

# РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

---

Факультет физико-математических и естественных наук

## Отчёт по лабораторной работе №1

---

**Дисциплина:** Информационная безопасность

**Студент:** Оразгелдиева Огулнур

**Группа:** НПИбд-02-20

---

### Лабораторная работа №1

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

**Управление версиями**

---

#### Цель работы

Приобретение практических навыков установки операционной системы на виртуальную машину, настройки минимально необходимых для дальнейшей работы сервисов [1].

Изучить идеологию и применение средств контроля версий.

Освоить умения по работе с git.

---

#### Теоретические сведения

Система контроля версий Git представляет собой набор программ командной строки. Доступ к ним можно получить из терминала посредством ввода команды git с различными опциями.

Благодаря тому, что Git является распределённой системой контроля версий, резервную копию локального хранилища можно сделать простым копированием или архивацией [2].

---

#### Выполнение работы

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

1. Перед началом работы скачиваем и устанавливаем Oracle VirtualBox, а также скачиваем образ виртуальной машины Rocky 9.
2. Создайте новую виртуальную машину. Для этого в VirtualBox выберите "Машина" - "Создать" (см. рис. 1)

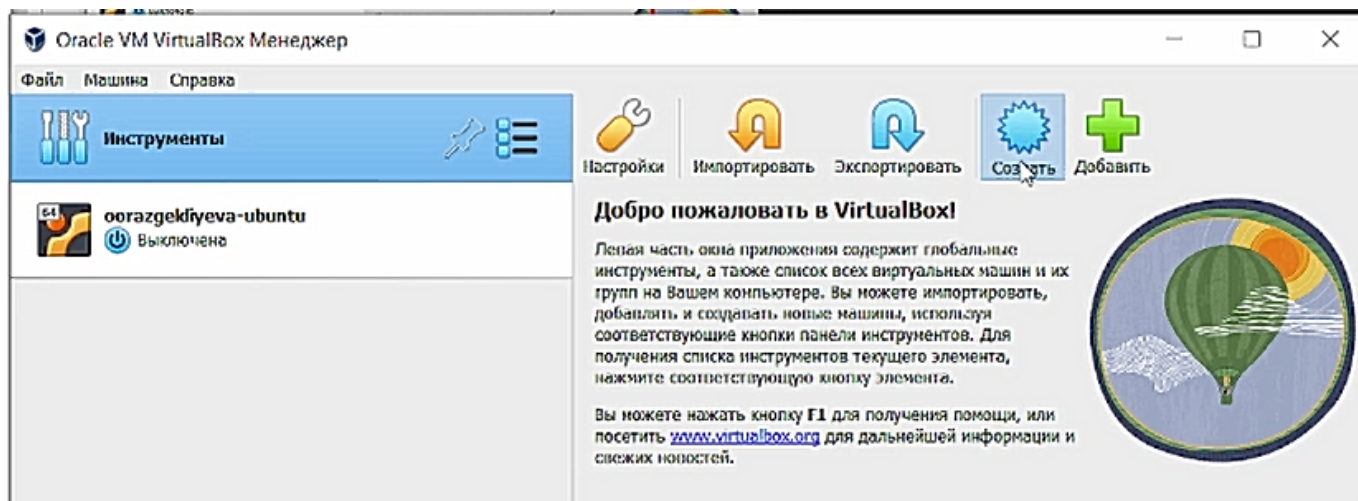


Рисунок 1. Создание виртуальной машины

3. Указываем имя виртуальной машины, тип операционной системы — Linux, RedHat (см. рис.2)

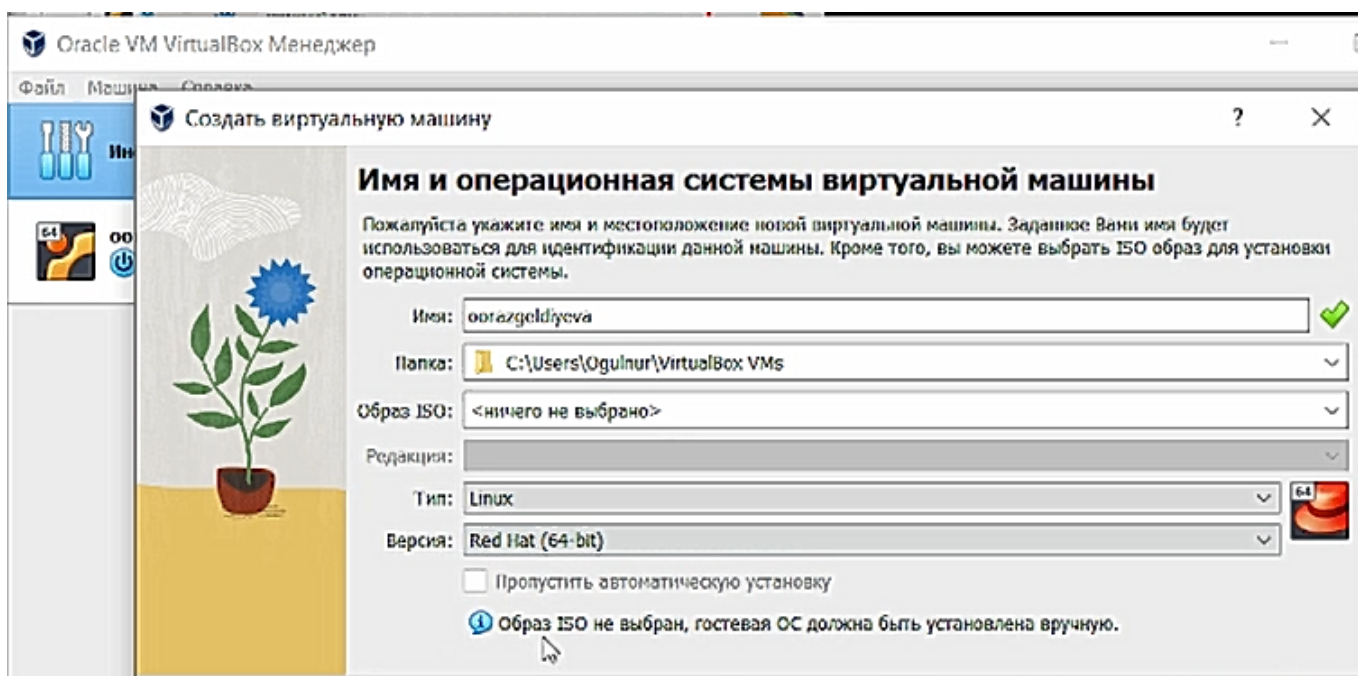


Рисунок 2. Имя и тип виртуальной машины

4. Указываем размер основной памяти виртуальной машины — 2048 МБ (см. рис. 3)

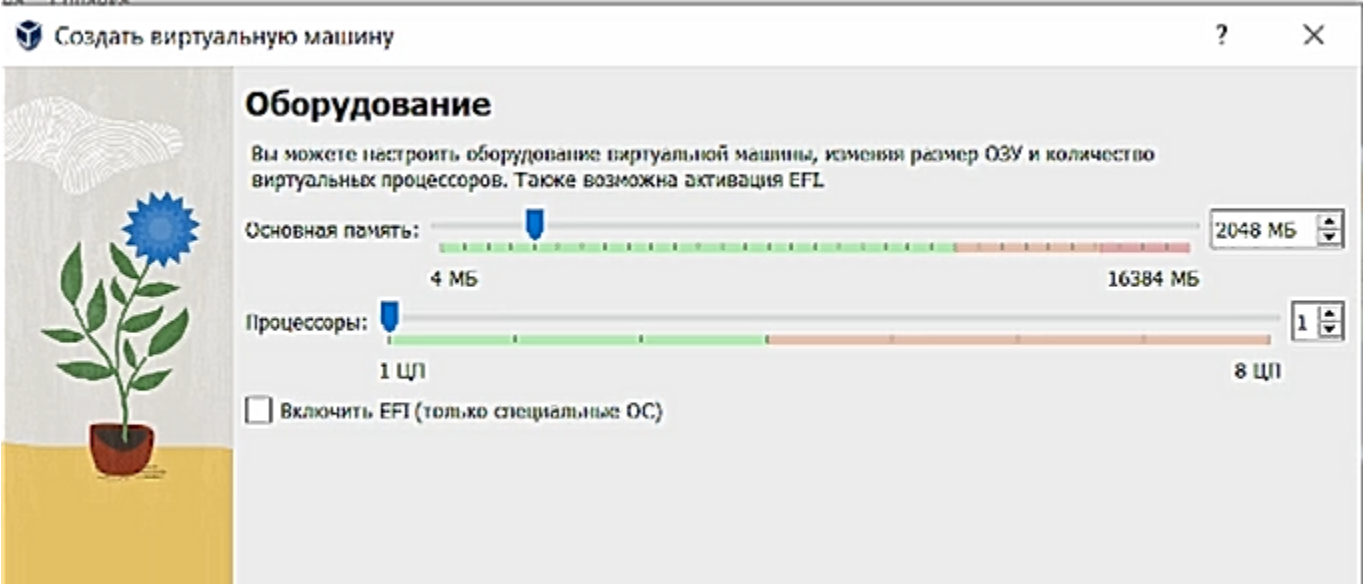


Рисунок 3. Размер основной памяти

5. Зададим размер диска — 40 ГБ (см. рис. 4)

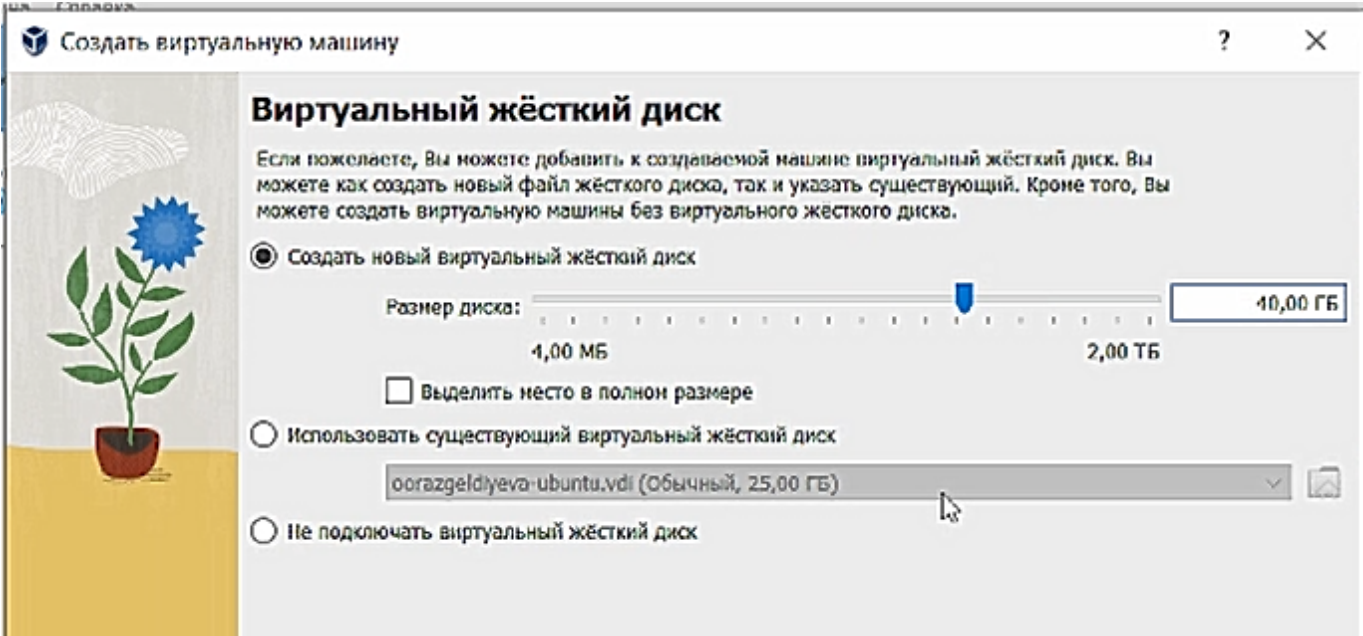


Рисунок 4. Размер диска

6. Подтверждаем заданные конфигурации (см. рис. 5)

