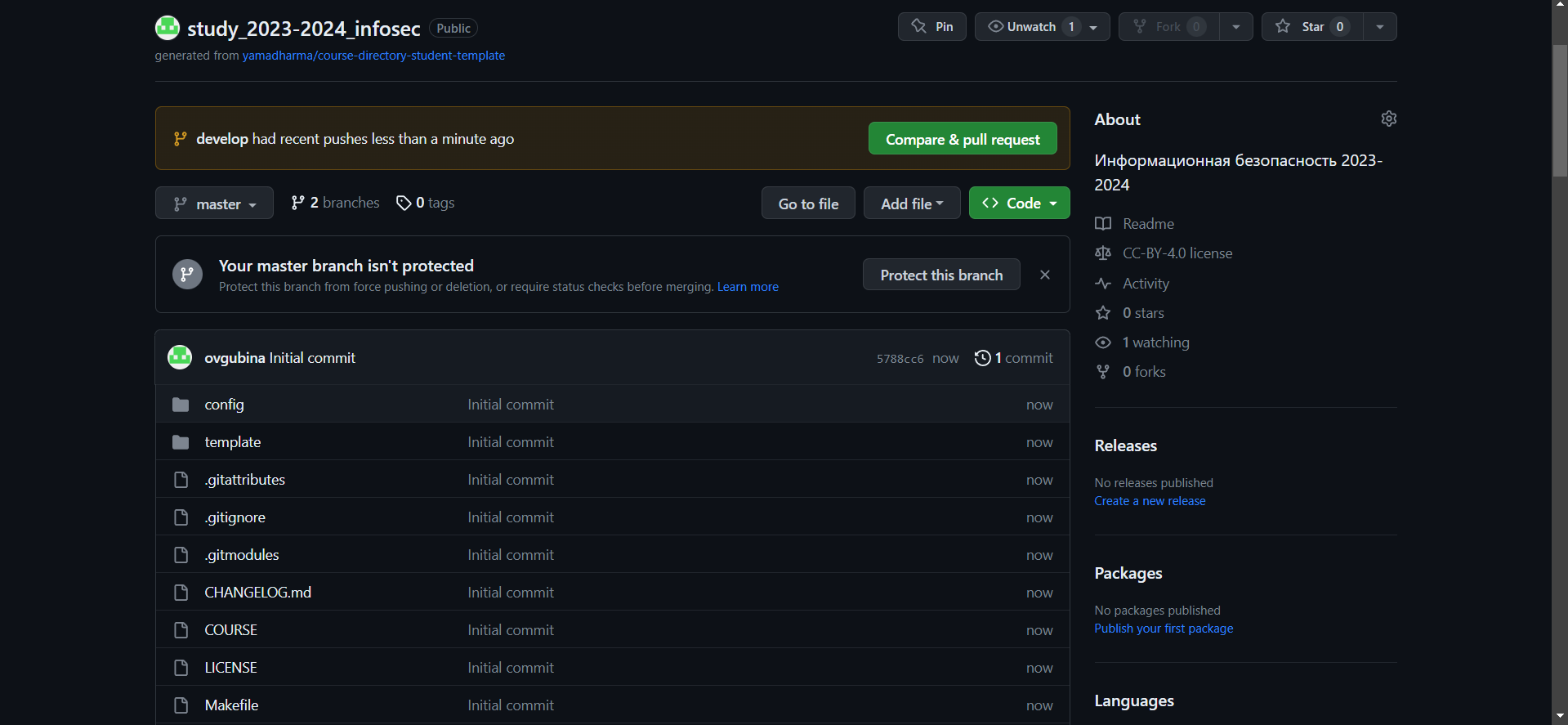
Создание каталога work/2023-2024/Information\_security

1. Создадим репозиторий по предложенному шаблону вручную через <git.com> с наименованием study\_2023-2024\_infosec (рис. [??]).



Создание репозитория по шаблону

Видим, что репозиторий успешно создан в нашем профиле (рис. [??]).

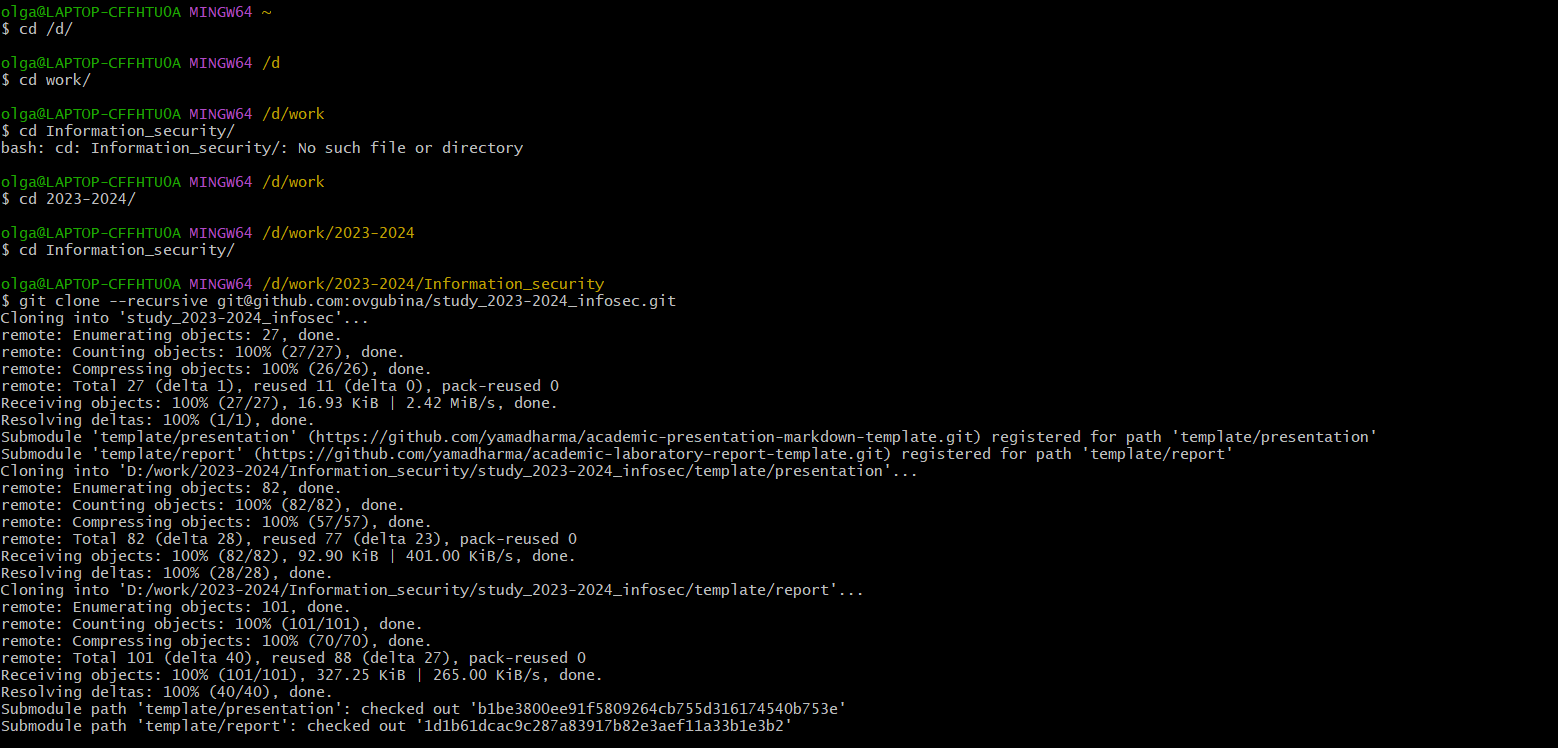


Создание репозитория по шаблону

1. Через Git Bash подключим репозиторий к директории.

