Отчёт по лабораторной работе №5

Дисциплина: Операционные системы

Верниковская Екатерина Андреевна

Содержание

# 1 Цель работы

Настроить рабочую среду и получить навыки работы с менеджером паролей pass и с chezmoi.

# 2 Задание

1. Установить gopass и настроить всё, что надо.
2. Настроить интерфейс с браузером.
3. Установить chezmoi.
4. Использовать chezmoi на нескольких машинах.

# 3 Выполнение лабораторной работы

## 3.1 Менеджер паролей pass

### 3.1.1 Установка

Установим pass и gopass, с помощью двух команд: *dnf install pass pass-otp* и *dnf install gopass* (рис. 1), (рис. 2)

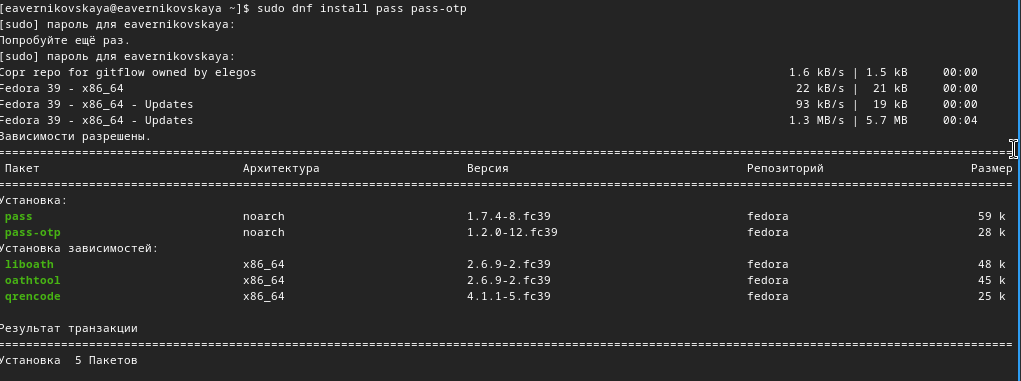


Рис. 1: Установка pass

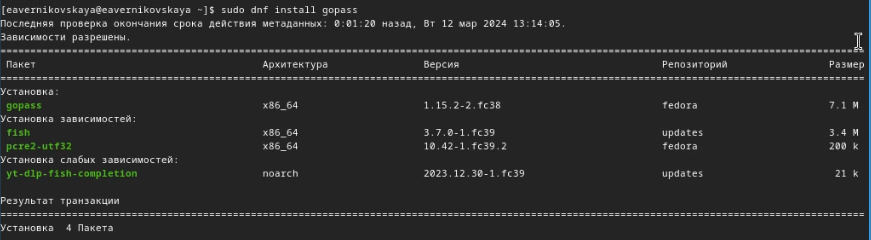


Рис. 2: Установка gopass

### 3.1.2 Настройка

Просмотрим список ключей, введя *gpg –list-secret-keys* (рис. 3)

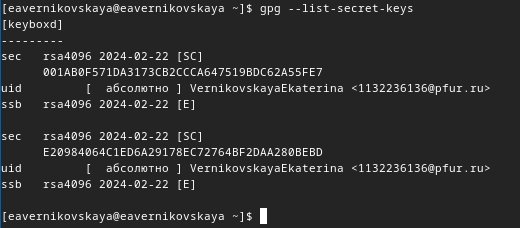


Рис. 3: Список ключей

Далее инициализируем хранилище командой *pass init* . Я ввела email (рис. 4)

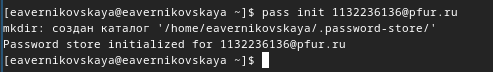


Рис. 4: Инициализация хранилища

Создаём структуру с помощью *pass git init* (рис. 5)