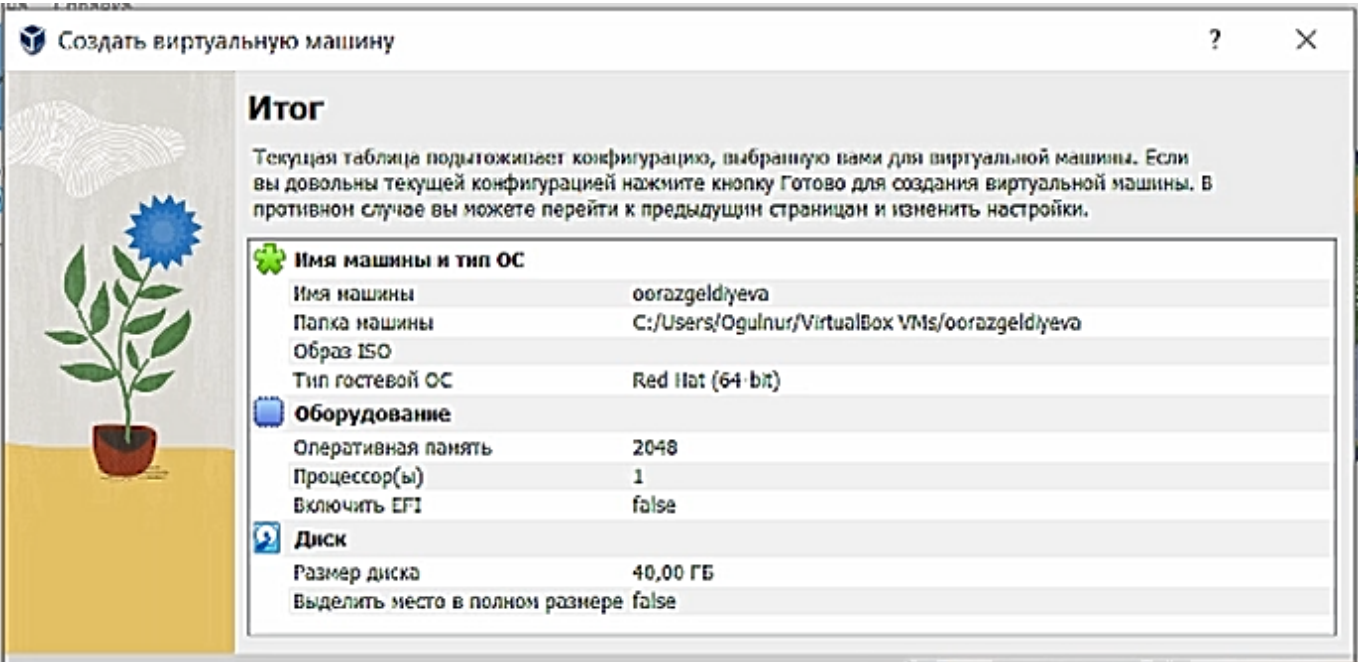


Рисунок 5. Создание виртуальной машины

7. После этого заходим в настройки созданной виртуальной машины и добавляем образ ОС (см. рис. 6)

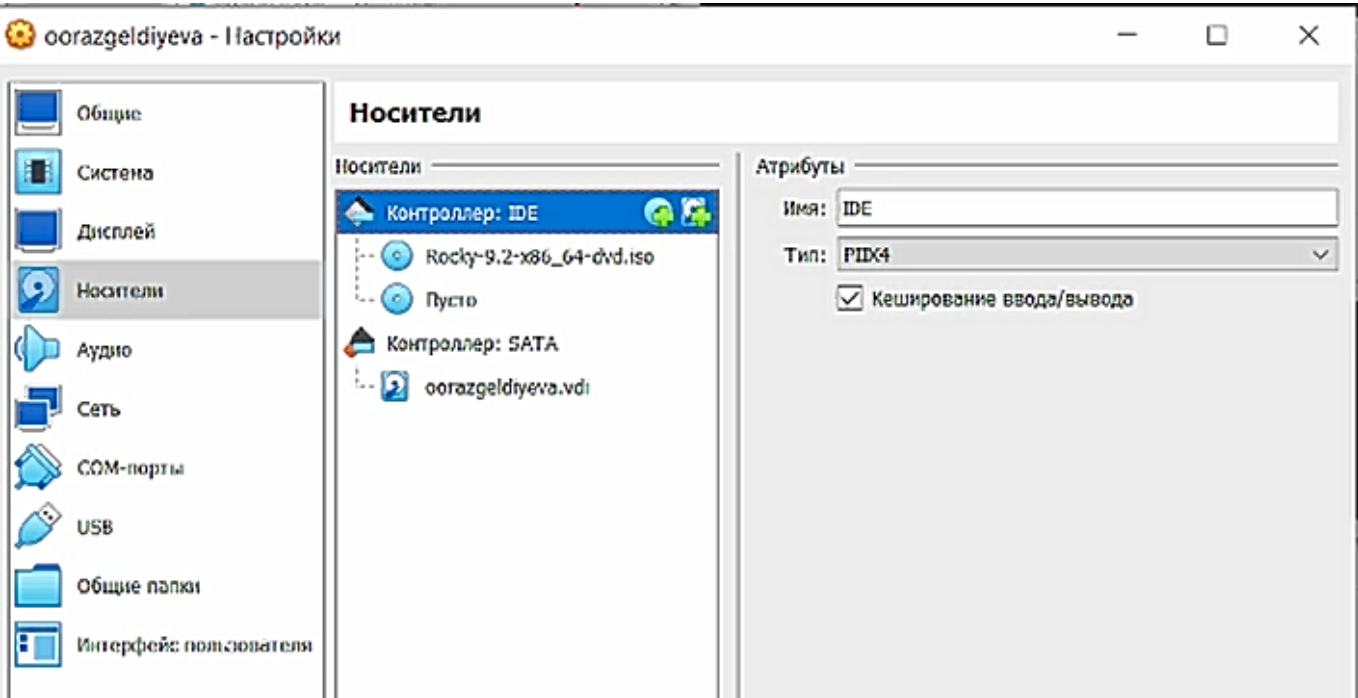


Рисунок 6. Образ ОС

Ждем загрузки (см. рис. 7)

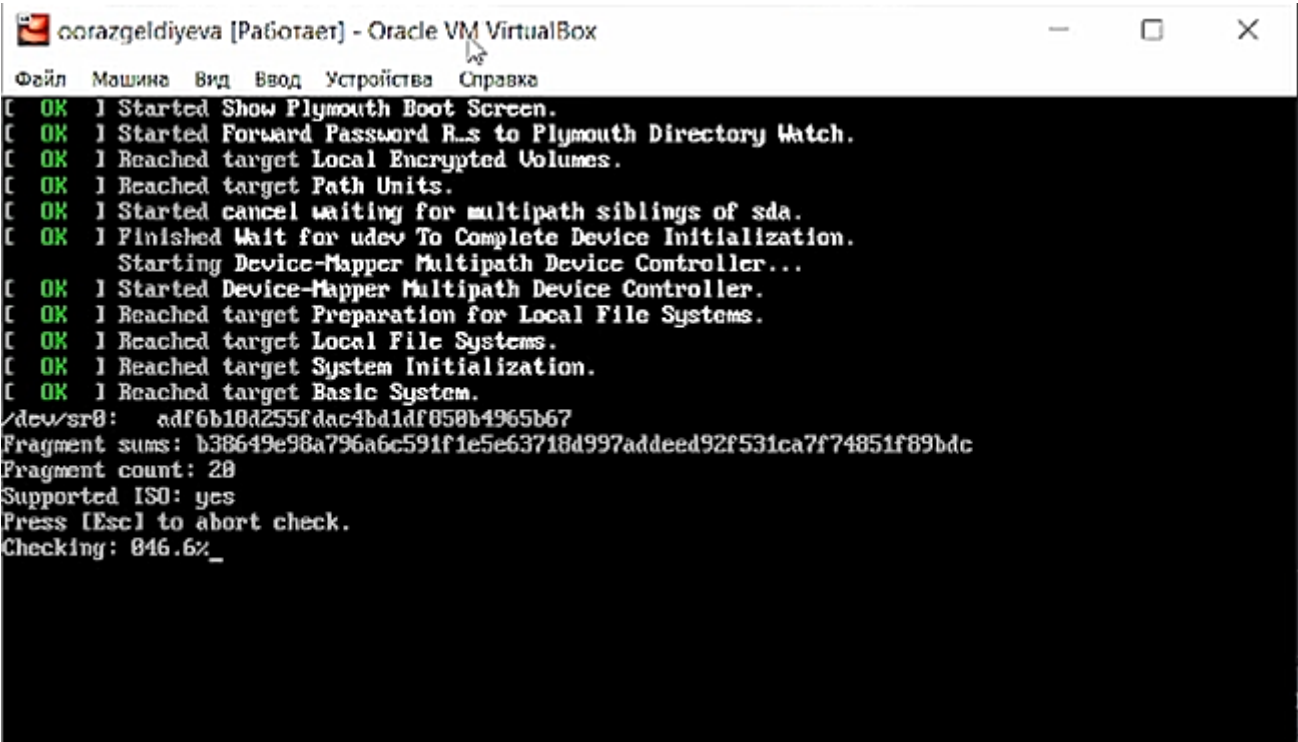


Рисунок 7. Загрузка виртуальной машины

8. Выбираем язык для процесса установки (см. рис. 8)