С прошлого предмента все ключи у меня сохранились, поэтому сначала переходим в наш рабочий каталог и выполняем команду (рис. [??]):

git clone --recursive <repo\_name\_ssh>



Клонирование репозитория

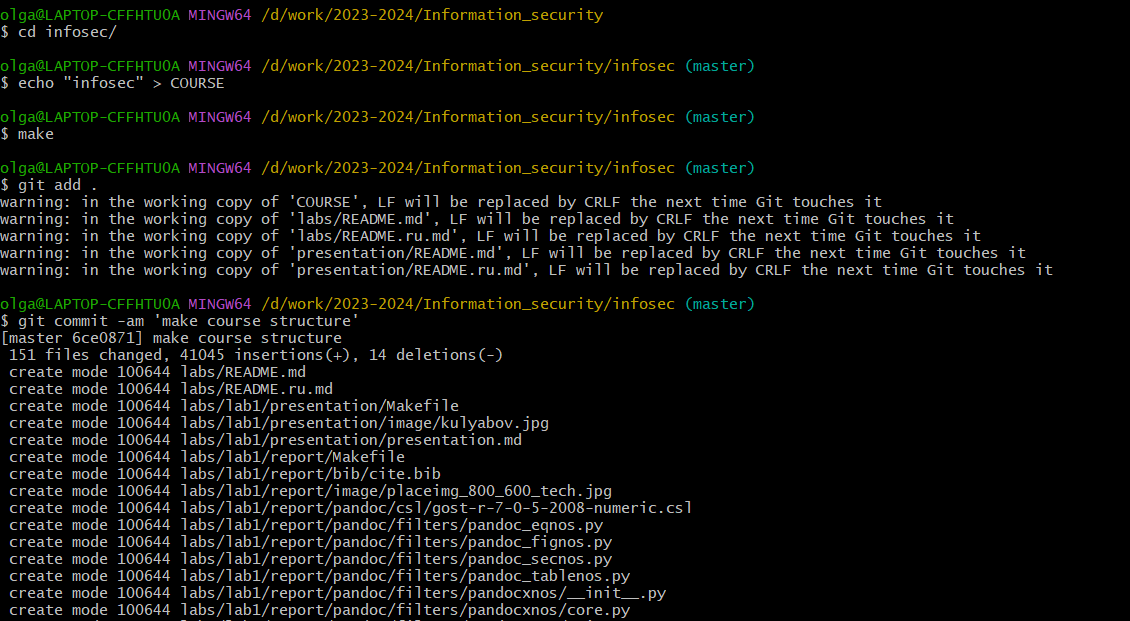
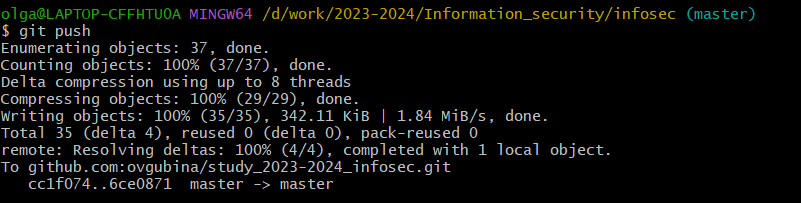
После этого действия в рабочем каталоге появляется папка согласно названиию репозитория - study\_2023-2024\_infosec, для удобства я переименовала эту папку в infosec.

Далее мы переходим в созданную папку в Git Bash (предвалительно я вручную удалила файл package.json) и создаем необходимые каталоги (рис. [??]):

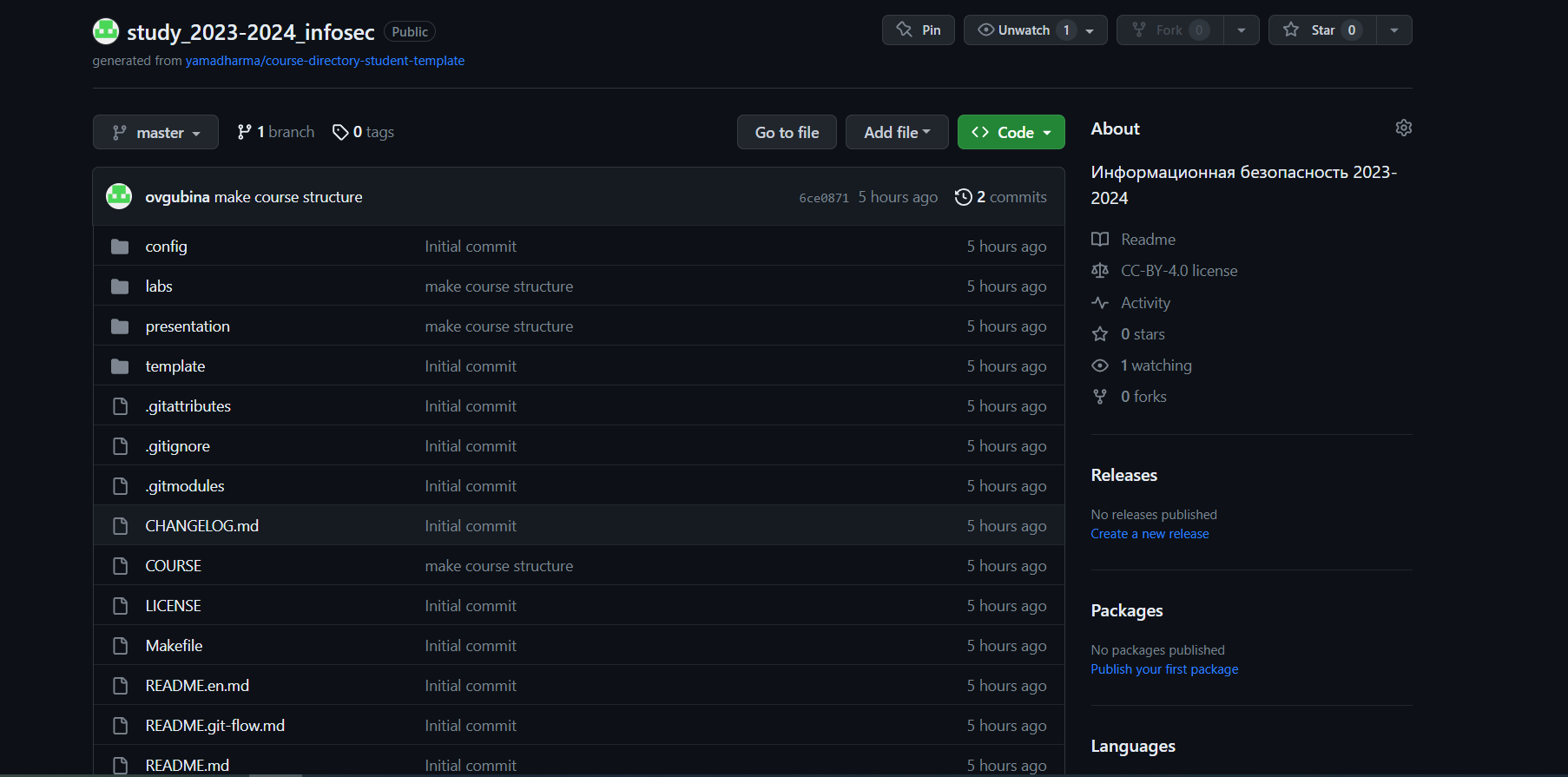
echo "infosec" > COURSE  
make

После этого Отправляем файлы на сервер (рис. [??]-[??]):

git add .  
git commit -am '<commit\_name>'  
git push

Переходим на github и видим, что создалась папка labs - изменения внесены успешно (рис. [??]):

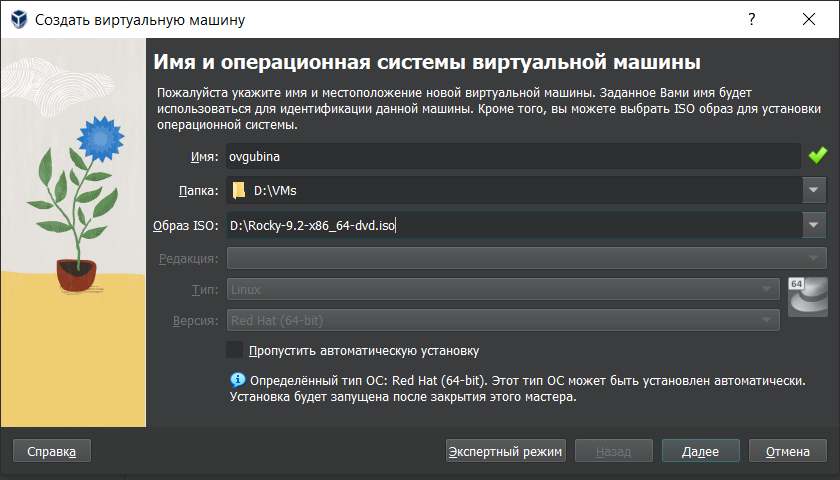


Проверка git

1. Теперь перейдем к созданию виртуальной машины.

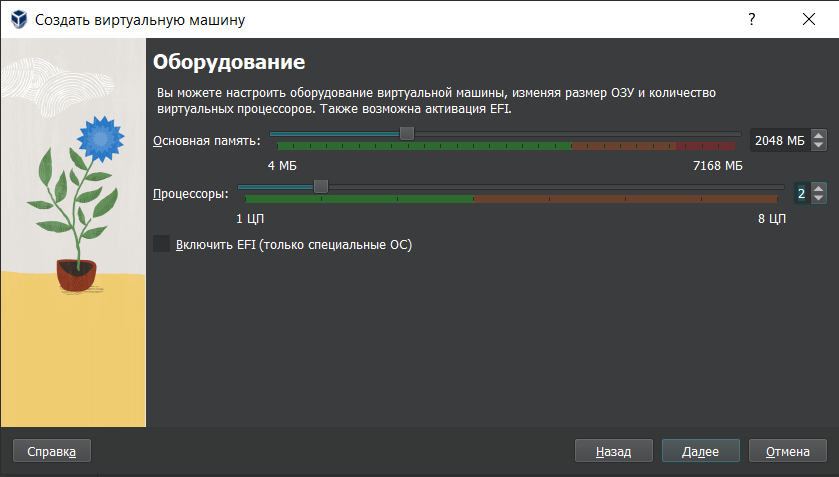
Для этого предварительно был скачан образ Rocky Linux 9.2 dvd.