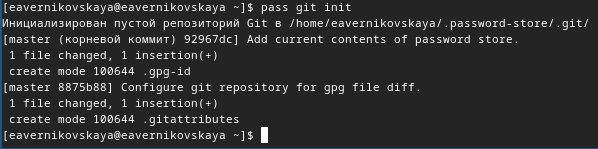


Рис. 5: Создание структуры git

Далее создаём репозиторий с именем pass и задаём адрес репозитория на хостинге, введя *pass git remote add origin https://github.com//.git* (рис. 6), (рис. 7)



Рис. 6: Создание репозитория

Адрес репозитория на хостинге

Рис. 7: Адрес репозитория на хостинге

Потом выполняем синхронизацию с помощью двух команд *pass git push* и *pass git pull* (рис. 8)

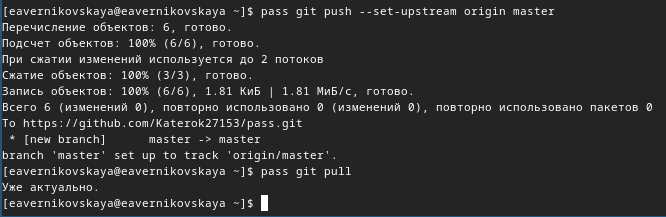


Рис. 8: Синхронизация

Далее вручную коммитим и выкладываем изменения (рис. 9)

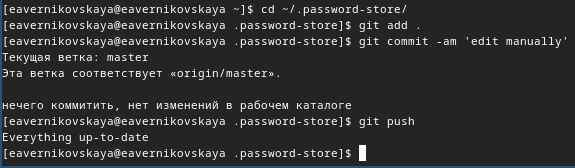


Рис. 9: Выкладываем изменения

Проверяем статус синхронизации командой *pass git status* (рис. 10)

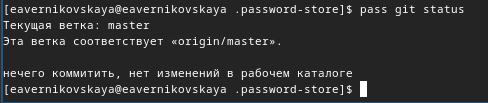


Рис. 10: Проверка статуса синхронизации

### 3.1.3 Настройка интерфейса с браузером

Переходим на сайт с плагином для Firefox и добавляем этот плагин (рис. 12), (рис. 12)

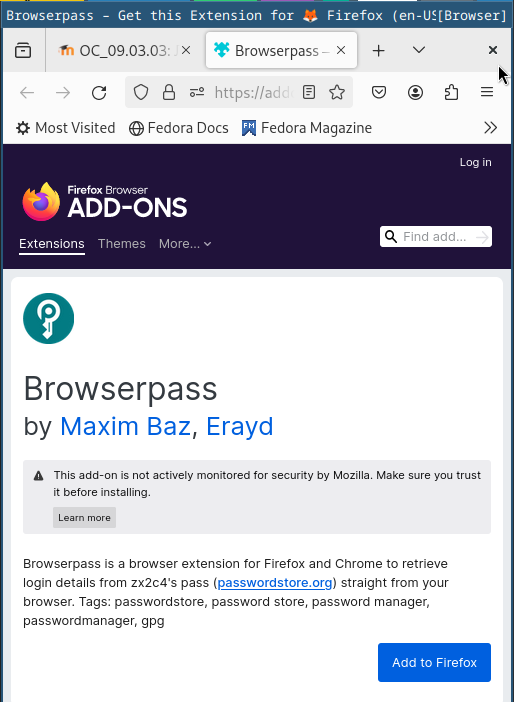


Рис. 11: Сайт с плагтином для Firefox

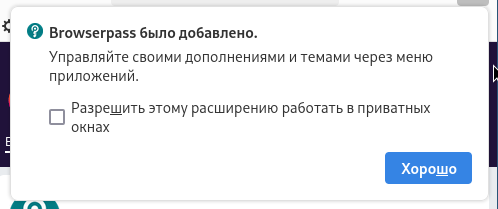


Рис. 12: Добавление плагина

Далее подключаем репозиторий и скачиваем browserpass, с помощью двух команд: *dnf copr enable maximbaz/browserpass* и *dnf install browserpass* (рис. 13), (рис. 14)