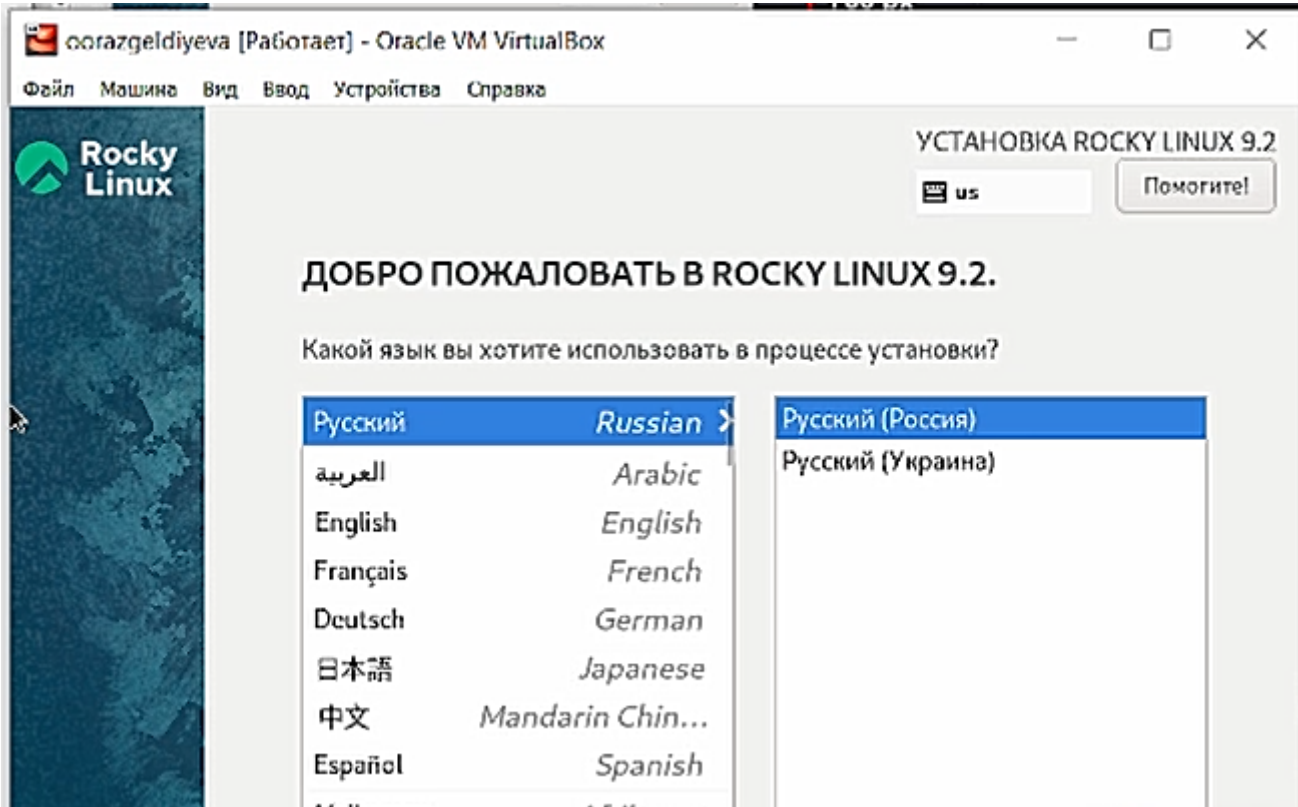


Рисунок 8. Выбор языка

9. Теперь начинаем установку. Выбираем место установки (см. рис. 9-10)



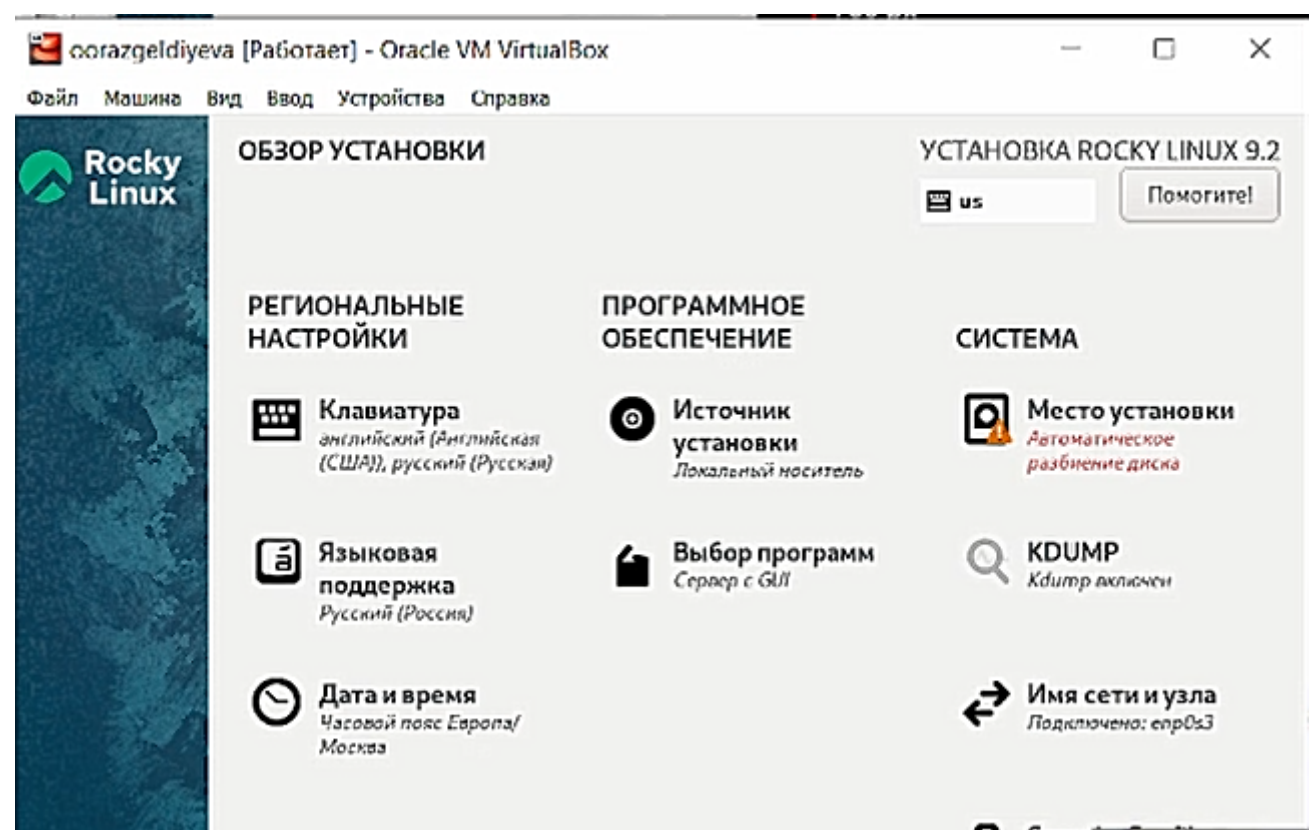


Рисунок 9. Установка виртуальной машины

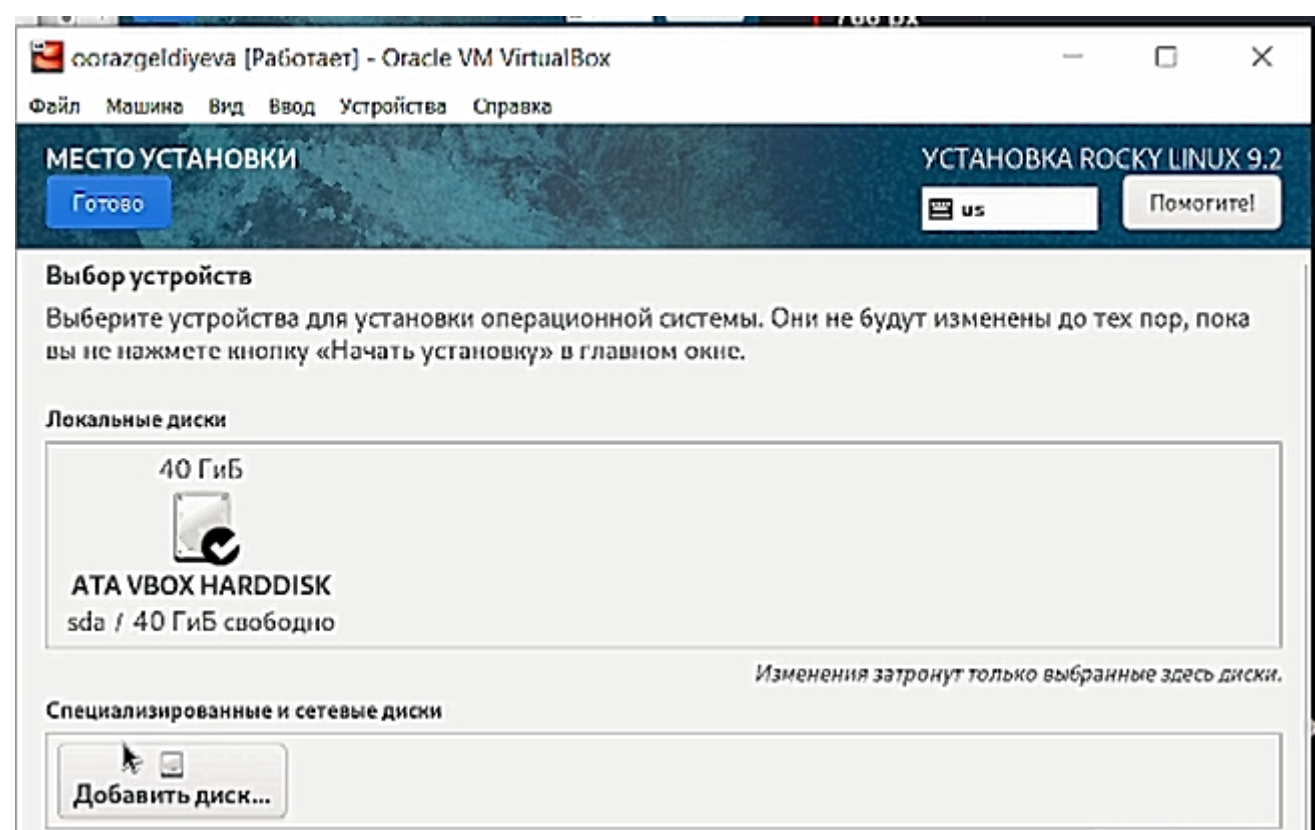


Рисунок 10. Выбор места установки

10. Далее выбираем программы установки (средства разработки) и отключаем kdump (см. рис. 11-12)