Создаем виртуальную машину в соответствие с соглашением о наименовании, на диск D, используя установленный образ (рис. [??]):

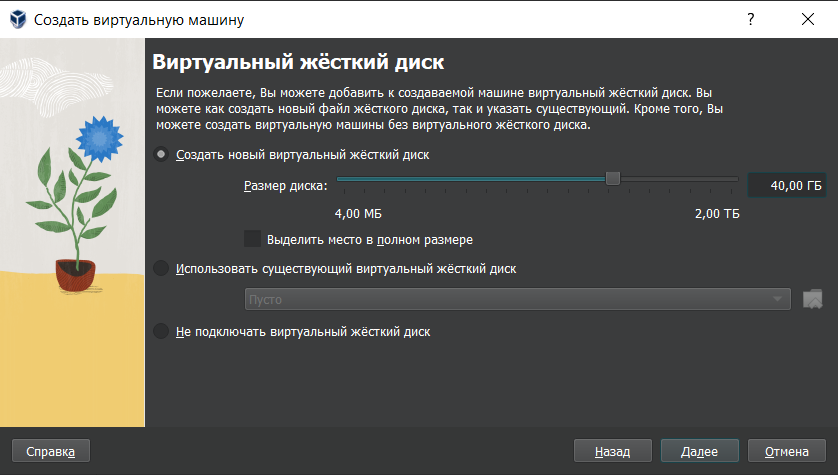


Создание виртуальной машины

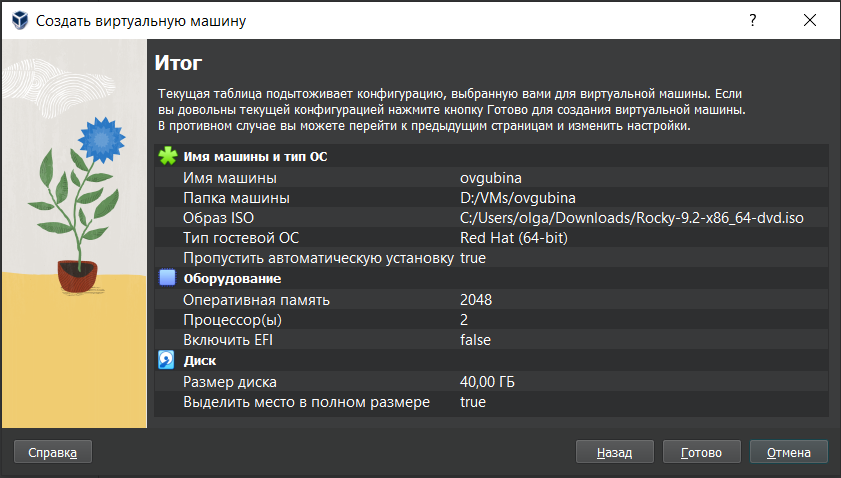
Выбираем параметры будущей машины - основная память 2Гб, 2 процессора, размер виртуального жесткого диска 40Гб (рис. [??]-[??])



Параметры ВМ

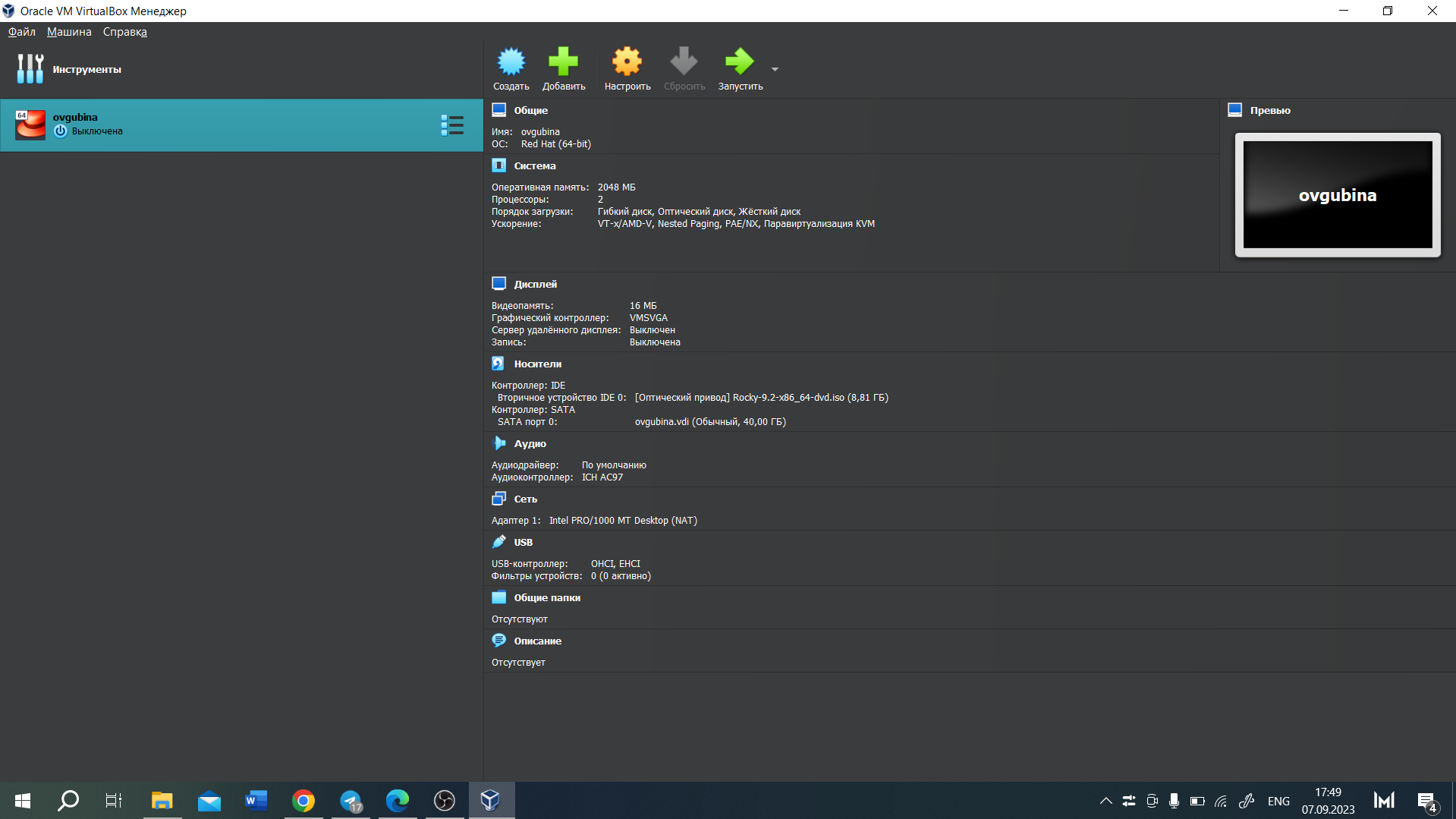


Параметры ВМ



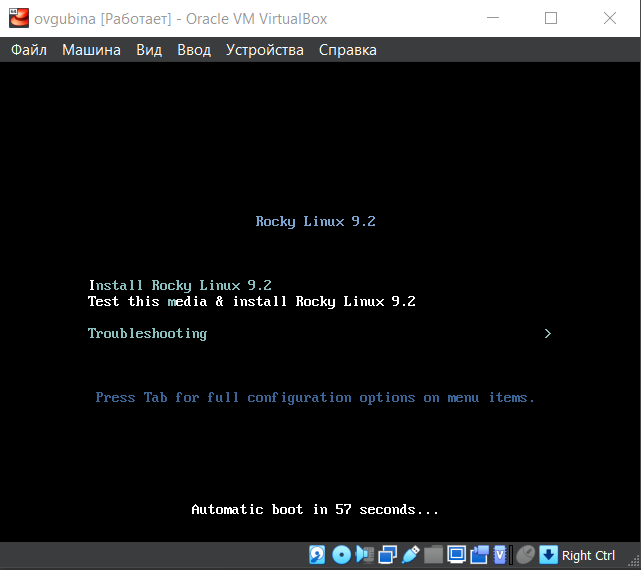
Итог - параметры ВМ

Виртуальная машина успешно создана (рис. [??]):