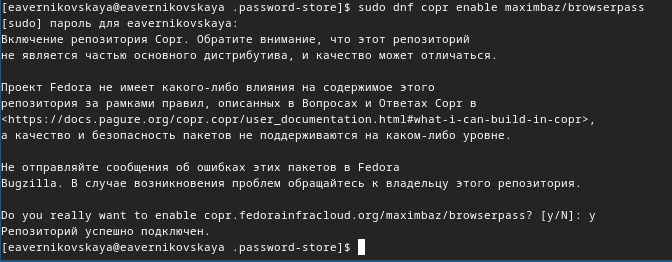


Рис. 13: Подключение репозитория

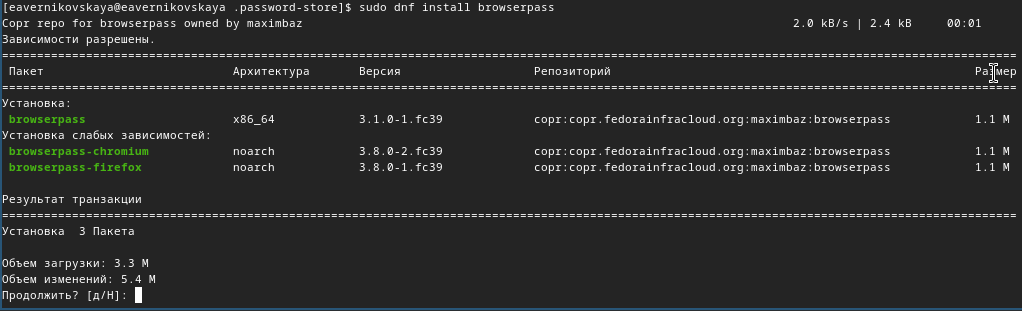


Рис. 14: Установка browserpass

### 3.1.4 Сохранение пароля

Создаём новый файл с помощью *touch* и добавляем пароль с помощью команды *pass insert [FILENAME]* (рис. 15)

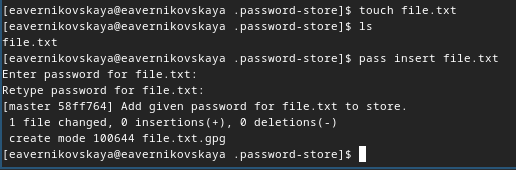


Рис. 15: Добавление пароля

Отобразить пароль для указанного файла можно командой *pass [FILENAME]* (рис. 16)

Отображение пароля

Рис. 16: Отображение пароля

Далее можно заменить существующий пароль командой *pass generate –in-place FILENAME*. Сгенерируется новый рандомный пароль (рис. 17)

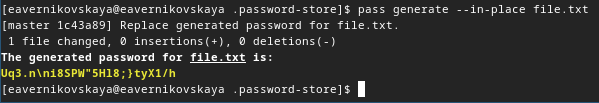


Рис. 17: Генерируем новый пароль

## 3.2 Управление файлами конфигурации

### 3.2.1 Дополнительное программное обеспечение

Утанавливаем дополнительное программное обеспечение (рис. 18), (рис. 19), (рис. 20), (рис. 21), (рис. 22), (рис. 23), (рис. 24), (рис. 25), (рис. 26), (рис. 27), (рис. 28), (рис. 29)

