Шаблон отчёта по лабораторной работе 5

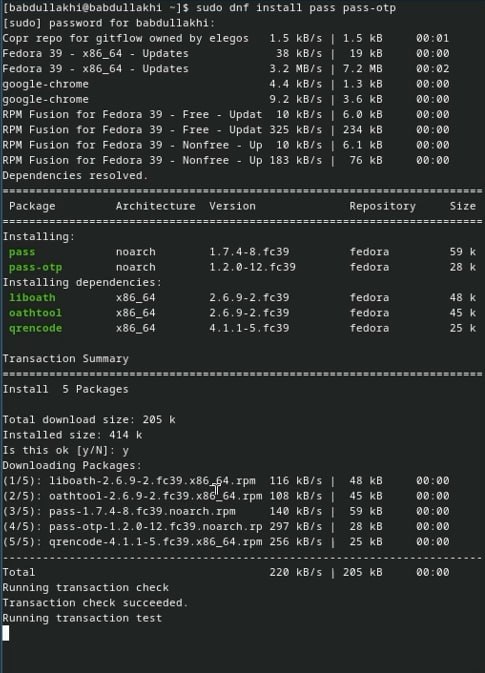
Простейший вариант

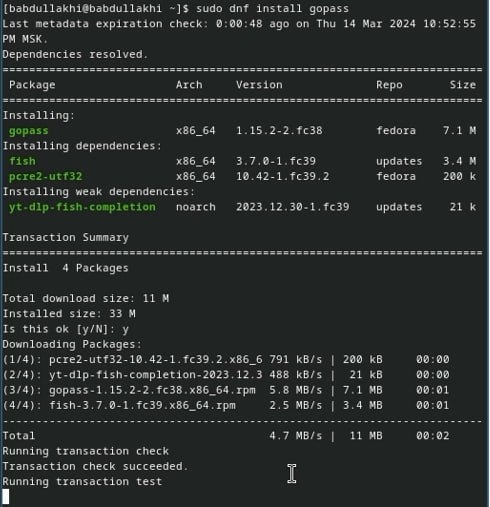
Абдуллахи Бахара

Содержание

# 1 Выполнение лабораторной работы

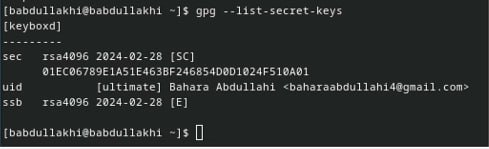
* Менеджер паролей pass:
* Установка
* pass , dnf install pass pass-otp
* gopass , dnf install gopass





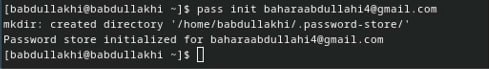
## 1.1 Настройка:

* Ключи GPG : Просмотр списка ключей



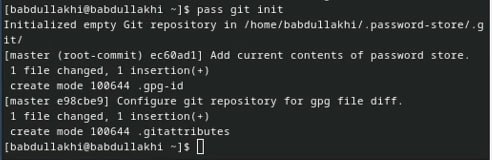
# 2 Инициализация хранилища:

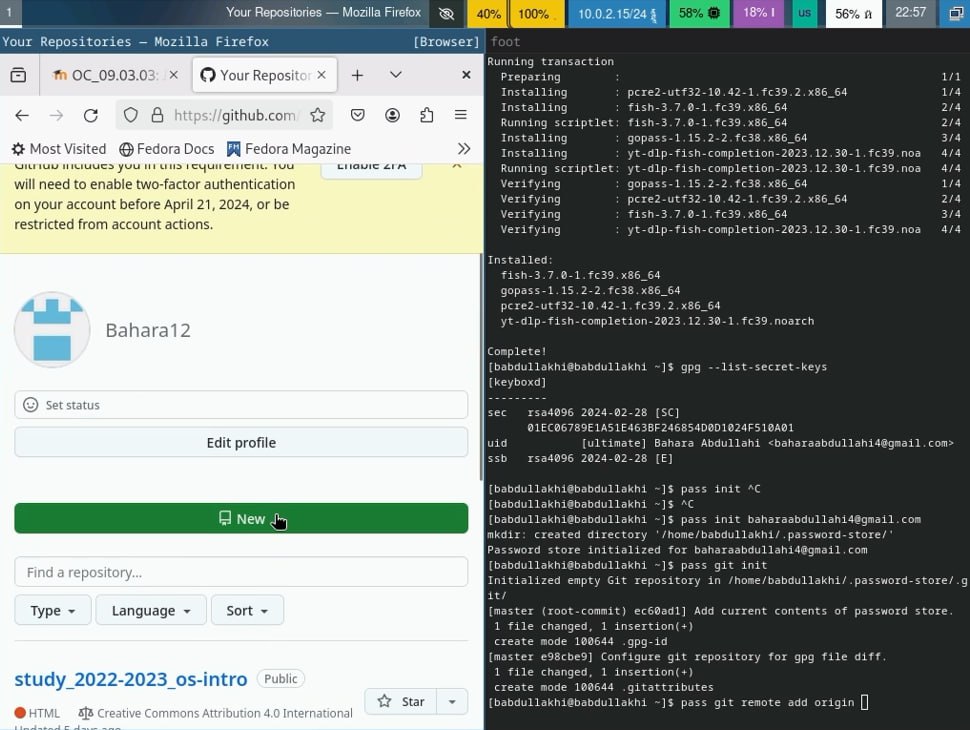
* Инициализируем хранилище: pass init

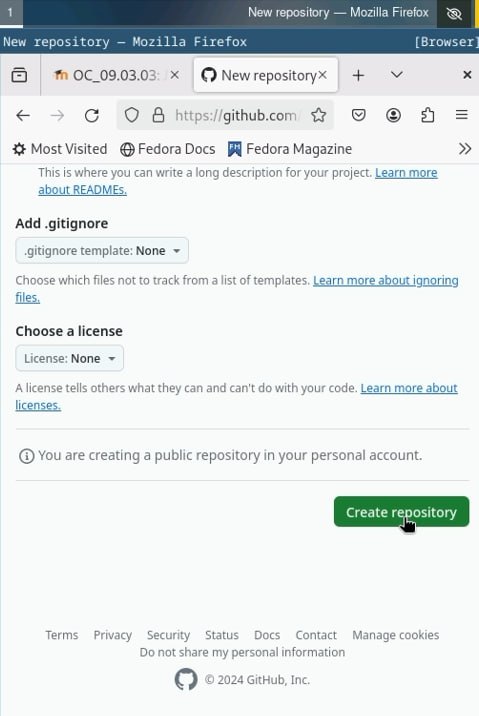


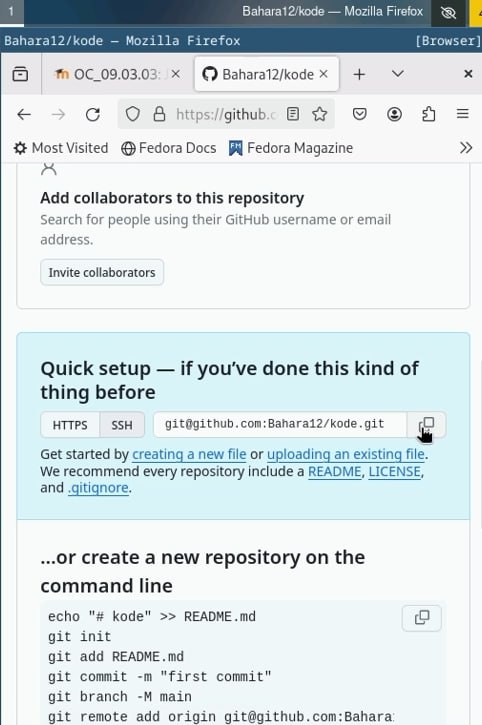
# 3 Синхронизация с git:

* Создадим структуру git:
* pass git init:





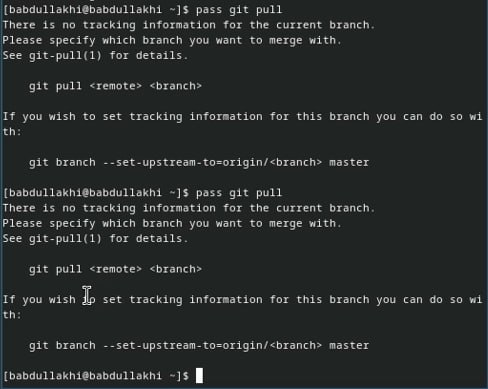


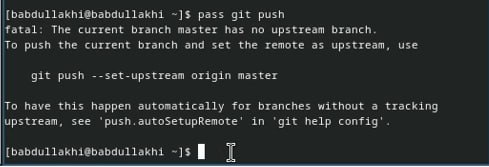


* Также можно задать адрес репозитория на хостинге (репозиторий необходимо предварительно создать)

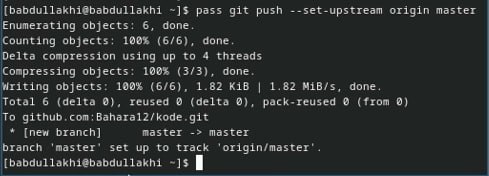


* Для синхронизации выполняется следующая команда:
* pass git pull





* pass git push

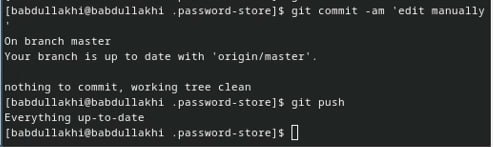


{#fig:001 width=100%

## 3.1 Прямые изменения:

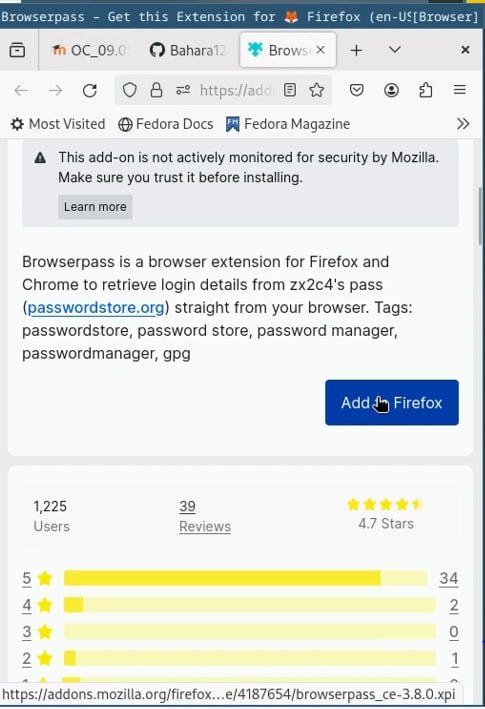
* Следует заметить, что отслеживаются только изменения, сделанные через сам gopass (или pass).

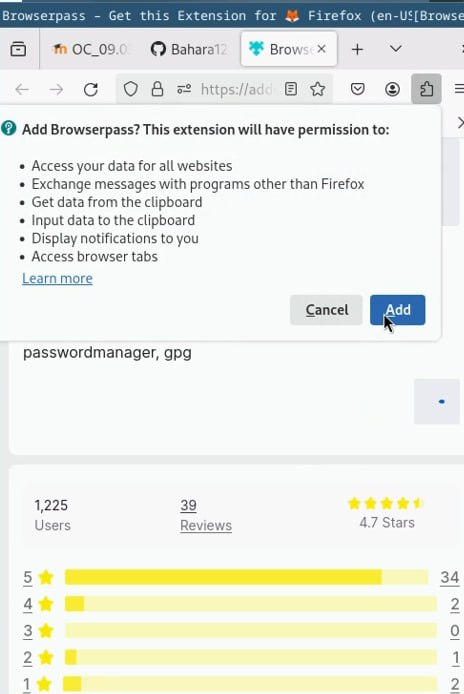
{#fig:001 width=100%

{#fig:001 width=100%

## 3.2 Настройка интерфейса с броузером:

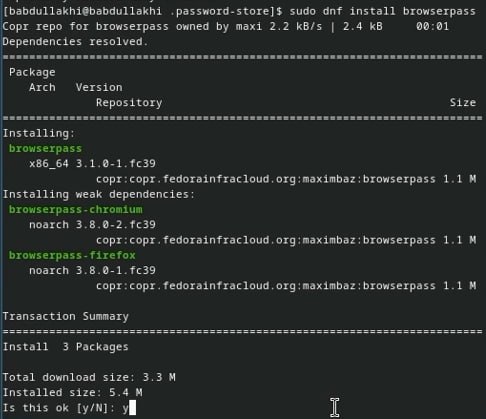
* Для взаимодействия с броузером используется интерфейс native messaging.
* Плагин browserpass: Плагин для Firefox: https://addons.mozilla.org/en-US/firefox/addon/browserpass-ce/.