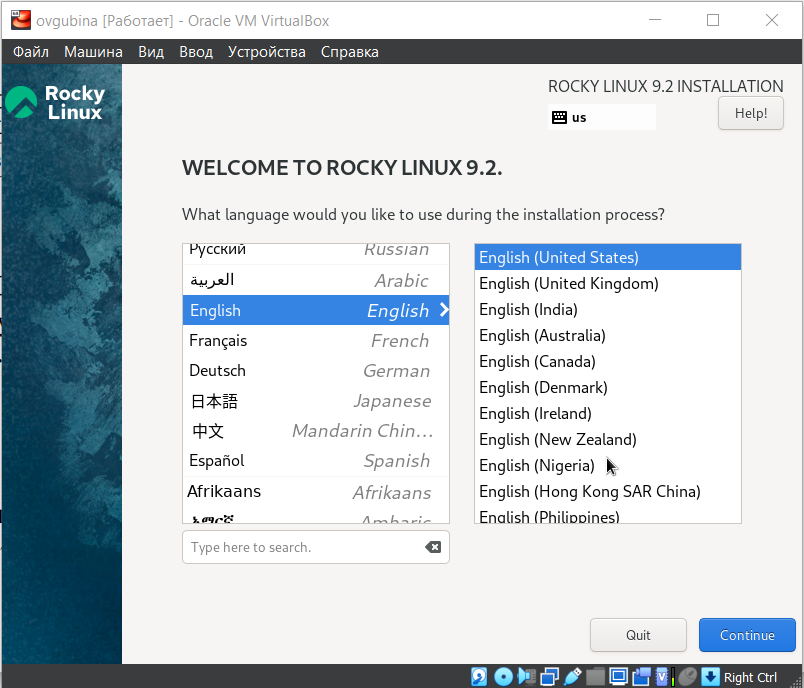
ВМ ovgubina

1. Запускаем виртуальную машину (рис. [??]):

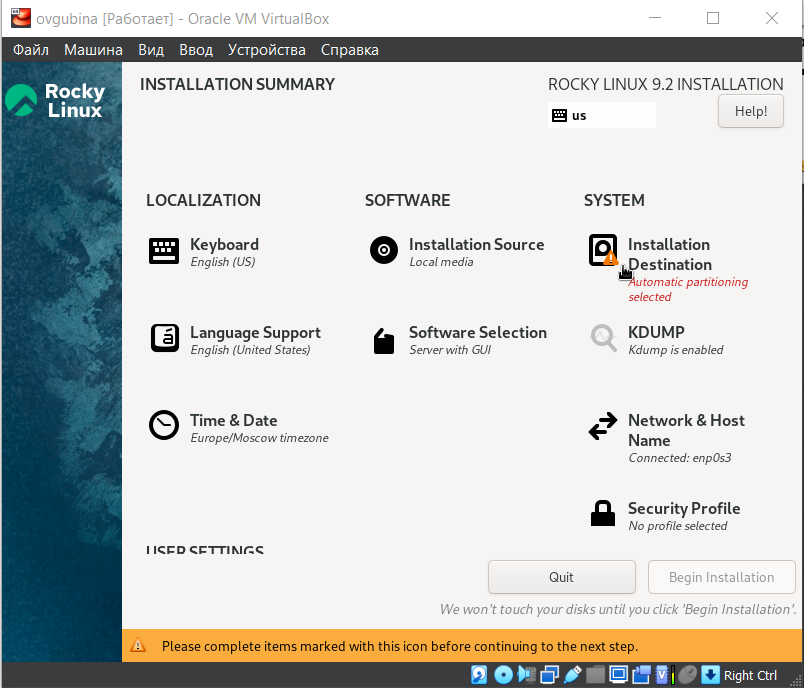


Запуск ВМ

1. Далее необходимо произвести настройку Rocky Linux и выбрать язык (рис. [??]-[??]).

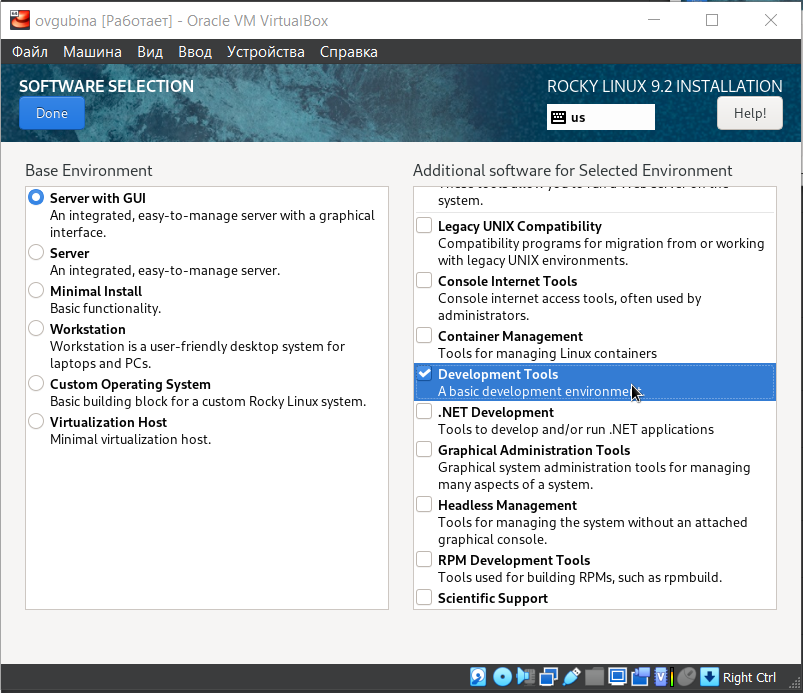


Выбор языка



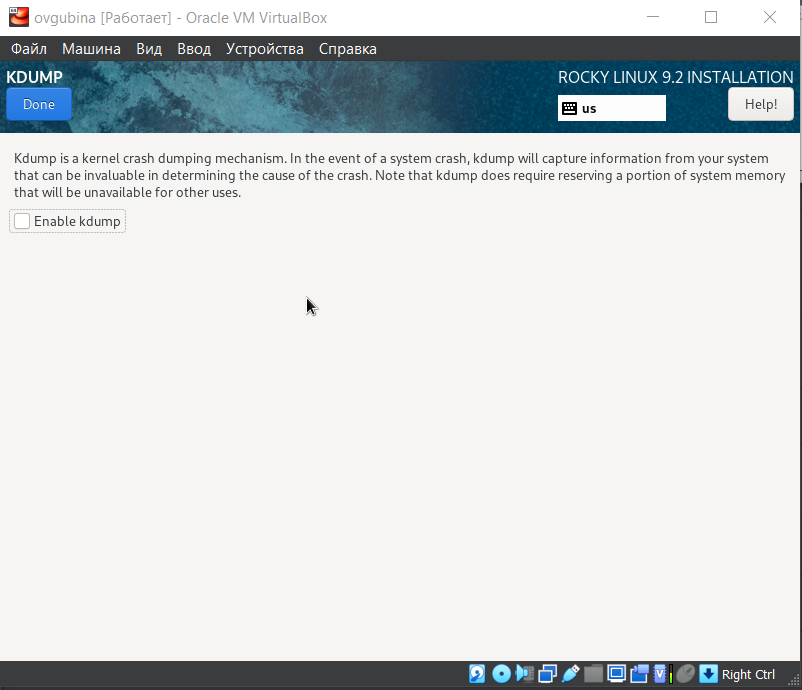
Параметры настройки установки образа ОС

В разделе выбора программ указываем в качестве базового окружения Server with GUI, а в качестве дополнения — Development Tools (рис. [??])



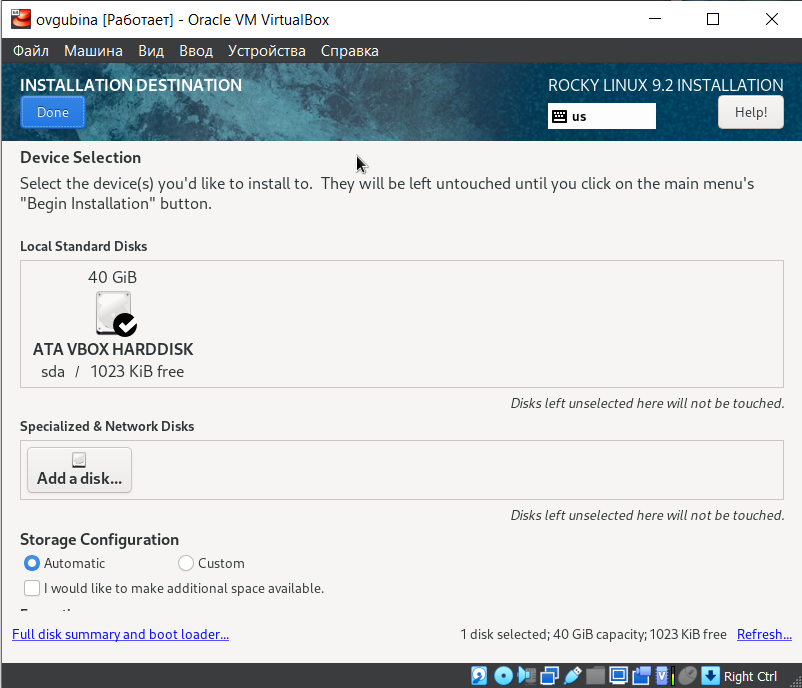
Выбор программ

Отключаем KDUMP (рис. [??]):