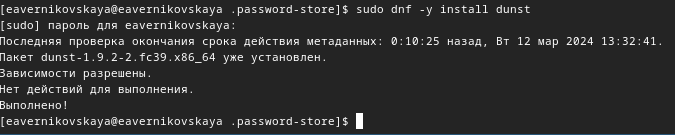


Рис. 18: Установка dunst

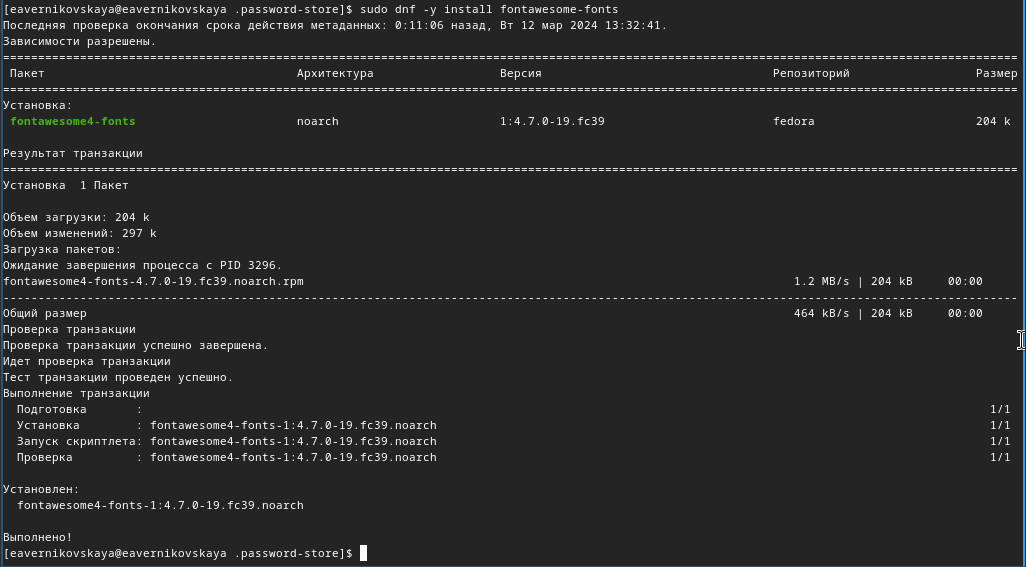


Рис. 19: Установка fontawesome-fonts

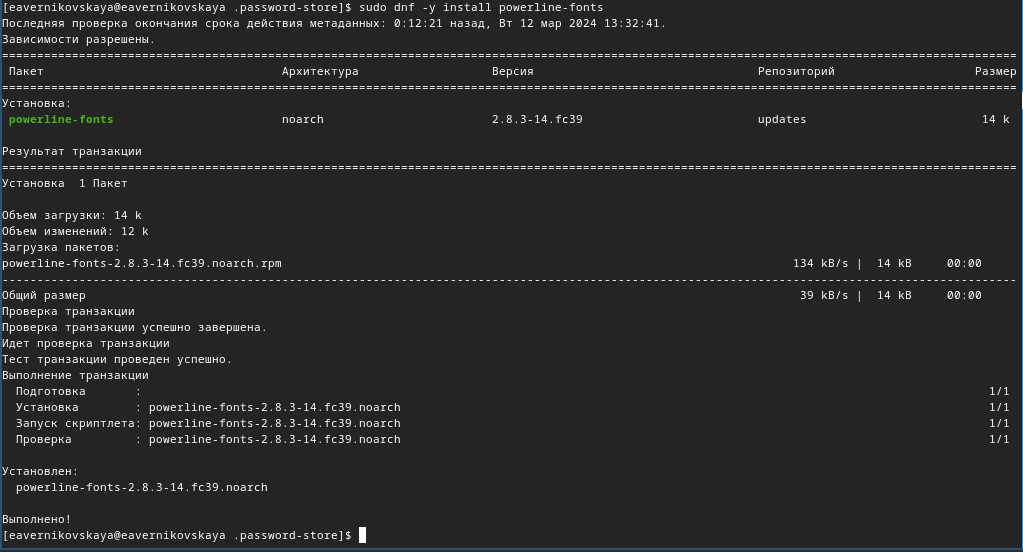


Рис. 20: Установка powerline-fonts

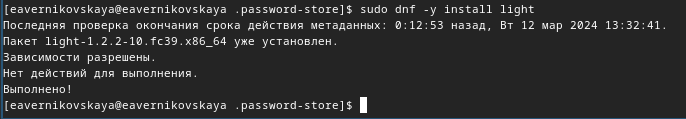


Рис. 21: Установка light