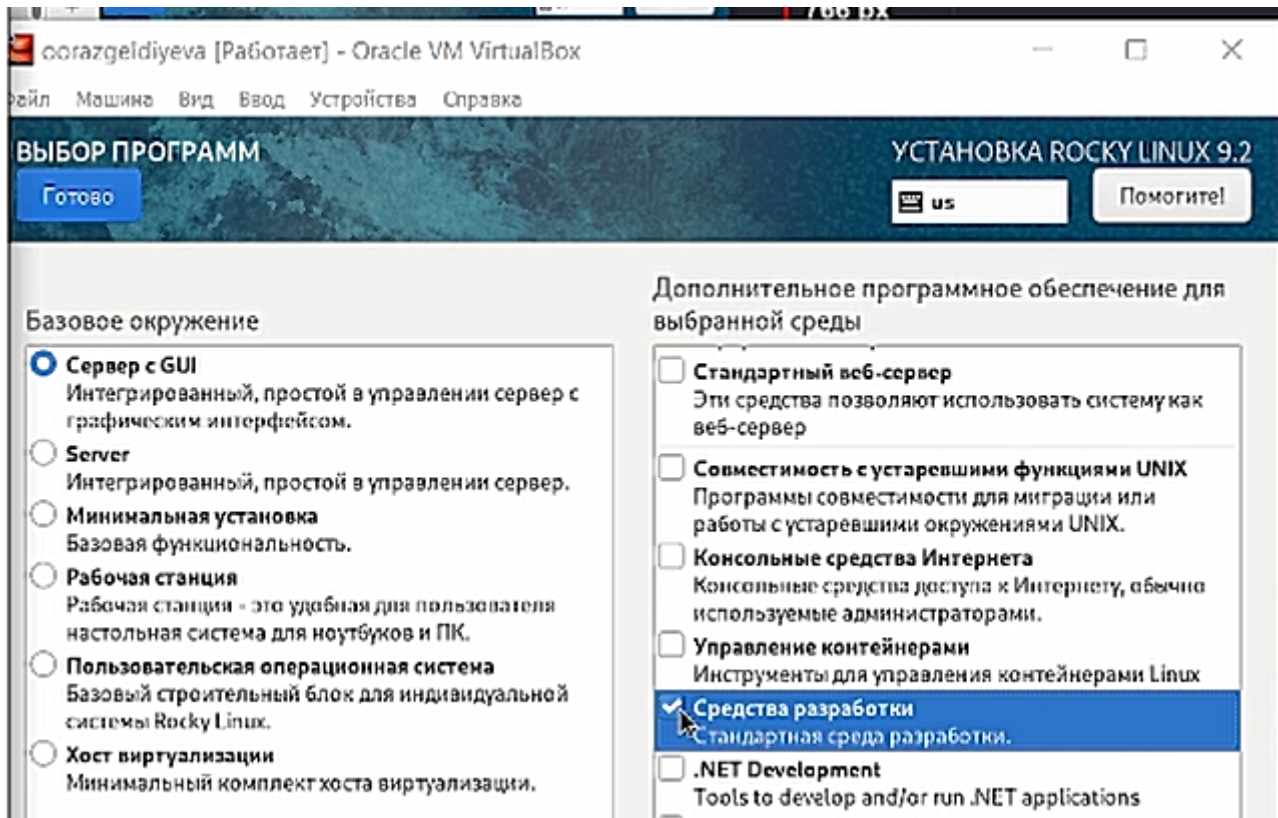


Рисунок 11. Выбор программ установки

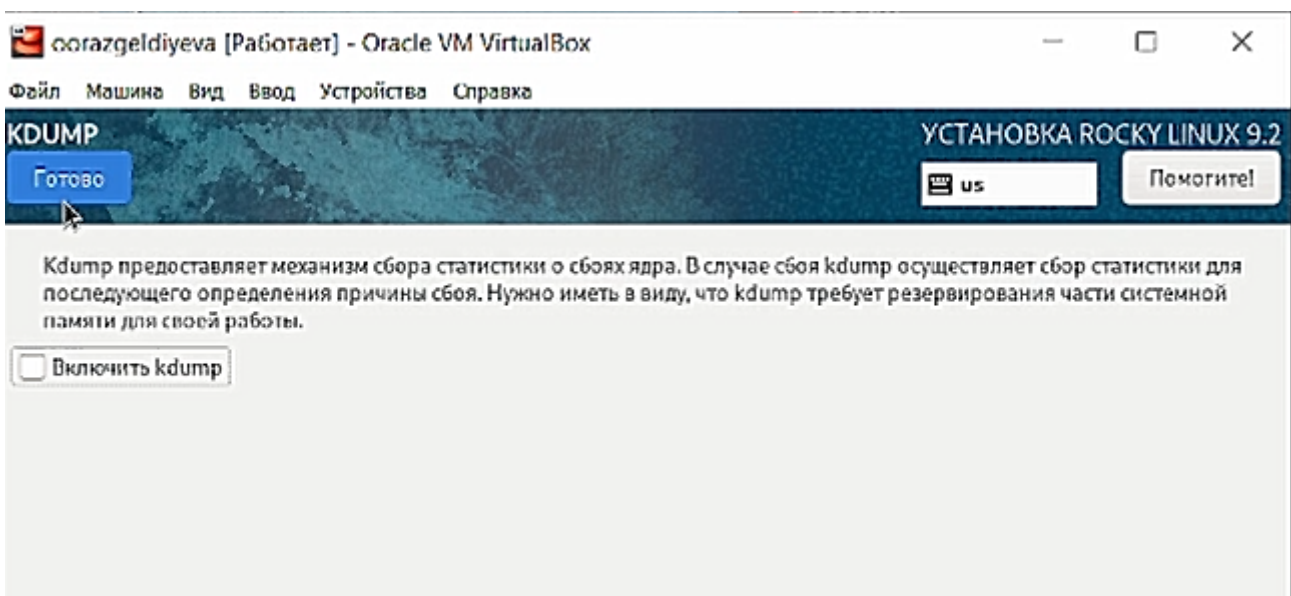


Рисунок 12. Отключение kdump

11. Задаем имя хоста, а также придумываем пароль для root(см. рис. 13-14)

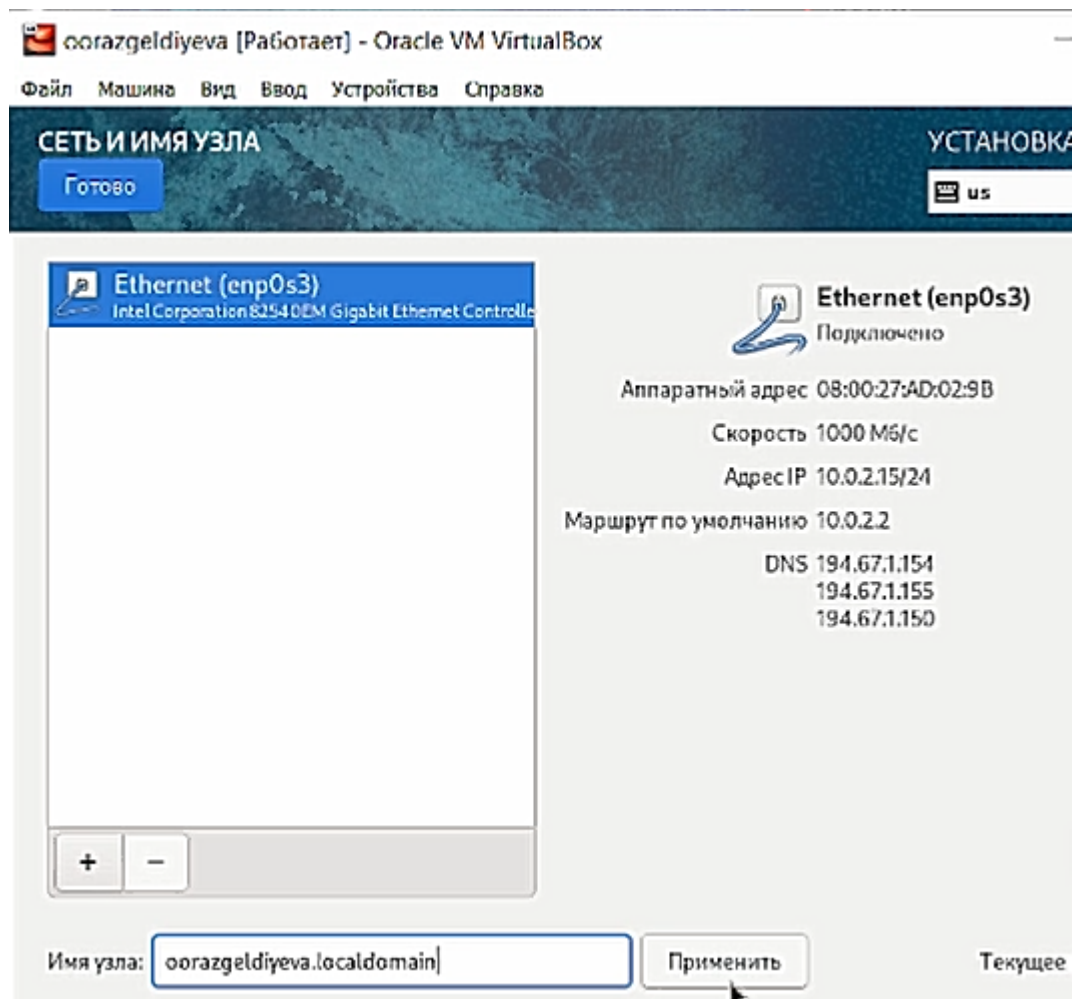


Рисунок 13. Имя хоста

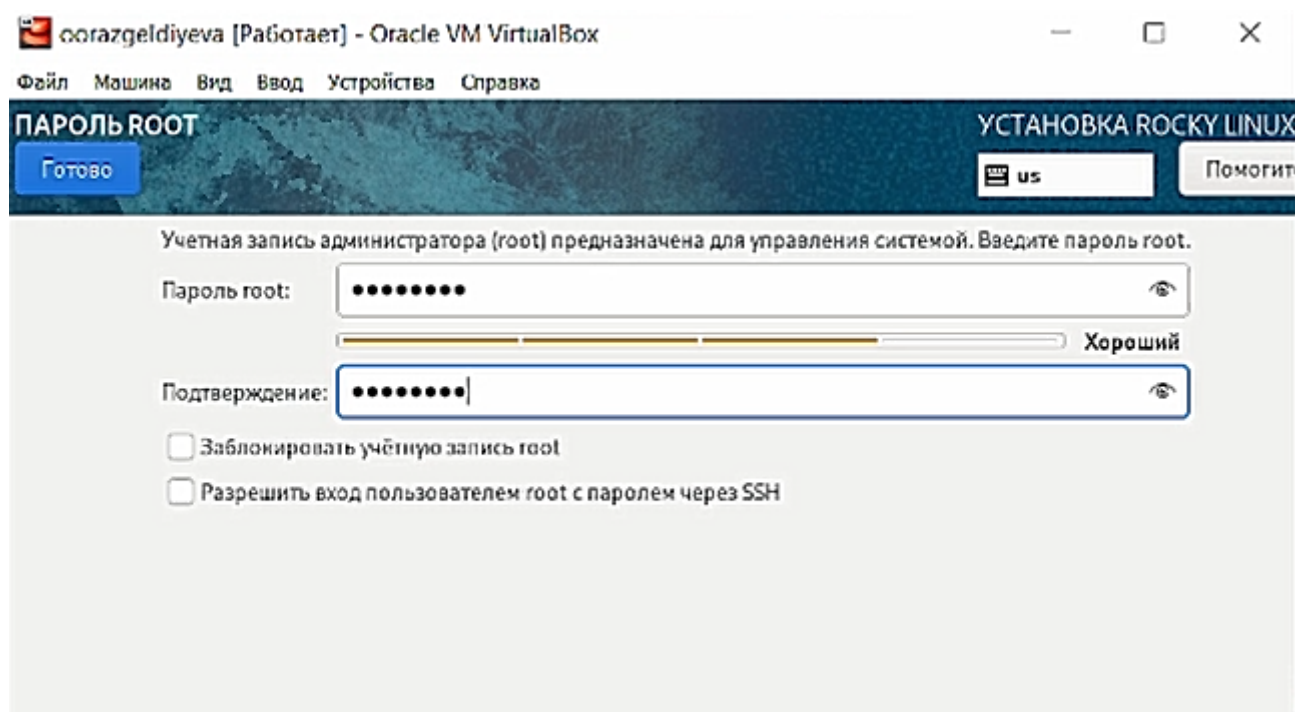


Рисунок 14. Пароль root

12. После установки создаем пользователя (см. рис. 15-17)