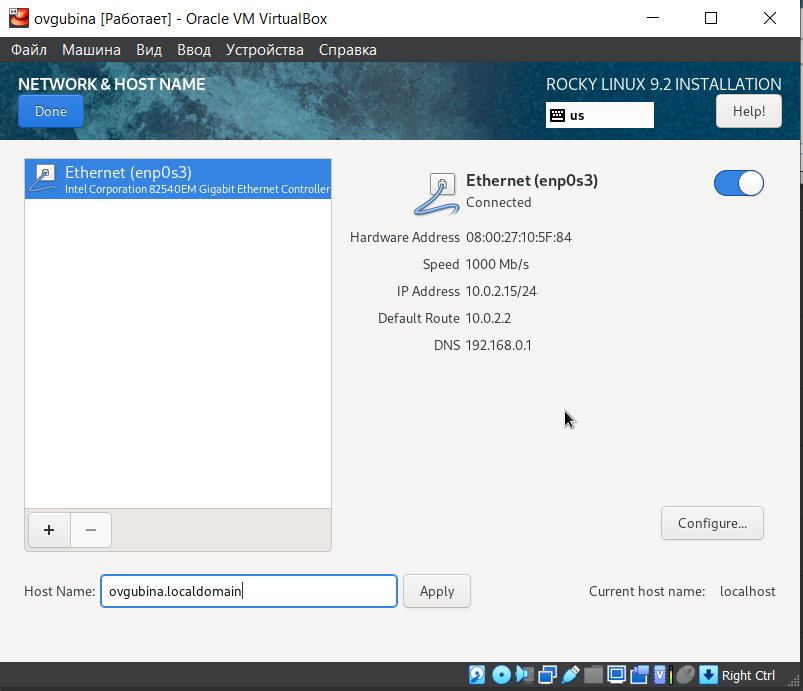
Отключение KDUMP

Место установки ОС оставляем без изменений (рис. [??]):



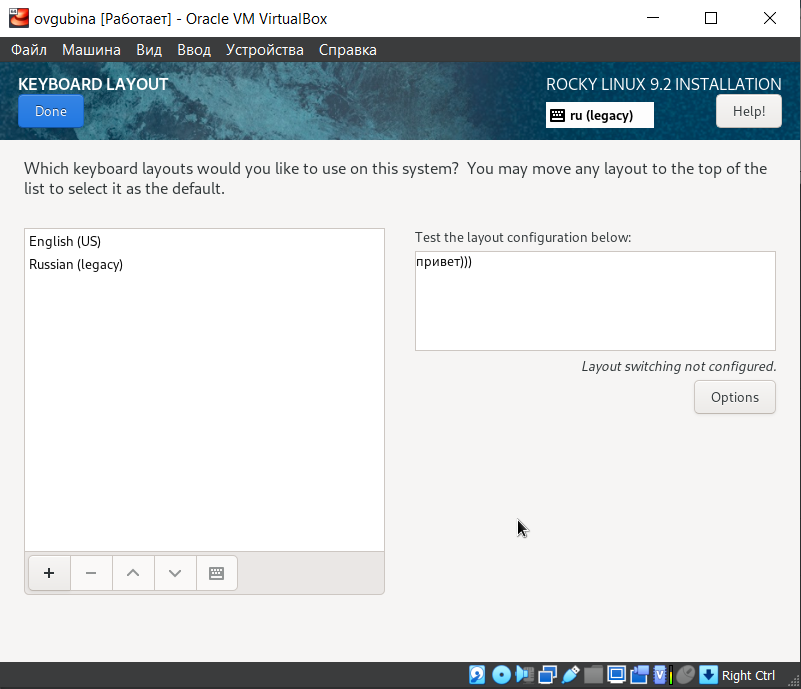
Место установки

Включим сетевое соединение и в качестве имени узла указываем ovgubina.localdomain (рис. [??]) в соответствии с соглашением об именовании.

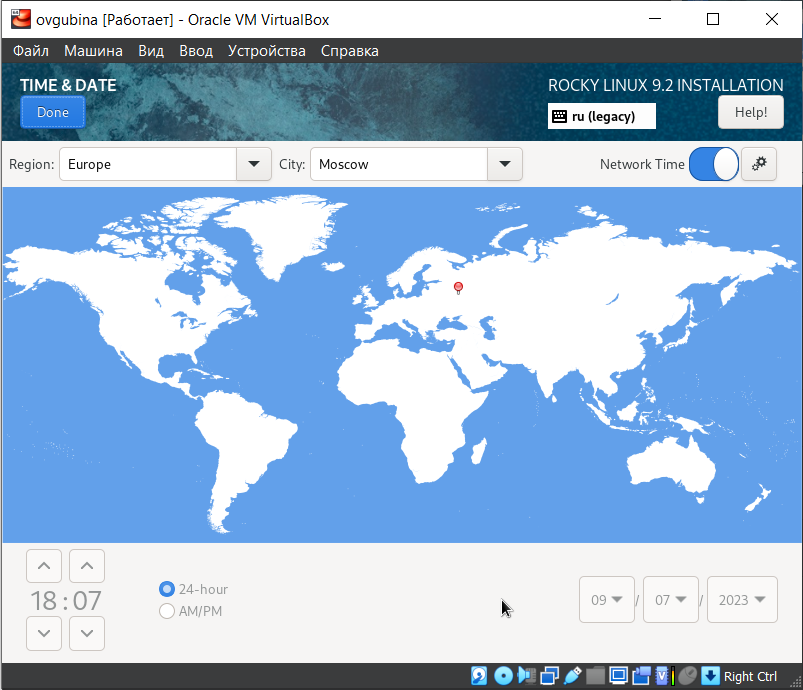


Сеть и имя узла

Проверяем дополнительные настройки (рис. [??]-[??]):

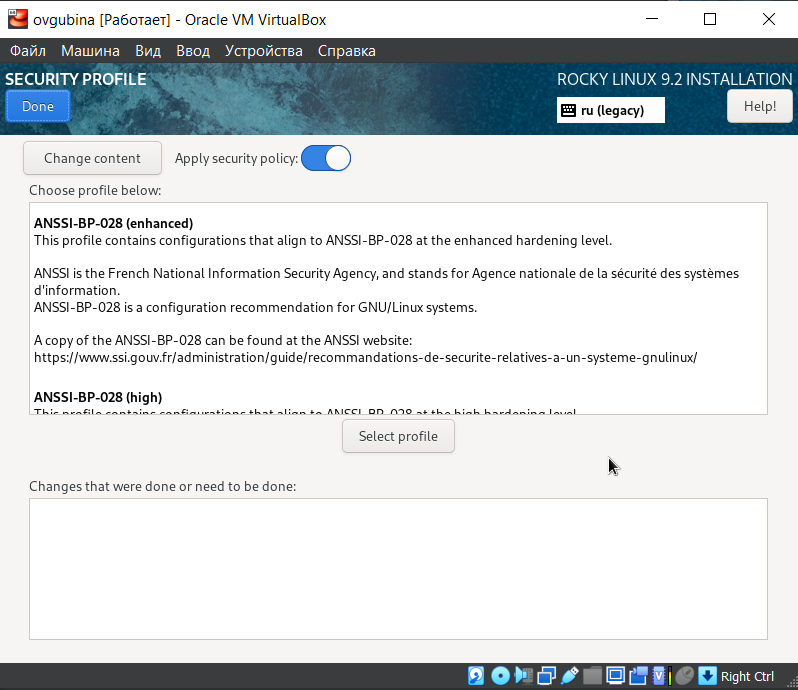


Выбор раскладки клавиатуры



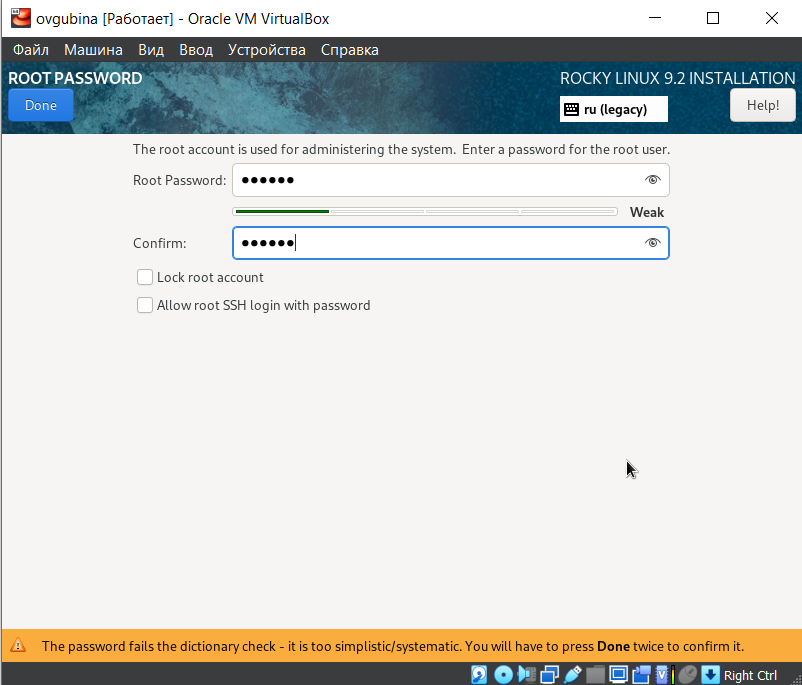
Выбор врмени и даты

Соглашаемся с политикой безопасновти (рис. [??]):