{#fig:001 width=100%

{#fig:001 width=100%

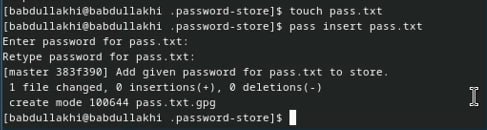
* Fedora:
* dnf copr enable maximbaz/browserpass
* dnf install browserpass

{#fig:001 width=100%

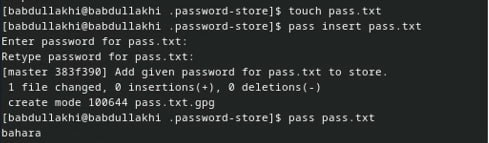
{#fig:001 width=100%

## 3.3 Сохранение пароля:

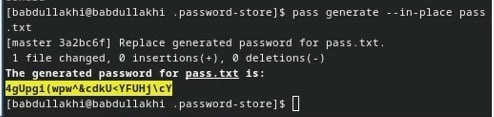
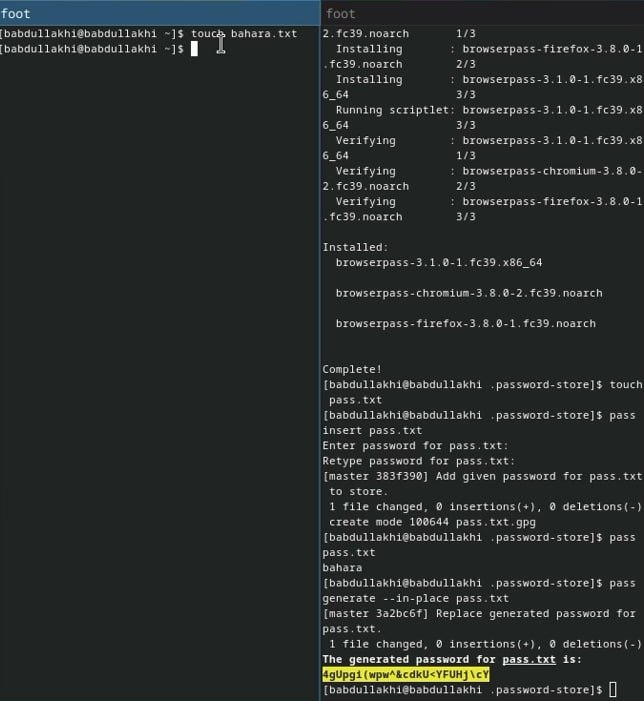
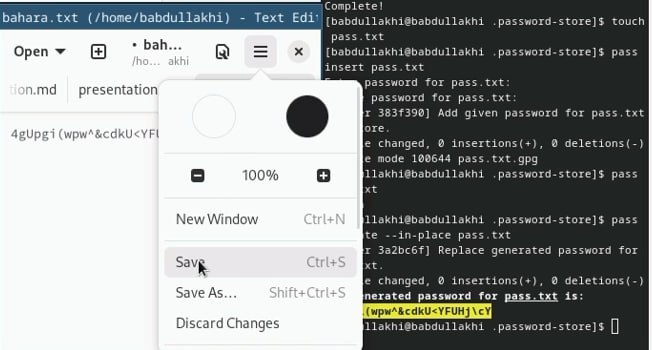
* Добавить новый файла;
* touch pass.txt

{#fig:001 width=100% - Добавить новый пароль

* pass insert [OPTIONAL DIR]/[FILENAME]

{#fig:001 width=100%

* Отобразите пароль для указанного имени файла:
* Замените существующий пароль:

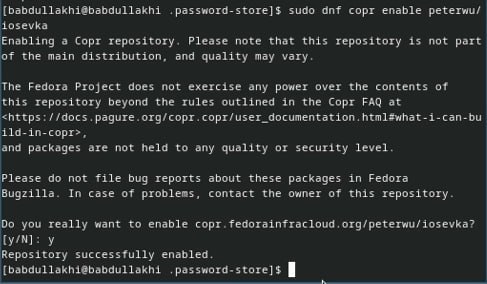
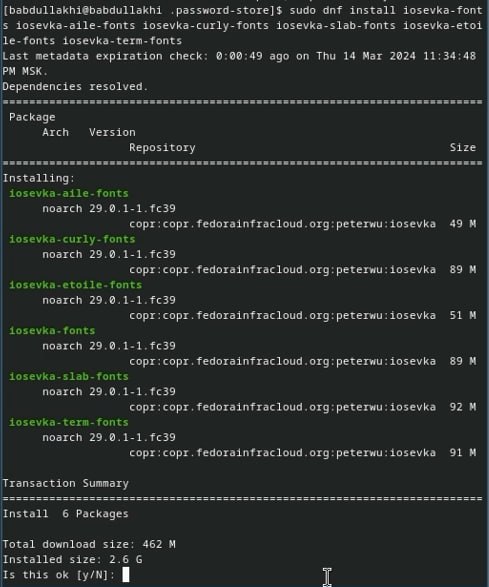
{#fig:001 width=100% {#fig:001 width=100% {#fig:001 width=100%

## 3.4 Управление файлами конфигурации:

* Дополнительное программное обеспечение;
* Установите дополнительное программное обеспечение:

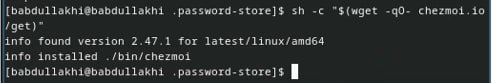
{#fig:001 width=100%

* установите шрифты:
* sudo dnf copr enable peterwu/iosevka

{#fig:001 width=100% sudo dnf search iosevka{#fig:001 width=100% {#fig:001 width=100%

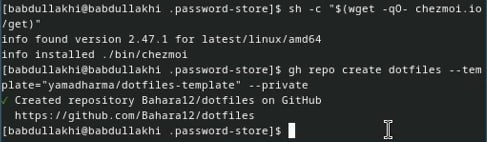
## 3.5 Установка:

* Установка бинарного файла. Скрипт определяет архитектуру процессора и операционную систему и скачивает необходимый файл:
* с помощью wget:

{#fig:001 width=100%

## 3.6 Создание собственного репозитория с помощью утилит:

* Будем использовать утилиты командной строки для работы с github.
* Создадим свой репозиторий для конфигурационных файлов на основе шаблона:

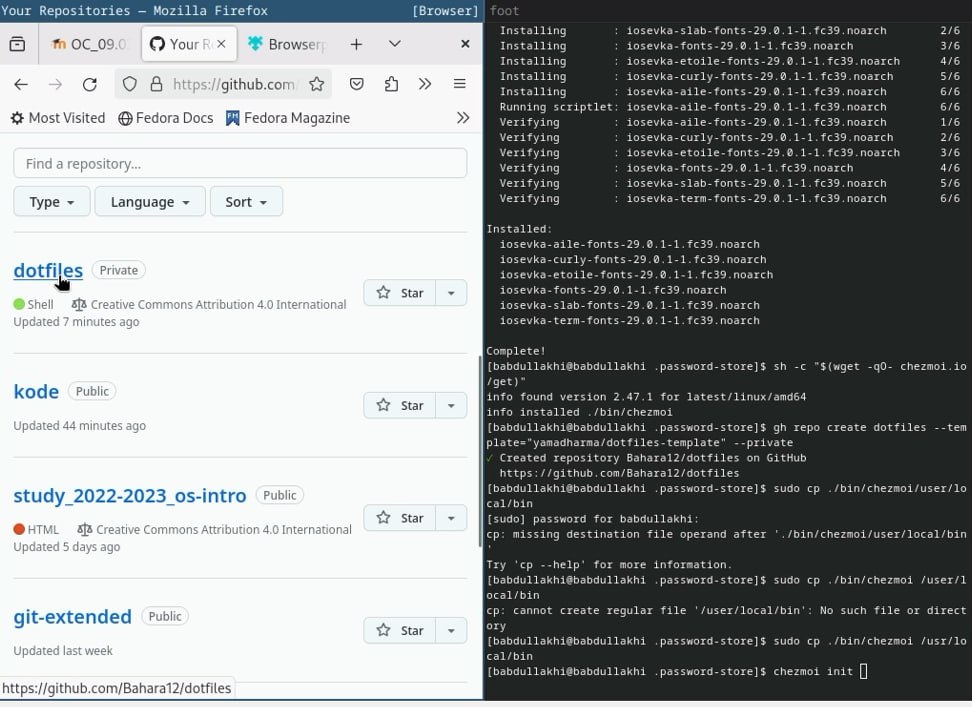
{#fig:001 width=100%

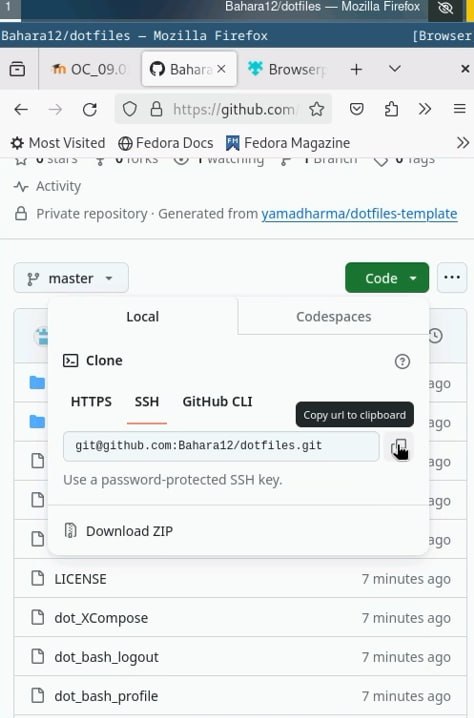
## 3.7 Подключение репозитория к своей системе:

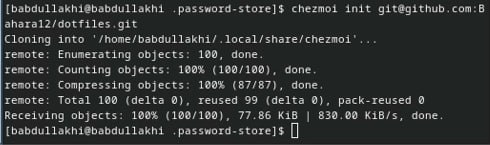
* копиравала chezmoi с помощью этого команда :sudo cp ./bin/chezmoi /usr/local/bin

копиравала chezmoi с помощью этого команда{#fig:001 width=100%

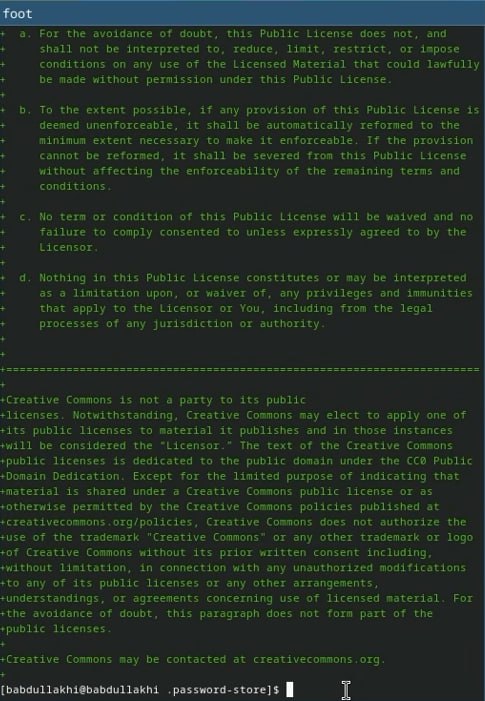
* Инициализируйте chezmoi с вашим репозиторием dotfiles:

{#fig:001 width=100%

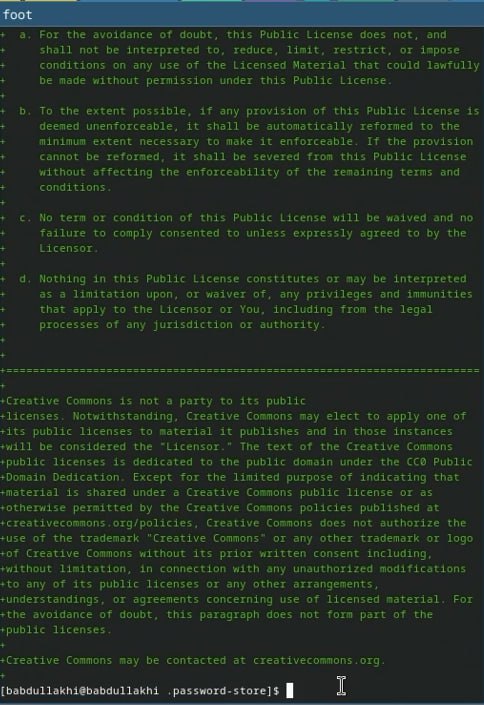
{#fig:001 width=100%

{#fig:001 width=100%

* Проверьте, какие изменения внесёт chezmoi в домашний каталог, запустив:

{#fig:001 width=100%

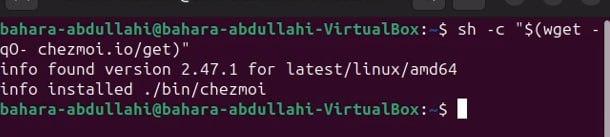
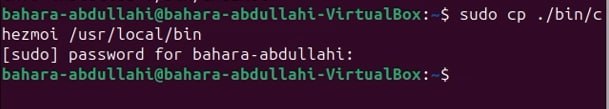
* Если вас устраивают изменения, внесённые chezmoi, запустите:

{#fig:001 width=100%

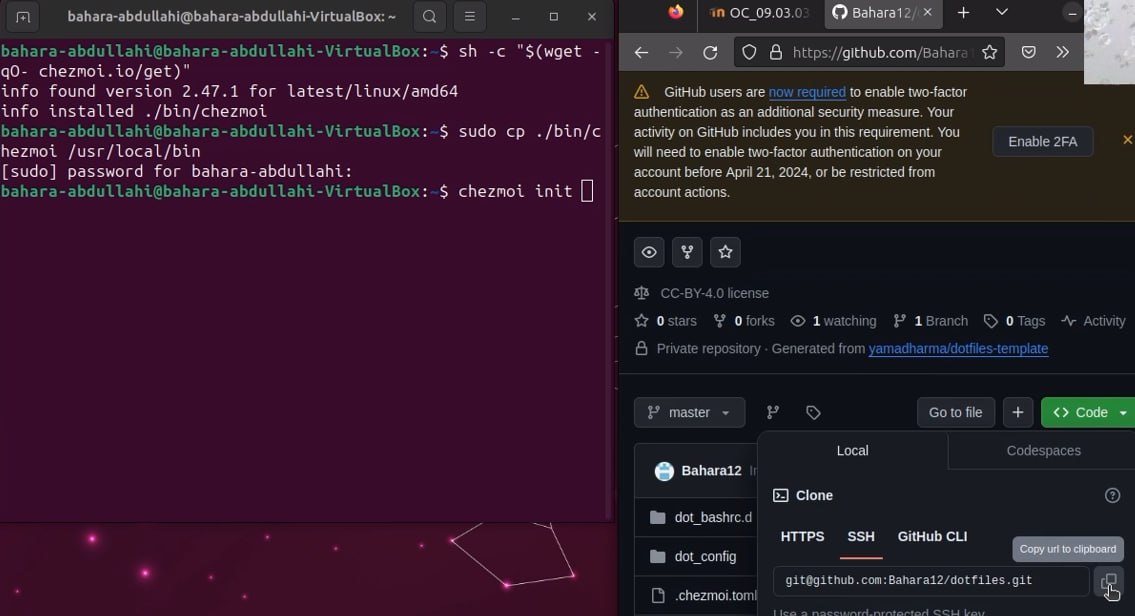
## 3.8 Использование chezmoi на нескольких машинах:

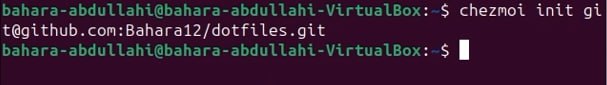
-Установка; - Установка бинарного файла. Скрипт определяет архитектуру процессора и операционную систему и скачивает необходимый файл:

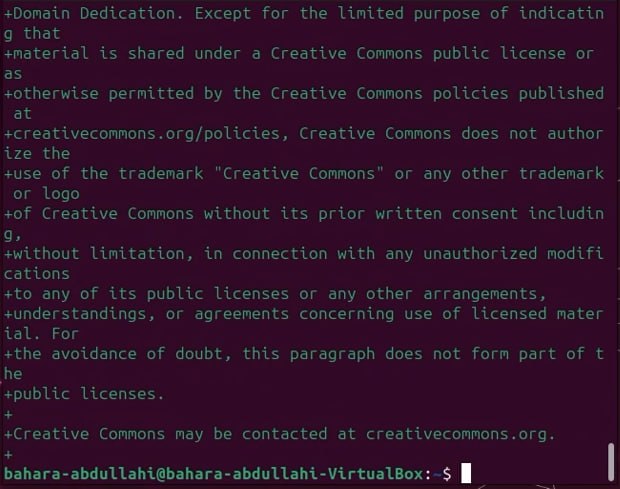
* с помощью wget: sh -c “$(wget -qO- chezmoi.io/get)”

{#fig:001 width=100% {#fig:001 width=100%

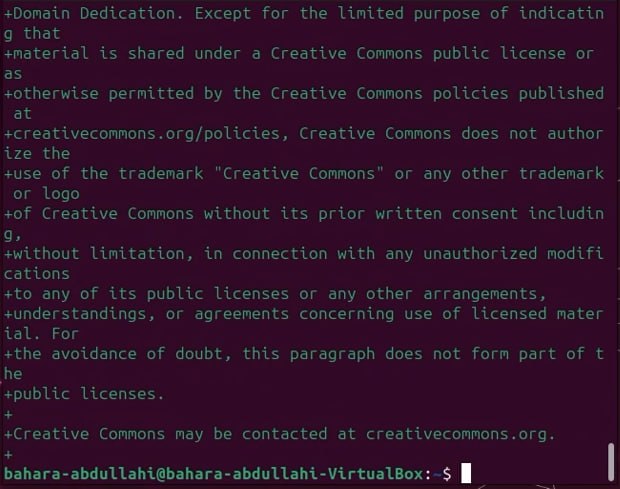
* На второй машине инициализируйте chezmoi с вашим репозиторием dotfiles:
* копиравала chezmoi с помощью этого команда в ubuntu :sudo cp ./bin/chezmoi /usr/local/bin

{#fig:001 width=100%

{#fig:001 width=100%

{#fig:001 width=100%

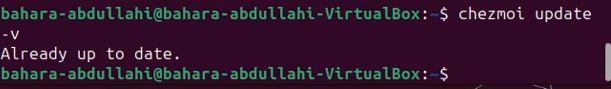
* При существующем каталоге chezmoi можно получить и применить последние изменения из вашего репозитория:

{#fig:001 width=100%

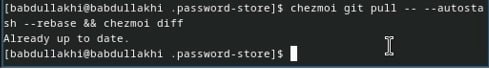
## 3.9 Настройка новой машины с помощью одной команды:

# 4 Ежедневные операции c chezmoi:

* Извлеките последние изменения из репозитория и примените их
* Можно извлечь изменения из репозитория и применить их одной командой:

{#fig:001 width=100%

* Извлеките последние изменения из своего репозитория и посмотрите, что изменится, фактически не применяя изменения:

{#fig:001 width=100% {#fig:001 width=100%

* Это запускается git pull –autostash –rebase в вашем исходном каталоге, а chezmoi diff затем показывает разницу между целевым состоянием, вычисленным из вашего исходного каталога, и фактическим состоянием.
* Если вы довольны изменениями, вы можете применить их:

{#fig:001 width=100%

# 5 Автоматически фиксируйте и отправляйте изменения в репозиторий;

* Можно автоматически фиксировать и отправлять изменения в исходный каталог в репозиторий.
* Чтобы включить её, добавьте в файл конфигурации ~/.config/chezmoi/chezmoi.toml следующее:

[git] autoCommit = true autoPush = true

* ~/.config/chezmoi/chezmoi.toml следующее:

{#fig:001 width=100%

{#fig:001 width=100%