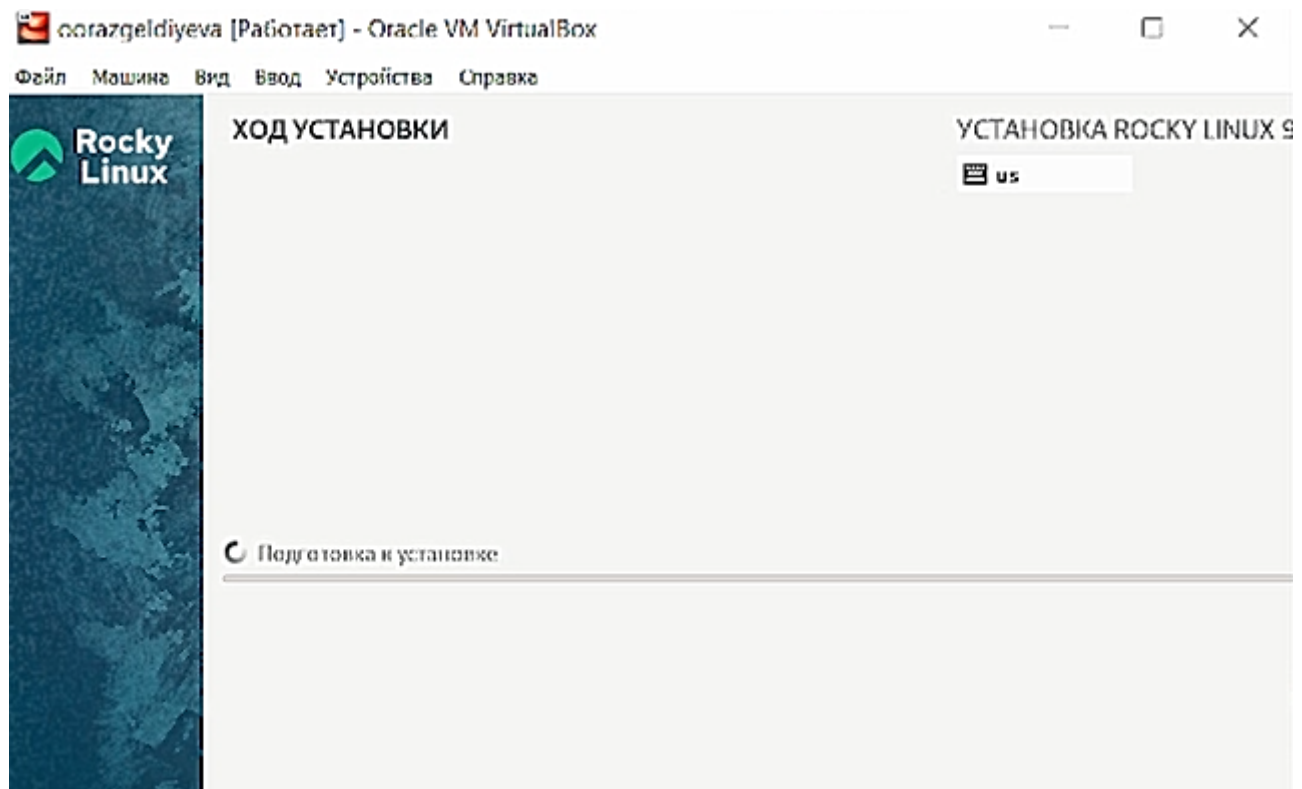


Рисунок 15. Установка

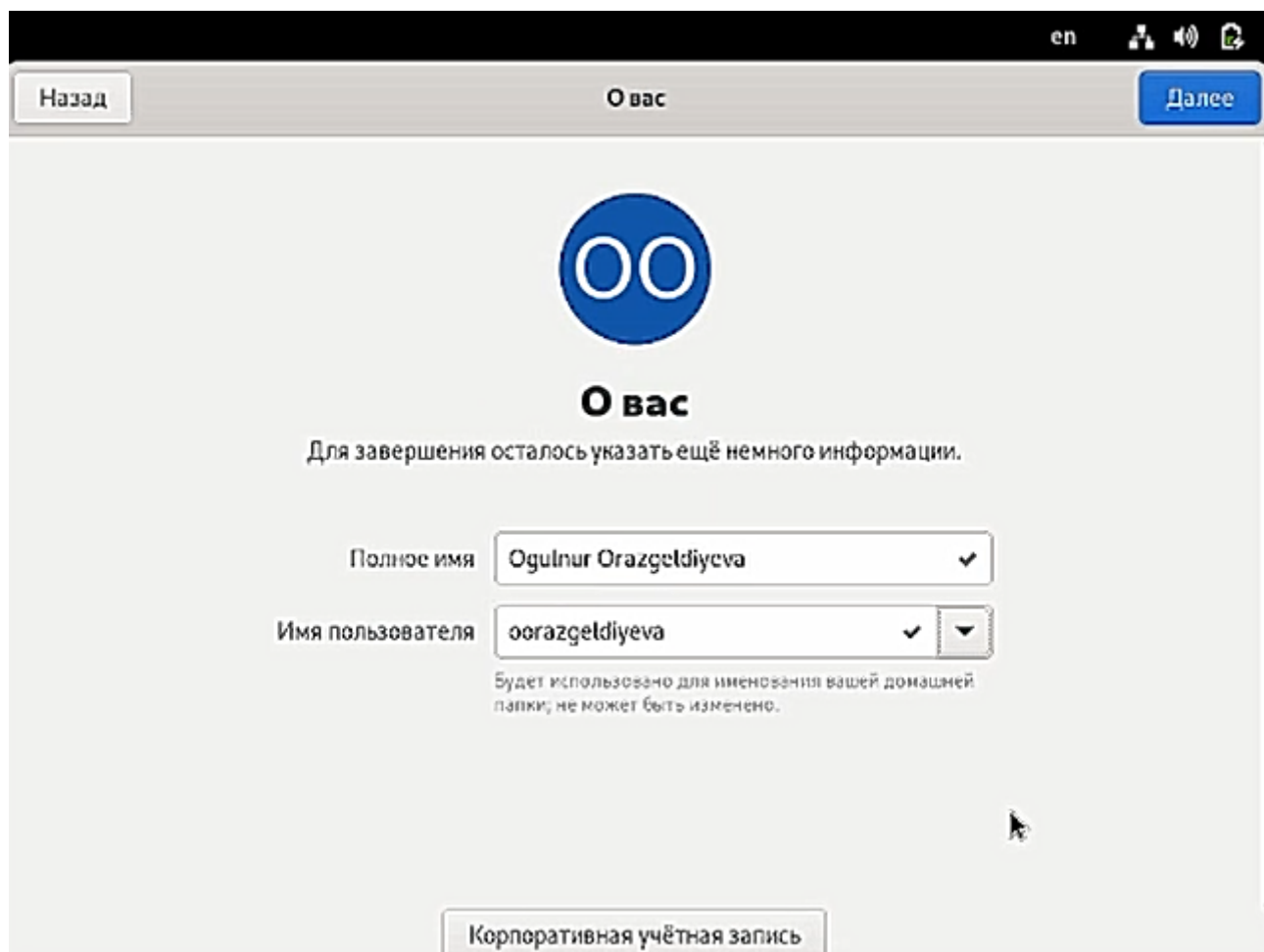


Рисунок 16. Создание учетной записи

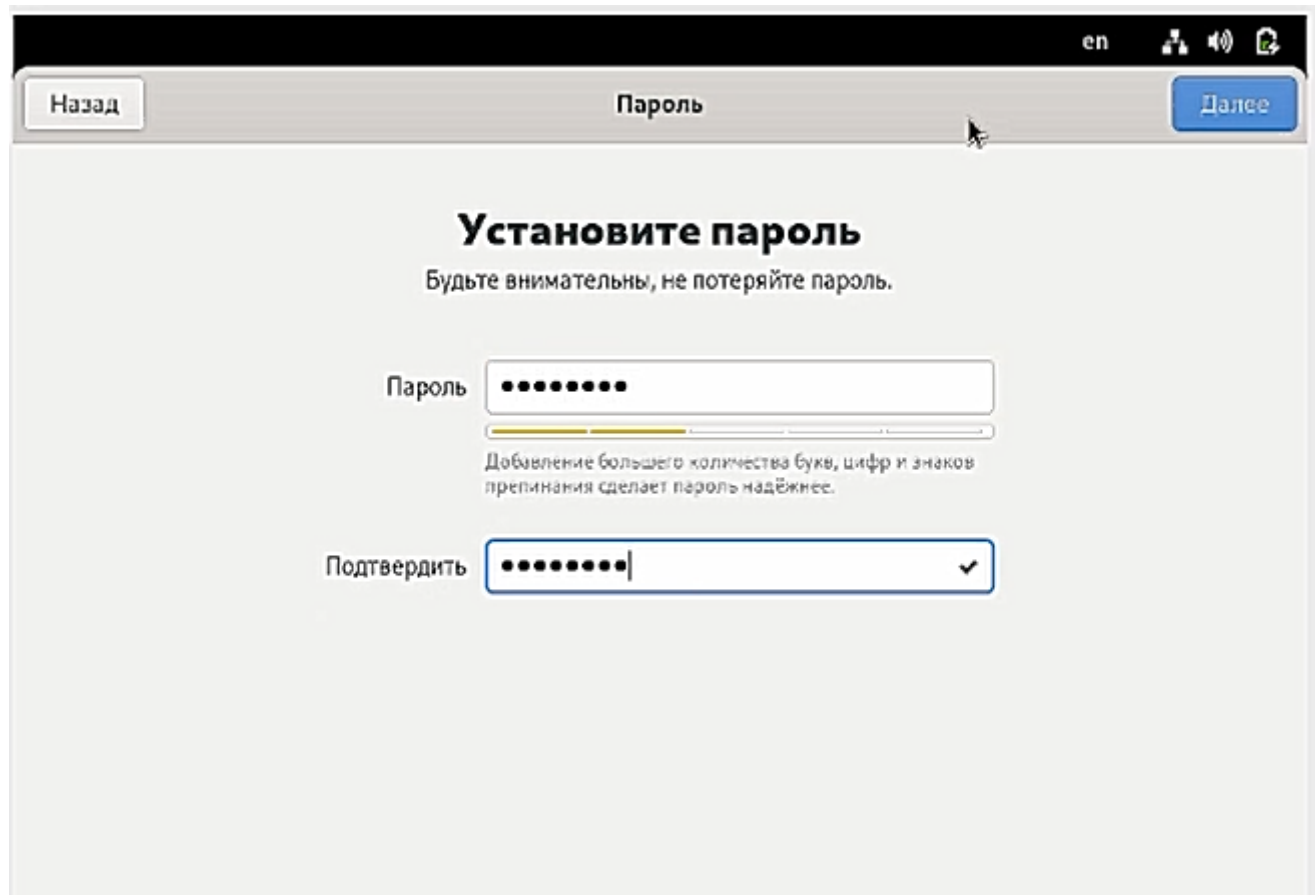


Рисунок 17. Создание учетной записи

13. После этого подключаем образ дополнений гостевой ОС (см. рис. 18-19)