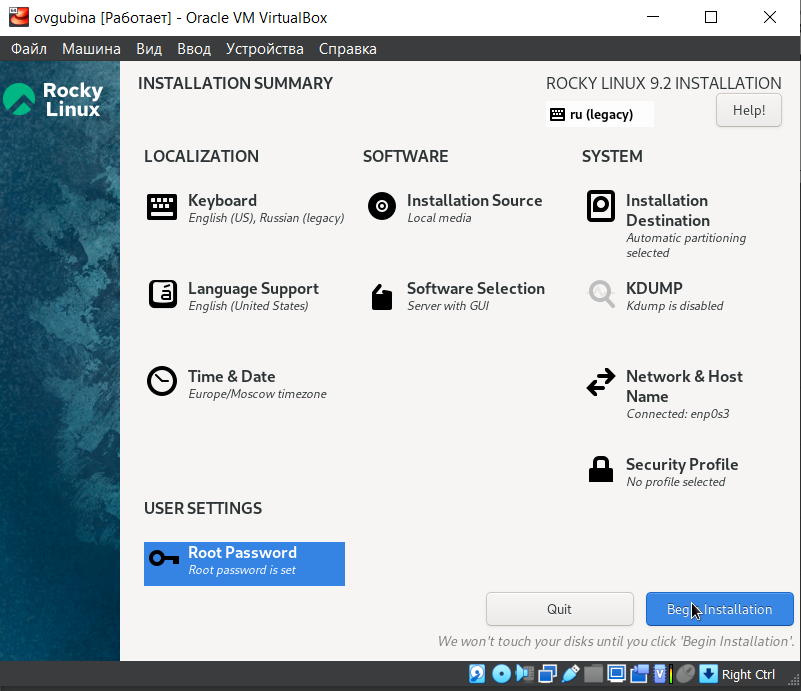
Политика безопасности

Устанавливаем пароль для пользователя root (рис. [??]):



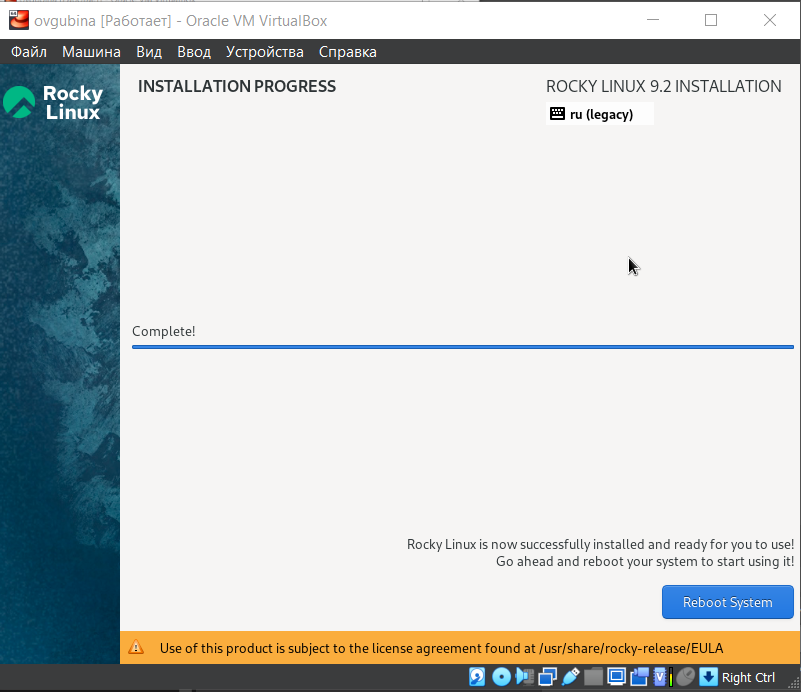
Установка пароля для root

После этого мы можем начать установку (рис. [??]):



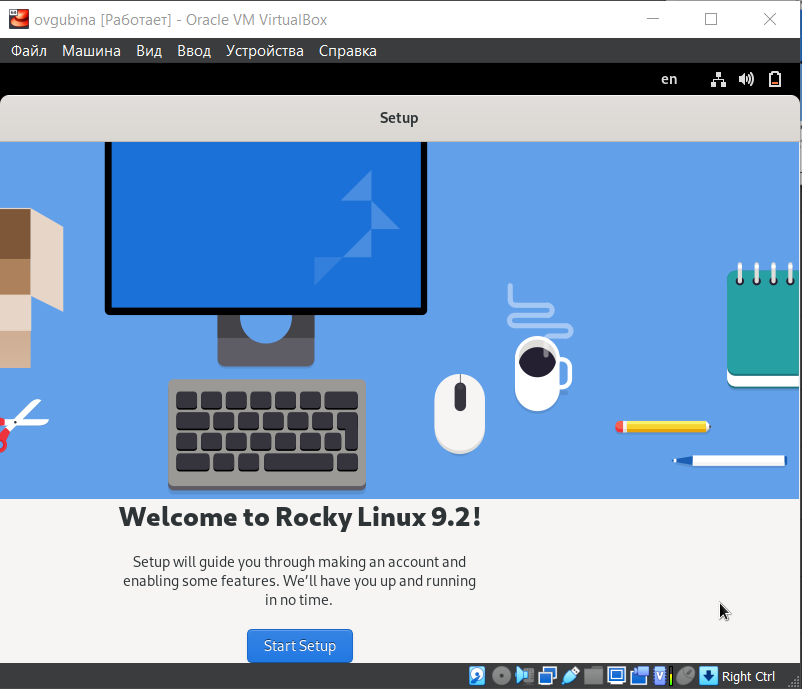
Завершение настроек, начало установки

Видим, что установка завершена, перезапускаем систему (рис. [??]):

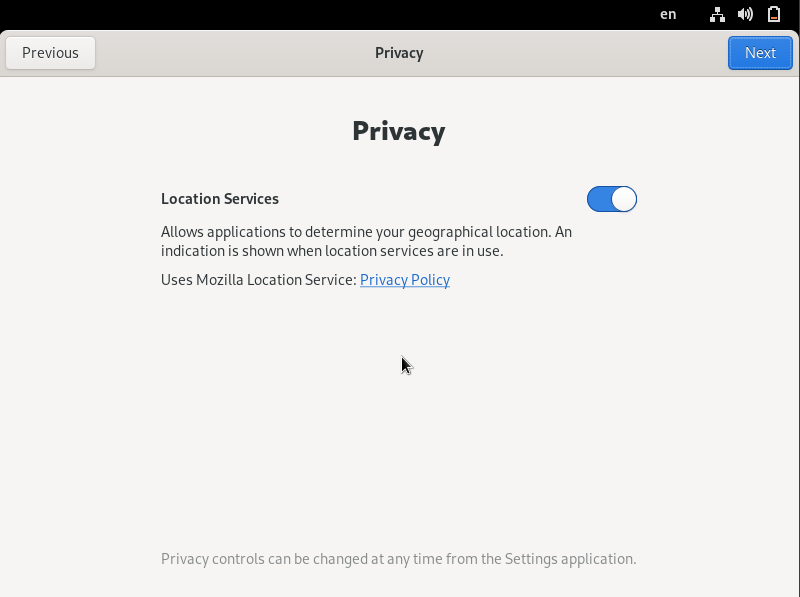


Успешное завершение установки образа

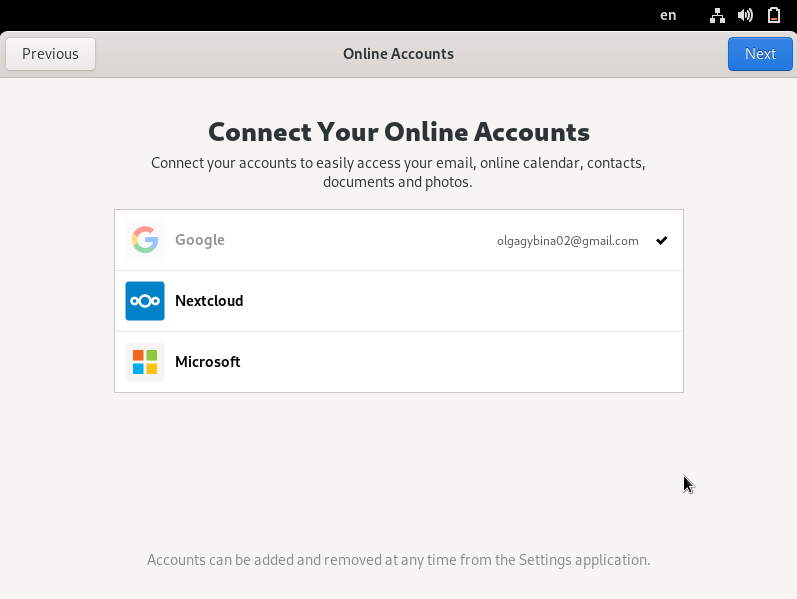
После перезапуска настраиваем ОС под пользователя (рис. [??]-[??]).