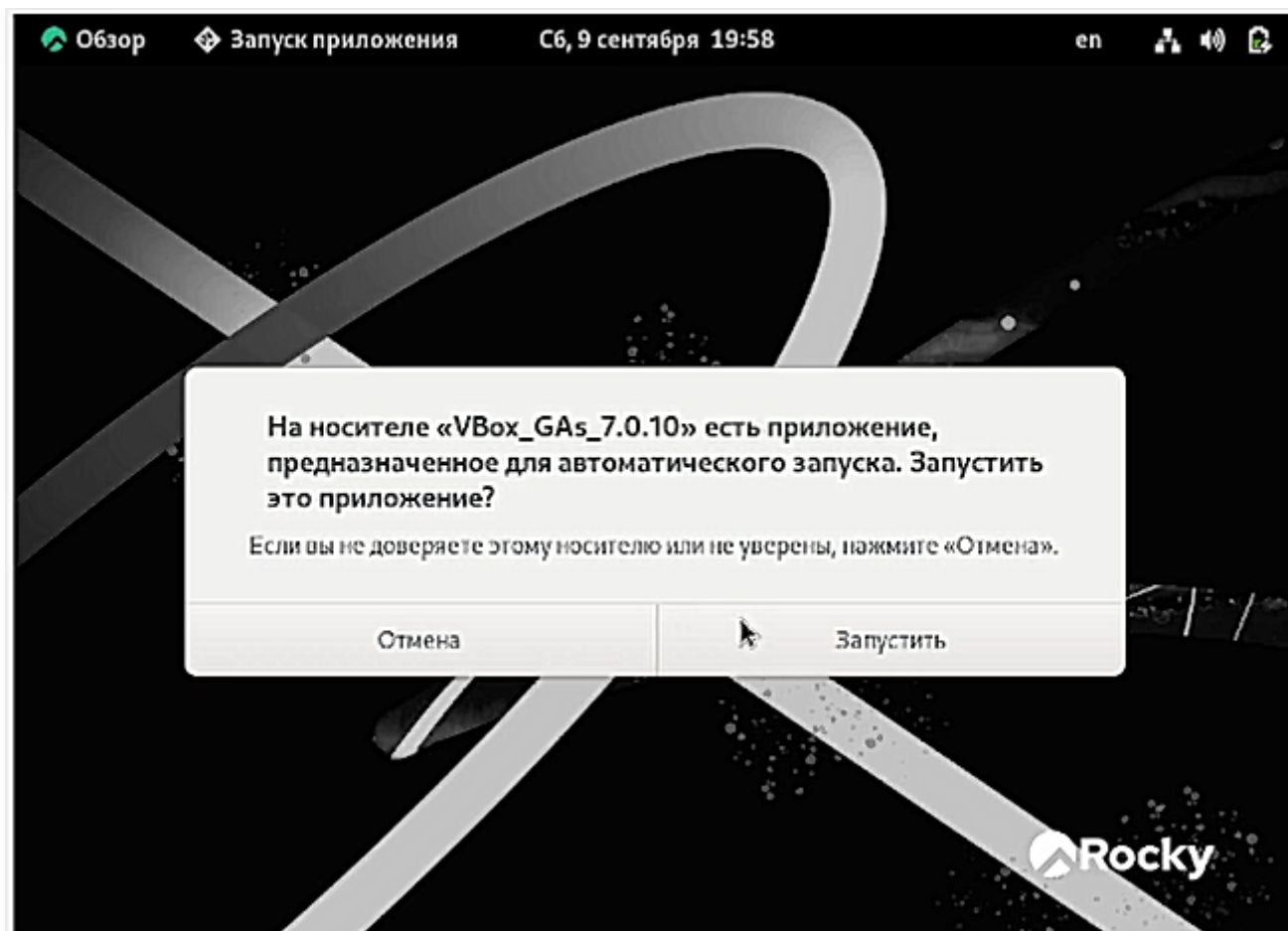


Рисунок 18. Образ дополнений гостевой ОС

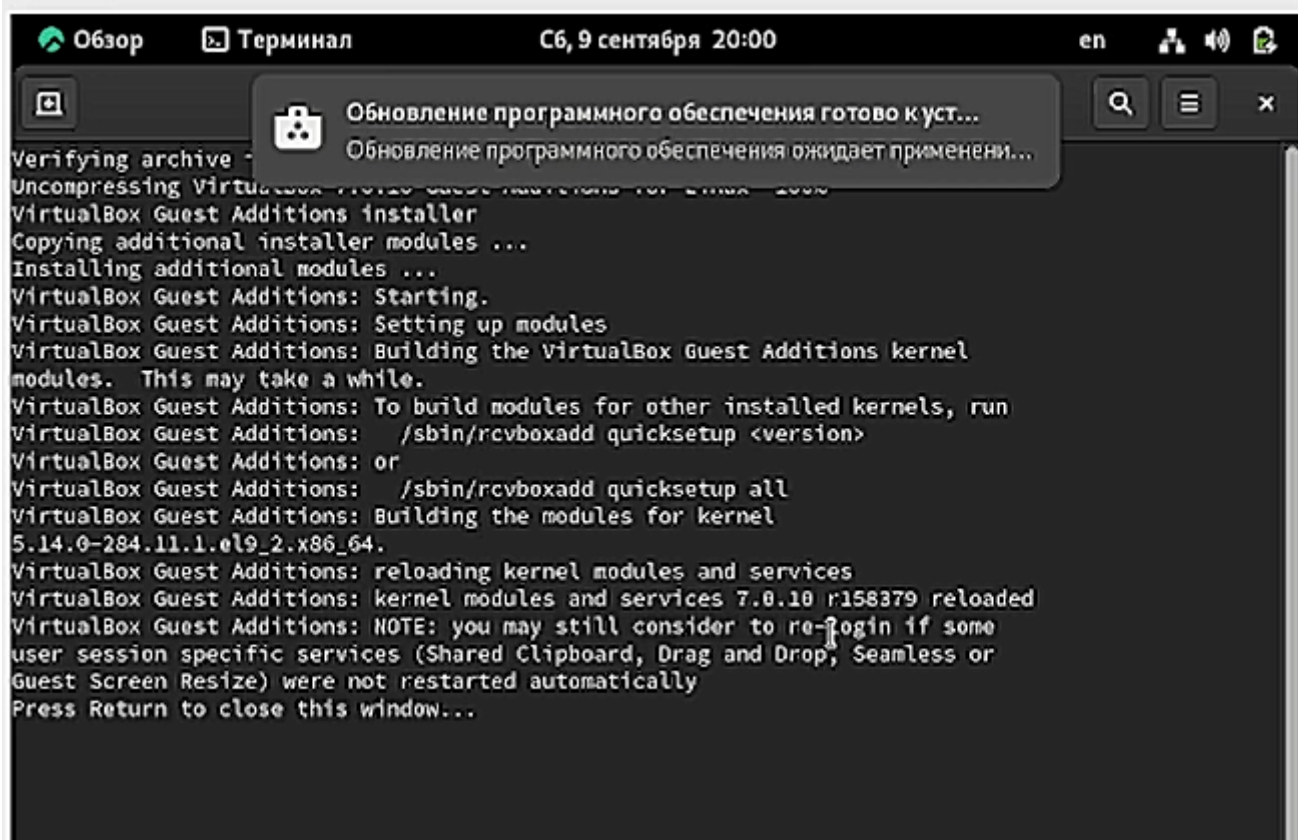


Рисунок 19. Образ дополнений гостевой ОС

Установка и конфигурации ОС на виртуальную машину завершена.

---

## Управление версиями

1. Перед началом нужно создать учетную запись на Github (см. рис. 1)

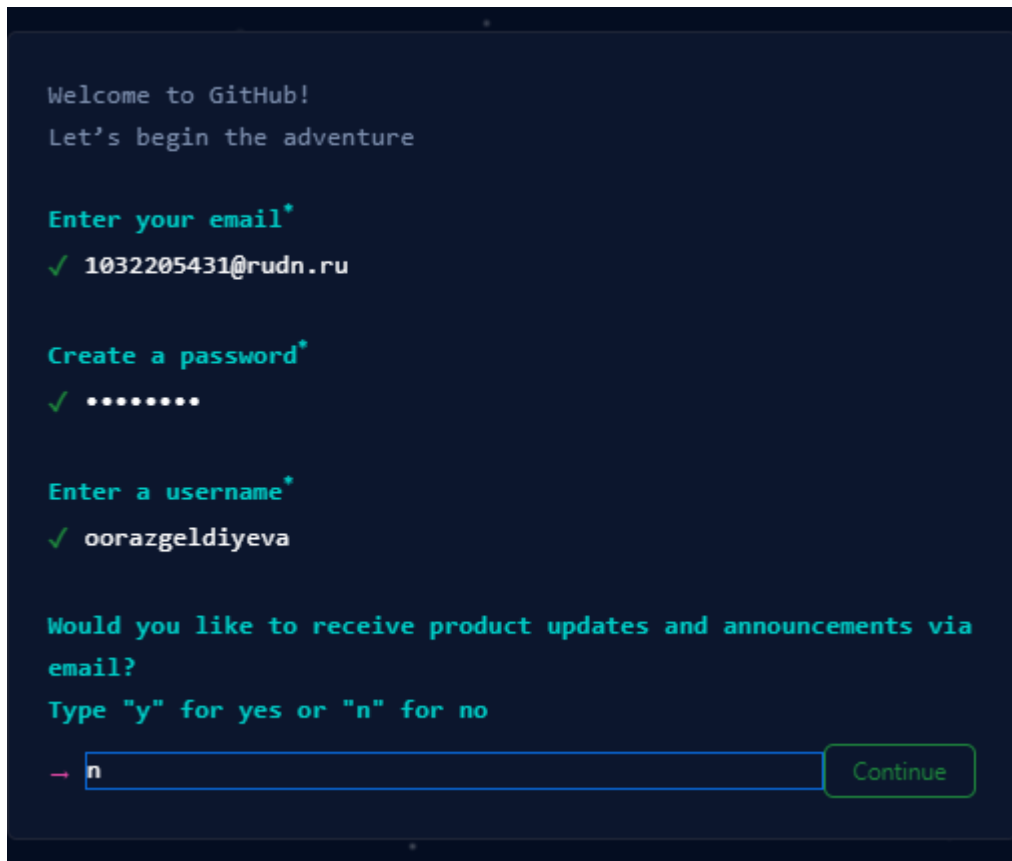


Рисунок 20. Создание

учетной записи

2. Далее откроем терминал и зададим имя и email владельца репозитория (см. рис. 21)
3. Настроим utf-8 в выводе сообщений git (см. рис. 21)
4. Настройте верификацию и подписание коммитов git.
5. Зададим имя начальной ветки (будем называть её master)
6. Конфигурируем параметры autocrlf и safecrlf (см. рис. 21)