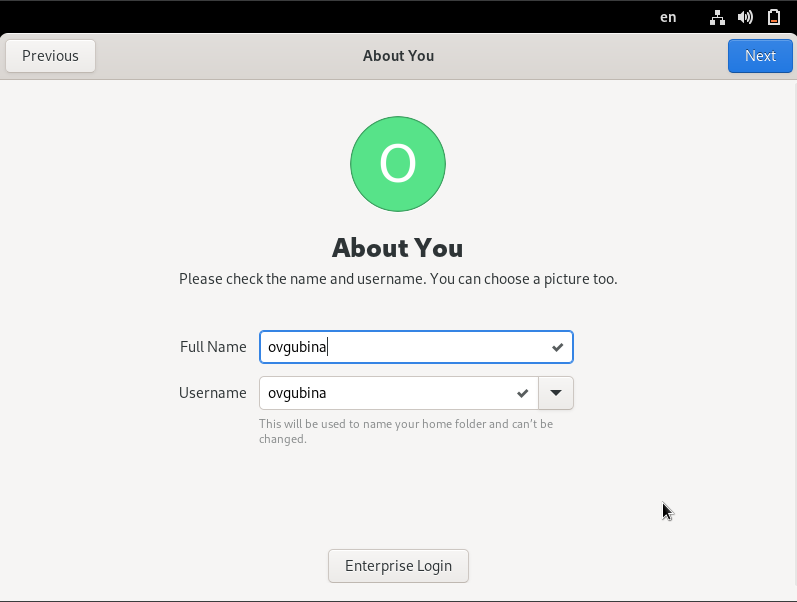
Начало работы в Rocky Linux



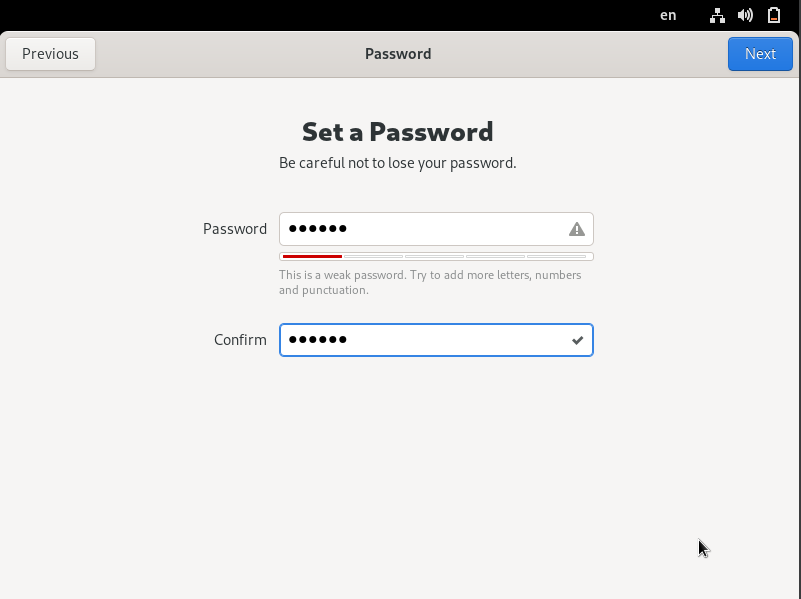
Доступ к геолокации



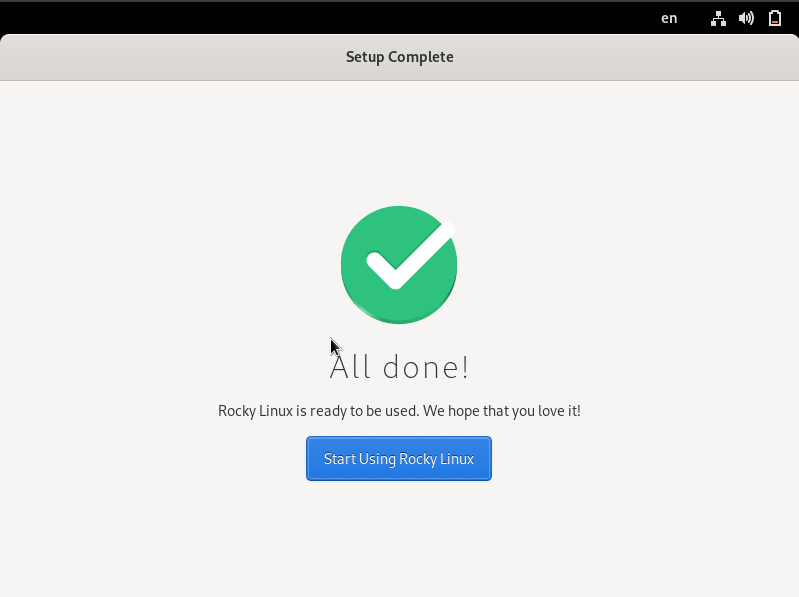
Подключение гугл-аккаунта



Имя пользователя

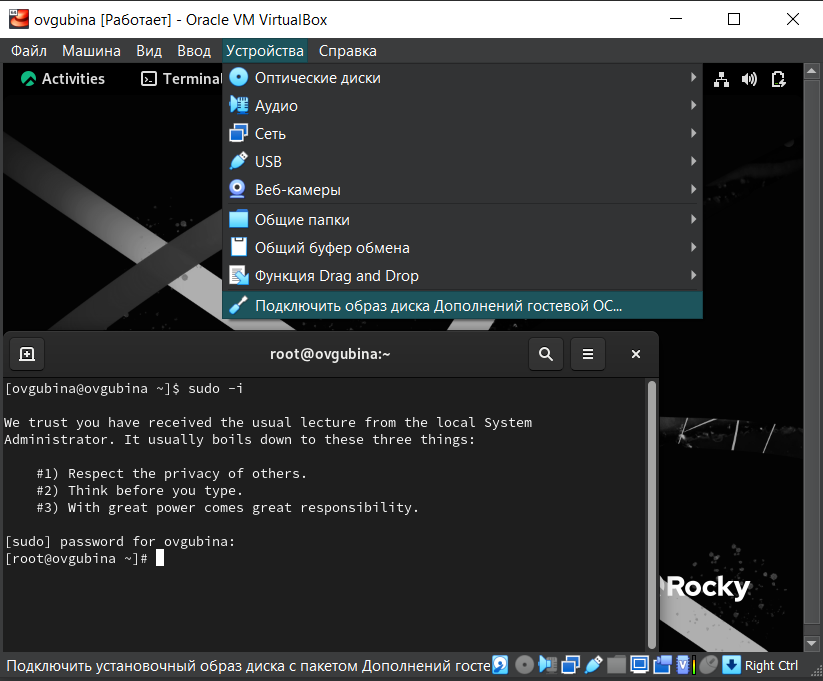


Установка пароля пользователя

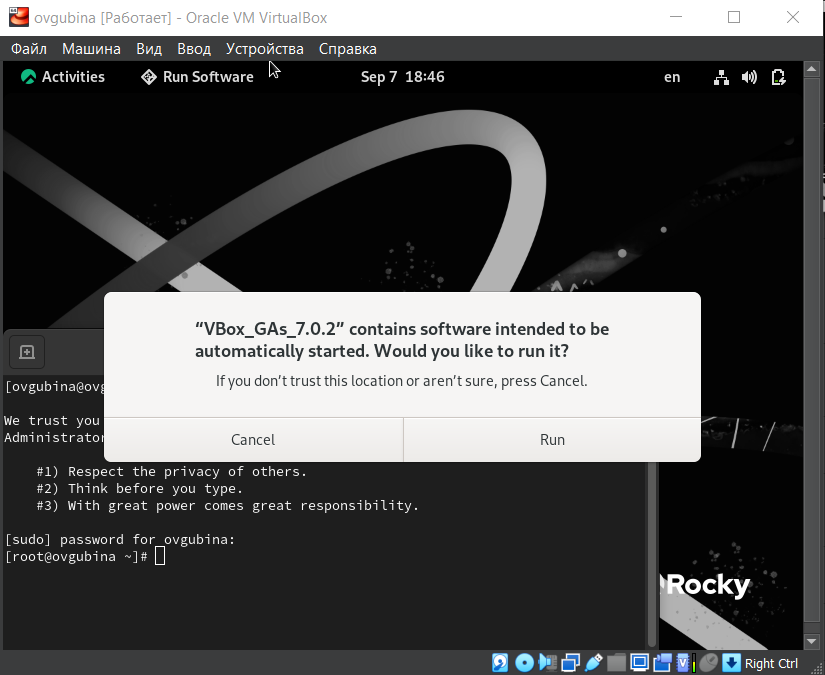


Окончание настройки ОС

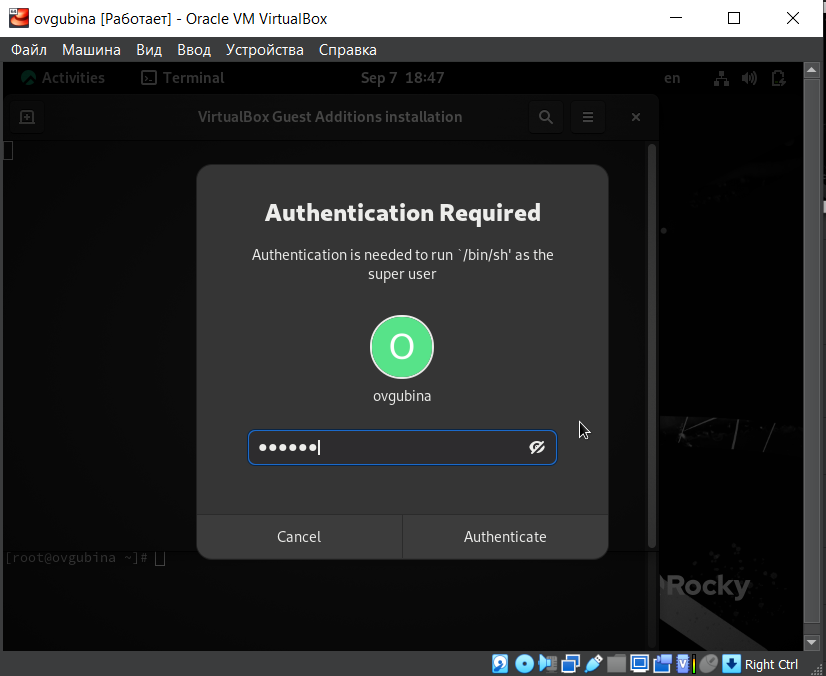
В меню “Устройства виртуальной машины” подключим образ диска дополнений гостевой ОС (рис. [??]-[??]), вводим пароль пользователя root виртуальной ОС (рис. [??]).



Подключение образа диска дополнений гостевой ОС



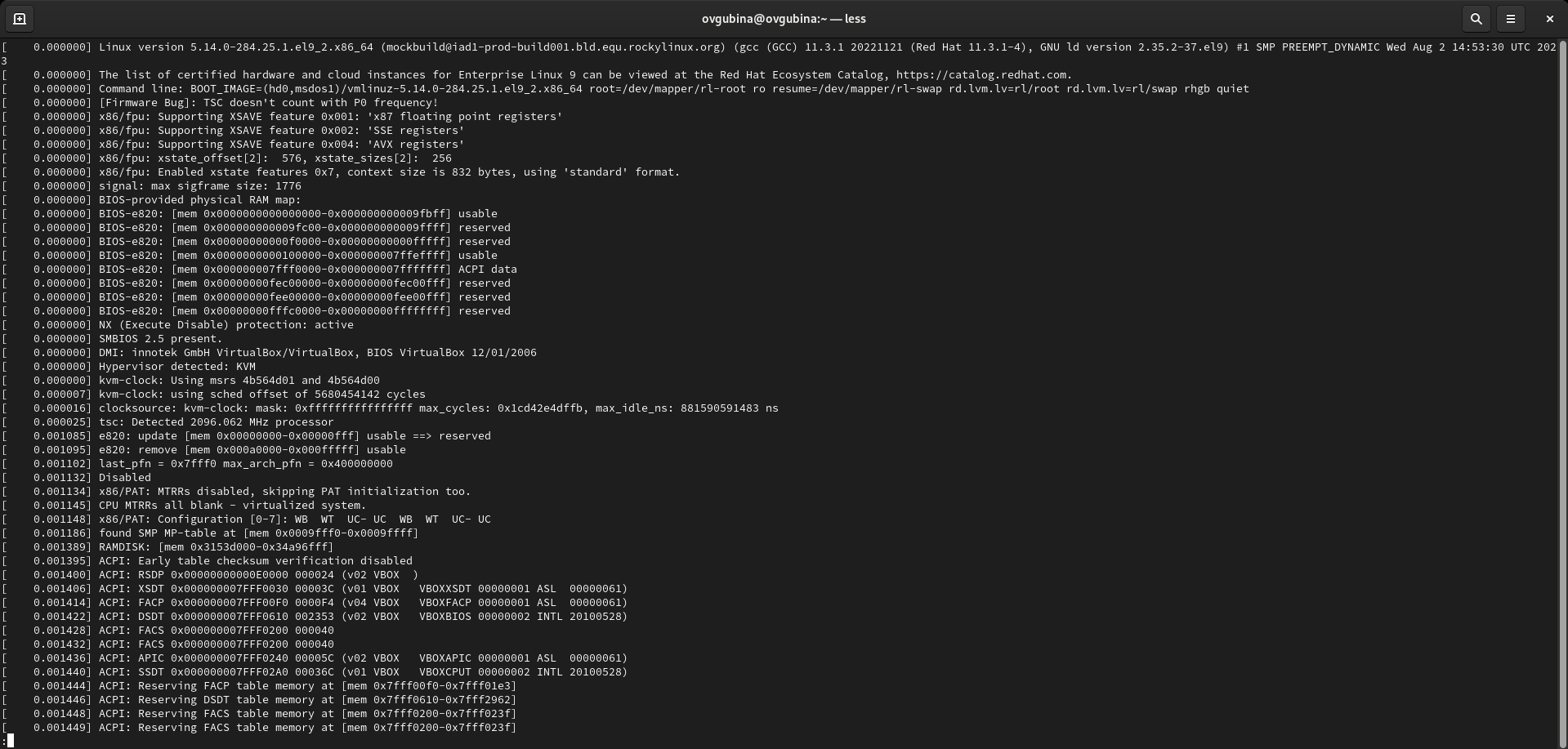
Запуск образа диска дополнений гостевой ОС



Подтверждение запуска

1. После перезапуска начинаем работу с домашним заданием.

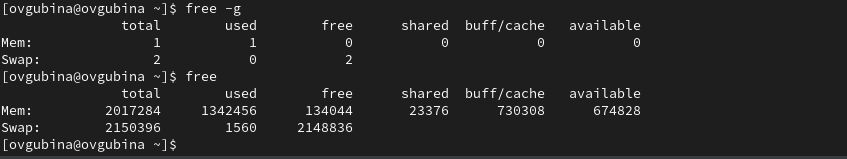
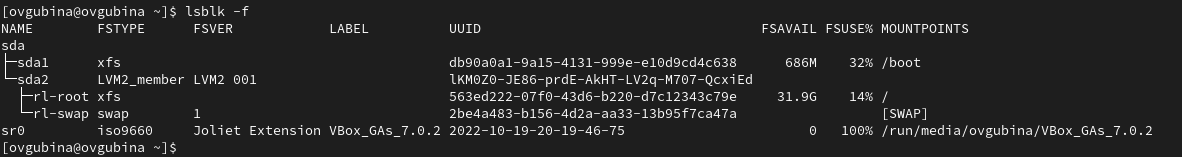
Дождидаемся загрузки графического окружения и открываем терминал. В окне терминала просмотрим последовательность загрузки системы, выполнив команду dmesg | less (рис. [??]).



Последовательность загрузки системы

Далее от нас требуется узнать: - Версия ядра Linux (Linux version) (рис. [??]).

```uname -r```  
  
![Информация о системе](image/34.png){#fig:38 width=70%}

* Частота процессора (Detected Mhz processor) (рис. [??]).
* cat /proc/cpuinfo | grep "cpu MHz - выводит частоту обоих процессоров - 2096.862MHz.