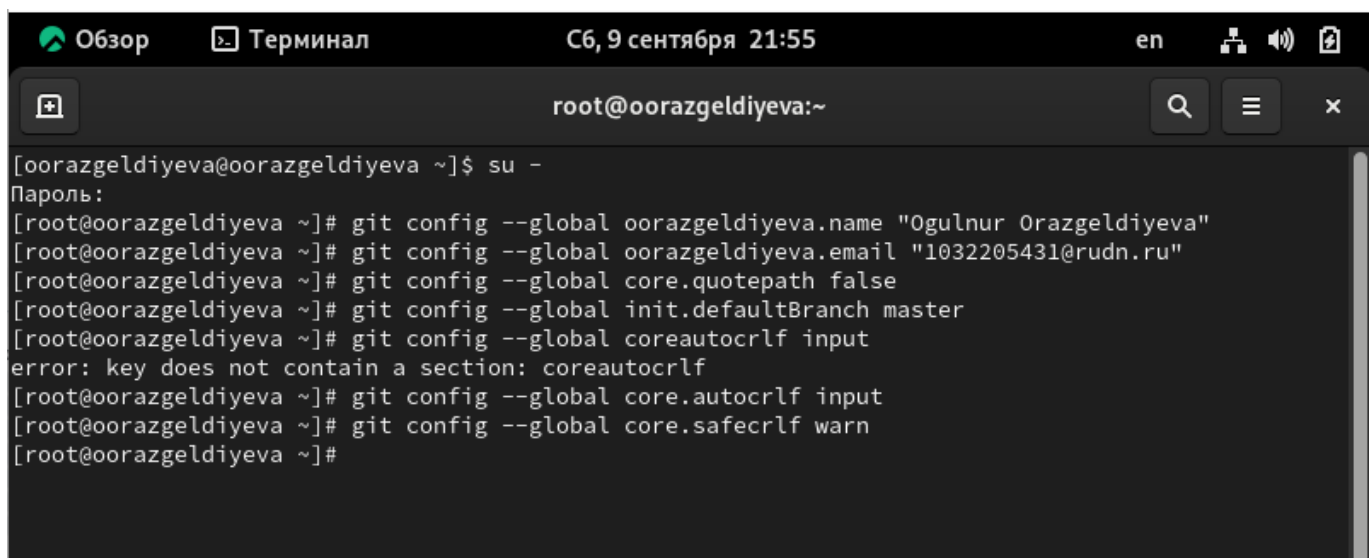


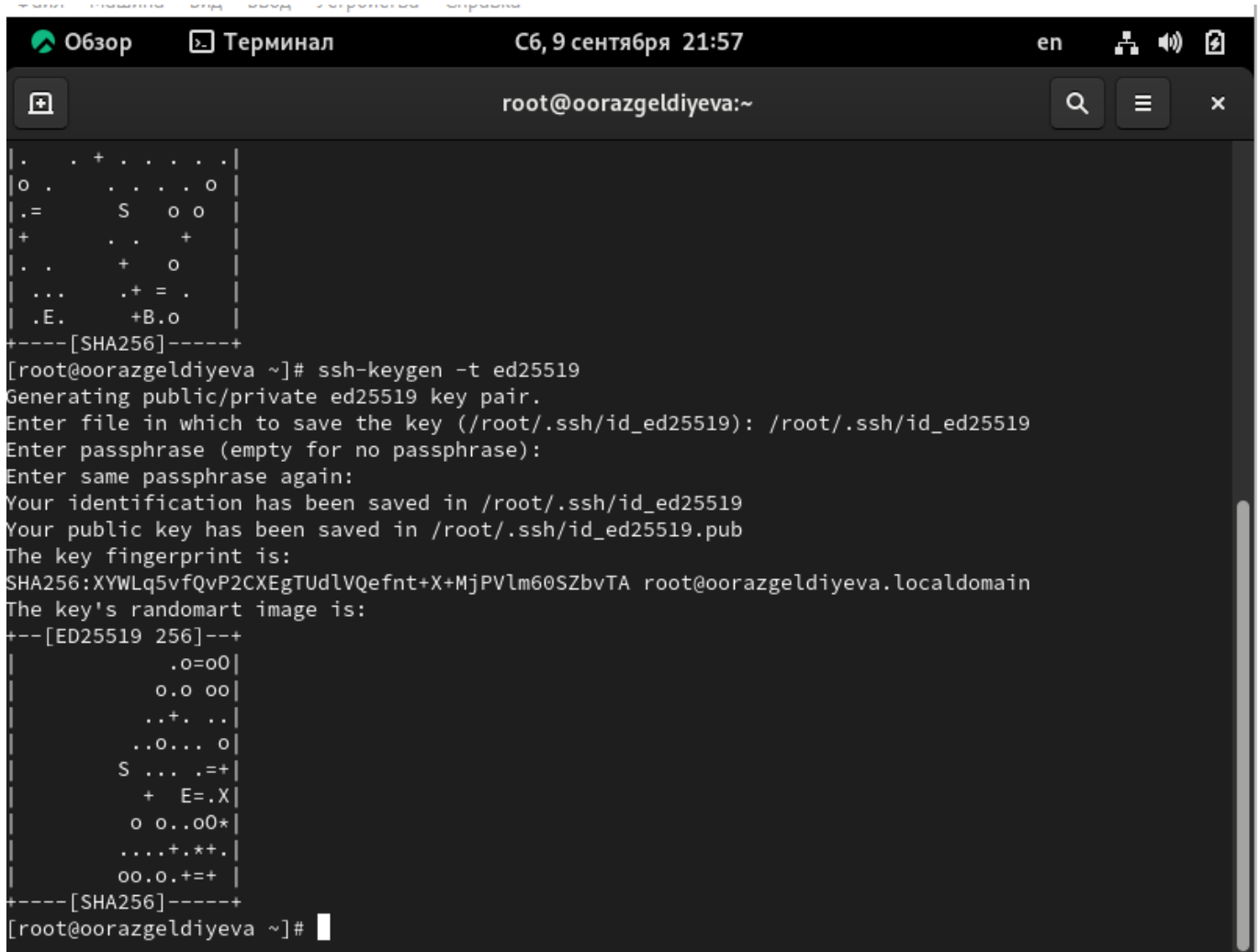
Рисунок 21. Настройка git

7. Создадим ssh-ключ по алгоритму rsa с ключём размером 4096 бит, по алгоритму ed25519 (см. рис. 22-23)



```
[root@oorazgeldiyeva ~]# ssh-keygen -t rsa -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa): /root/.ssh/id_rsa
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa
Your public key has been saved in /root/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:9Qflp7Ka7aHyDvwQwFglzRGyHWDYWBBYIzKknmyit+M root@oorazgeldiyeva.localdomain
The key's randomart image is:
+---[RSA 4096]-----+
|+..o=BB==o . |
|..o..o*..=o. o |
|. . . + . . . . |
|o . . . . . o |
|. = S o o |
|+ . . + |
|. . + o |
|... .+ = . |
|.E. +B.o |
+-----[SHA256]-----+
[root@oorazgeldiyeva ~]#
```

Рисунок 22. Создание ssh-ключа



```
Обзор Терминал С6,9 сентября 21:57 en
root@oorazgeldiyeva:~

|. . + . . . . |
|o . . . . . o |
|. = S o o |
|+ . . + |
|. . + o |
|... .+ = . |
|.E. +B.o |
+-----[SHA256]-----+
[root@oorazgeldiyeva ~]# ssh-keygen -t ed25519
Generating public/private ed25519 key pair.
Enter file in which to save the key (/root/.ssh/id_ed25519): /root/.ssh/id_ed25519
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_ed25519
Your public key has been saved in /root/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:XYWLq5vfQvP2CXEGtUdlVQefnt+X+MjPvIm60SZbvTA root@oorazgeldiyeva.localdomain
The key's randomart image is:
+---[ED25519 256]--+
| .o=o0 |
| o.o oo |
| ..+. .. |
| ..0... o |
| S ... .+= |
| + E=.X |
| o o..o0* |
| ....+.++. |
| oo.o.+=+ |
+-----[SHA256]-----+
[root@oorazgeldiyeva ~]#
```

Рисунок 23. Создание ssh-ключа

- Теперь копируем сгенерированный ключ и добавляем его в настройках профиля на github (см. рис. 24-26)

```
[root@oorazgeldiyeva ~]# cat /root/.ssh/id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQCN0gUV55aKoKbE+7ZcqpzviB1RRFBzXv9SDEsCGfaAey3k9JDkP+0j7Sdb2S
kjU02RF6GRz5ETyUKg0xZxUvujPM53JEL4we0oImZE7foDmwKhMBj4NM5bbpNyOYSqd2RUpgNuWZhwq34JucK9RcF9pA8QMa81
xy/5eQ6LGJUKWC++faXrE8WN9I2DKXI633buvrenTKHFFpxoHqKrNsU2tONsII2ujLIYYZynUYhm1DXHj3odJeMN7lQ34u77Fm
Chbwqa0inH0U6RbSBpij7bwWsbXs90Ua7v5kyueLR887lqbE6CwLQ7Ye40ASPU+VoWFUCngoTKe6YkGtyOIVL93djNCF7dze1n
KJ1ybDa5hmnX0aBCor/YnnrQYaA0b9ZDW6+h87tDLpsadLKLxQ4NDSYoSg08c+5AXI1qSAq1yF1kUmbcZWrAQBMD4BlvKCvjM
joNGmBDHa4OnPmWqtSFFwZ4ZAhslIwM3t4UC24EhZWf5078DEi1w5/dWATa/saiN5h6ZcQWYpxIikURqLpsfkNFeSRA8SwhB/Z
btIs6UYXb1r4o2oOeKRgR1n8bqvHv/SFZyHda10BRo/mImh9GBDOj0mQkx5YI4tgA5dA/QlXFESB1e/7Atc6FIyLeOX8dRupP6
FvgYLIgoDCVKVaIXyUoijxltYWNWqAqJT9ow== root@oorazgeldiyeva.localdomain
[root@oorazgeldiyeva ~]#
```

Рисунок 24. Копирование ssh-ключа

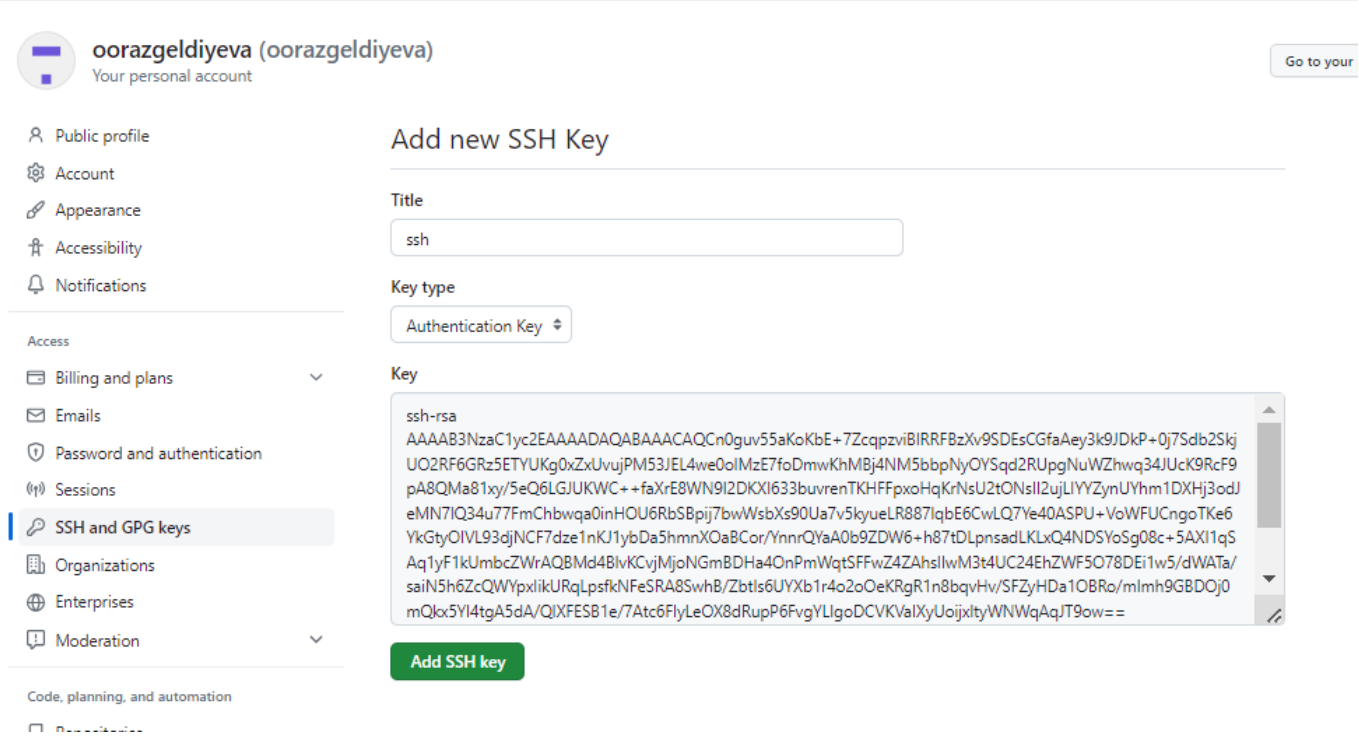


Рисунок 25. Добавление ssh-ключа

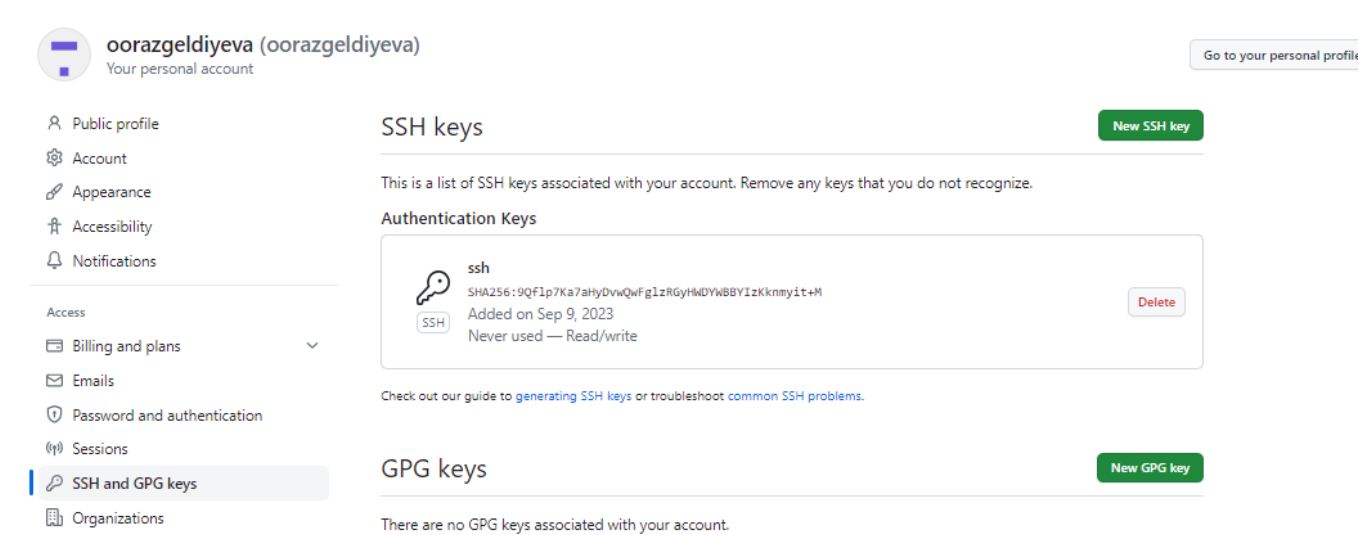


Рисунок 26. Добавление ssh-ключа

9. Теперь создаем репозиторий с помощью заданного нам шаблона (см. рис. 27-28)