* Модель процессора (CPU0).
* lscpu - раздел Vendor ID (рис. [??]).
* Объем доступной оперативной памяти (Memory available) (рис. [??]).
* free
* 
* Анализ памяти
* Тип обнаруженного гипервизора (Hypervisor detected).
* lscpu - раздел Virtualization features (рис. [??]).
* Тип файловой системы корневого раздела.
* lsblk -f - тип xfs (рис. [??])
* 
* Информация о файловых системах
* Последовательность монтирования файловых систем (рис. [??]).
* mount
* 
* Последовательность монтирования ФС

# 5 Выводы

Создала репозиторий study\_2023-2024\_infosec дисциплины Информационная базопасность на git.com в соответствие с требованиями по шаблону и соблюдением соглашения наименований, соглашения для путей к файлам; Созздала ВМ на основе образа Rocky Linux 9.2 dvd; Написала отчет при использовании языка разметки markdown по выполненной работе.

# Список литературы

1. Git [Электронный ресурс]. 2023. URL: <https://ru.wikipedia.org/wiki/Git>.

2. Работа с git [Электронный ресурс]. 2023. URL: <https://esystem.rudn.ru/pluginfile.php/1971716/mod_folder/content/0/git.pdf>.

3. Markdown [Электронный ресурс]. 2022. URL: <https://ru.wikipedia.org/wiki/Markdown>.

4. Язык Markdown [Электронный ресурс]. 2023. URL: <https://esystem.rudn.ru/pluginfile.php/1971716/mod_folder/content/0/markdown.pdf>.

5. Rocky Linux [Электронный ресурс]. 2023. URL: <https://ru.wikipedia.org/wiki/Rocky_Linux>.