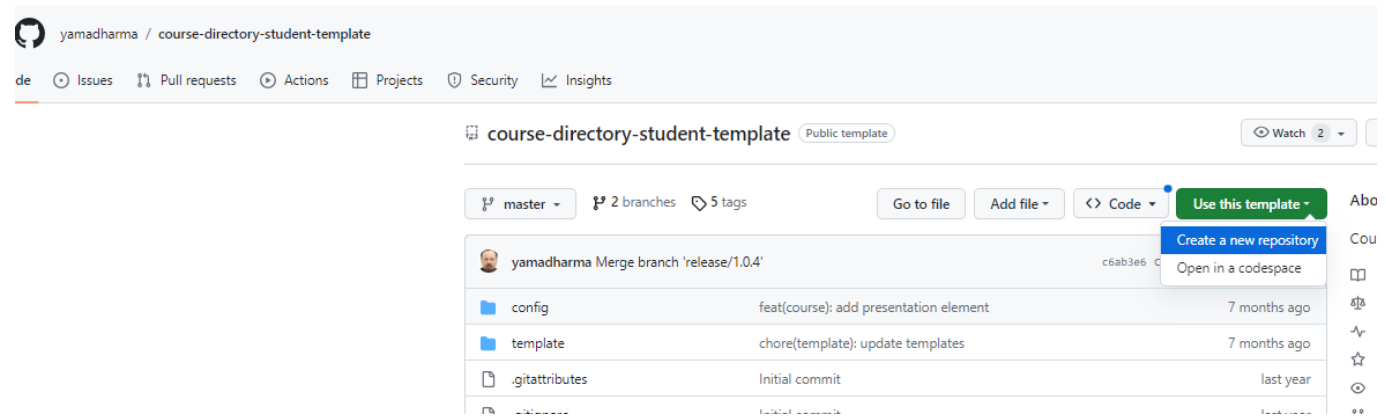


Рисунок 27. Создание репозитория

## Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Required fields are marked with an asterisk (\*).

Owner \*

oorazgeldiyeva



Repository name \*

study\_2023-2024

✔ study\_2023-2024 is available.

Great repository names are short and memorable. Need inspiration? How about **studious-spoon** ?

Description (optional)

- ☒  **Public**  
Anyone on the internet can see this repository. You choose who can commit.
- ☐  **Private**  
You choose who can see and commit to this repository.

 You are creating a public repository in your personal account.

Create repository

Рисунок 28. Создание репозитория

10. Создаем директории и клонируем их (см. рис. 29-30)

```

root@oorazgeldiyeva:~/work/study/2023-2024/Информационная безопасность
chbwqa0inH0U6RbSBpij7bwWsbXs90Ua7v5kyueLR887lqbE6CwLQ7Ye40ASPU+VoWFUCngoTKe6YkGty0IVL93dj
KJlybDa5HmnX0aBCor/YnnrQYaA0b9ZDW6+h87tDLpsadLKLxQ4NDSYoSg08c+5AXI1qSAqlyF1kUmbcZWrAQBMd
joNGmBDHa40nPmWqtSFFwZ4ZAhsliWm3t4UC24EhZWF5078DEilw5/dWATa/saiN5h6ZcQWYpxIikURqLpsfkNFES
otIs6UYXbl1r4o2o0eKRGRI1n8bqvHv/SFZYHda10BRo/mImh9GBD0j0mQkx5YI4tgA5dA/QLXFESB1e/7Atc6FIyLe
FvgYLIgoDCVKVaIXyUoijxltyWNWqAqJT9ow== root@oorazgeldiyeva.localdomain
[root@oorazgeldiyeva ~]#
[root@oorazgeldiyeva ~]#
[root@oorazgeldiyeva ~]# mkdir -p ~/work/study/2023-2024/Информационная безопасность
[root@oorazgeldiyeva ~]# cd ~/work/study/2023-2024/Информационная безопасность

```

Рисунок 29. Создание директории

```

[root@oorazgeldiyeva Информационная безопасность]# git clone --recursive git@github.com:oorazgeldiyeva/study_2023-2024.git infosec
Клонирование в «infosec»...
remote: Enumerating objects: 27, done.
remote: Counting objects: 100% (27/27), done.
remote: Compressing objects: 100% (26/26), done.
remote: Total 27 (delta 1), reused 11 (delta 0), pack-reused 0
Получение объектов: 100% (27/27), 16.94 КиБ | 403.00 КиБ/с, готово.
Определение изменений: 100% (1/1), готово.
Подмодуль «template/presentation» (https://github.com/yamadharma/academic-presentation-markdown-template.git) зарегистрирован по пути «template/presentation»
Подмодуль «template/report» (https://github.com/yamadharma/academic-laboratory-report-template.git) зарегистрирован по пути «template/report»
Клонирование в «/root/work/study/2023-2024/Информационная безопасность/infosec/template/presentation»...
remote: Enumerating objects: 82, done.
remote: Counting objects: 100% (82/82), done.
remote: Compressing objects: 100% (57/57), done.
remote: Total 82 (delta 28), reused 77 (delta 23), pack-reused 0
Получение объектов: 100% (82/82), 92.90 КиБ | 932.00 КиБ/с, готово.
Определение изменений: 100% (28/28), готово.
Клонирование в «/root/work/study/2023-2024/Информационная безопасность/infosec/template/report»...
remote: Enumerating objects: 101, done.
remote: Counting objects: 100% (101/101), done.
remote: Compressing objects: 100% (70/70), done.
remote: Total 101 (delta 40), reused 88 (delta 27), pack-reused 0
Получение объектов: 100% (101/101), 327.25 КиБ | 915.00 КиБ/с, готово.
Определение изменений: 100% (40/40), готово.
Submodule path 'template/presentation': checked out 'b1be3800ee91f5809264cb755d316174540b753e'
Submodule path 'template/report': checked out '1d1b61dcac9c287a83917b82e3aef11a33b1e3b2'
[root@oorazgeldiyeva Информационная безопасность]#

```

Рисунок 30. Клонирование в git

11. Переходим к созданный каталог, удаляем ненужный файл и создаем нужные нам каталоги (см. рис. 31)



```
[root@oorazgeldiyeva Информационная безопасность]# ls
infosec
[root@oorazgeldiyeva Информационная безопасность]# cd infosec
[root@oorazgeldiyeva infosec]# ls
CHANGELOG.md  COURSE  Makefile      README.en.md      README.md
config        LICENSE  package.json  README.git-flow.md  template
[root@oorazgeldiyeva infosec]# rm package.jaon
rm: невозможно удалить 'package.jaon': Нет такого файла или каталога
[root@oorazgeldiyeva infosec]# rm package.json
rm: удалить обычный файл 'package.json'? y
[root@oorazgeldiyeva infosec]# make COURSE=infosec
[root@oorazgeldiyeva infosec]# git add .
[root@oorazgeldiyeva infosec]# git commit -am 'feat(main): make course structure'
[master f33f0ac] feat(main): make course structure
Committer: root <root@oorazgeldiyeva.localdomain>
Ваше имя или электронная почта настроены автоматически на основании вашего
имени пользователя и имени машины. Пожалуйста, проверьте, что они
определены правильно.
Вы можете отключить это уведомление установив их напрямую:

    git config --global user.name "Ваше Имя"
    git config --global user.email you@example.com

После этого, изменить авторство этой коммита можно будет с помощью команды:

    git commit --amend --reset-author

150 files changed, 41044 insertions(+), 14 deletions(-)
create mode 100644 labs/README.md
create mode 100644 labs/README.ru.md
create mode 100644 labs/lab1/presentation/Makefile
create mode 100644 labs/lab1/presentation/image/kulyabov.jpg
create mode 100644 labs/lab1/presentation/presentation.md
create mode 100644 labs/lab1/report/Makefile
create mode 100644 labs/lab1/report/bib/cite.bib
create mode 100644 labs/lab1/report/image/placeimg_800_600_tech.jpg
create mode 100644 labs/lab1/report/pandoc/csl/gost-r-7-0-5-2008-numeric.csl
create mode 100755 labs/lab1/report/pandoc/filters/pandoc_eqnos.py
create mode 100755 labs/lab1/report/pandoc/filters/pandoc_fignos.py
create mode 100755 labs/lab1/report/pandoc/filters/pandoc_secnos.py
create mode 100755 labs/lab1/report/pandoc/filters/pandoc_tablenos.py
create mode 100644 labs/lab1/report/pandoc/filters/pandocxnos/__init__.py
create mode 100644 labs/lab1/report/pandoc/filters/pandocxnos/core.py
create mode 100644 labs/lab1/report/pandoc/filters/pandocxnos/main.py
create mode 100644 labs/lab1/report/pandoc/filters/pandocxnos/pandocattributes.py
create mode 100644 labs/lab1/report/report.md
create mode 100644 labs/lab2/presentation/Makefile
```

Рисунок 31. Создание каталогов

12. Отправляем файлы на сервер, создавая при этом коммит (см. рис. 31-32)

```
create mode 100644 presentation/report/report.md
[root@oorazgeldiyeva infosec]# git push
Перечисление объектов: 35, готово.
Подсчет объектов: 100% (35/35), готово.
Сжатие объектов: 100% (29/29), готово.
Запись объектов: 100% (34/34), 342.08 КиБ | 2.87 МиБ/с, готово.
Всего 34 (изменений 4), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
remote: Resolving deltas: 100% (4/4), completed with 1 local object.
To github.com:oorazgeldiyeva/study_2023-2024.git
 5c44609..f33f0ac master -> master
[root@oorazgeldiyeva infosec]#
```

Рисунок 32. Отправление файлов на сервер

## Вывод

На лабораторной работе приобрели практические навыки установки операционной системы на виртуальную машину, настройки минимально необходимых для дальнейшей работы сервисов; изучили идеологию и применение средств контроля версий; освоили умения по работе с git.

---

### **Источники**

1. Лабораторная работа №1. Установка и конфигурация операционной системы на виртуальную машину
2. Лабораторная работа № 2. Управление версиями