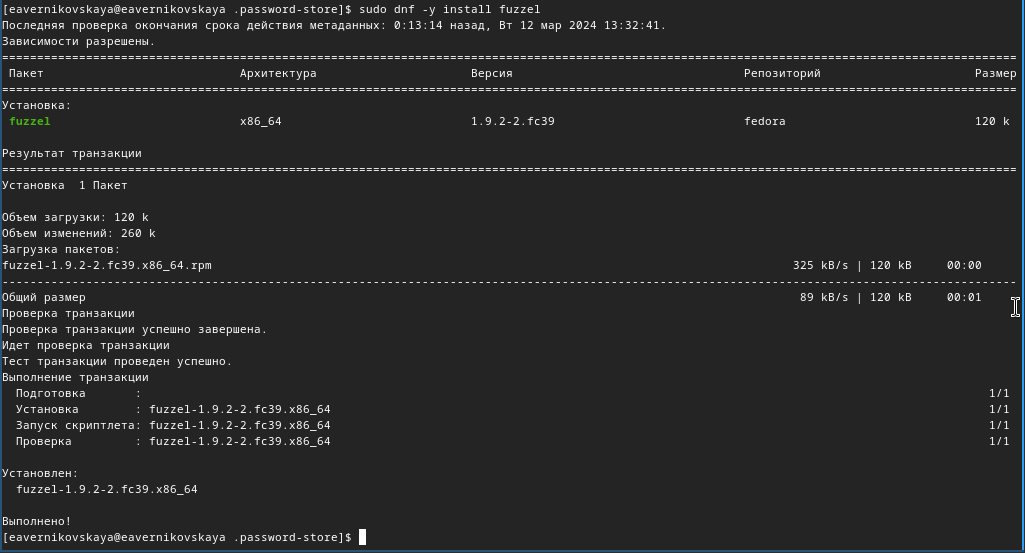


Рис. 22: Установка fuzzel

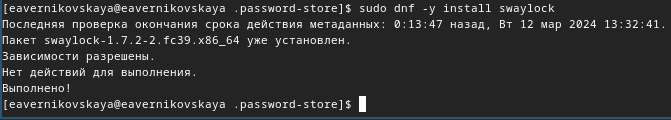


Рис. 23: Установка swaylock

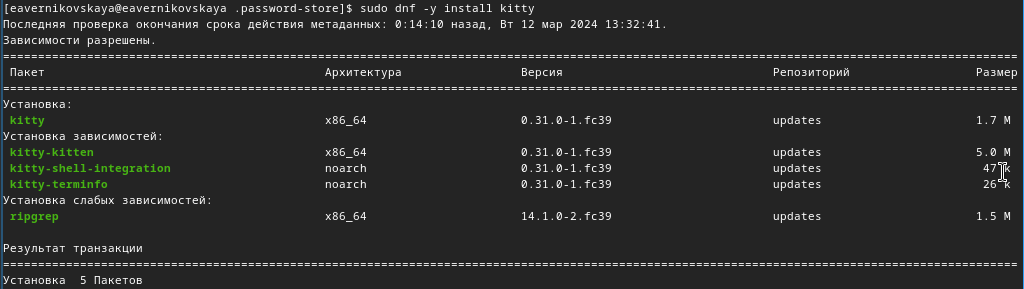


Рис. 24: Установка kitty

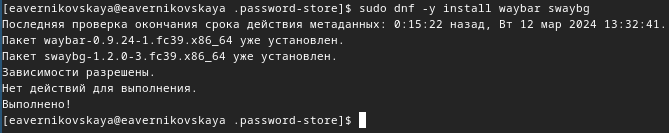


Рис. 25: Установка waybar swaybg

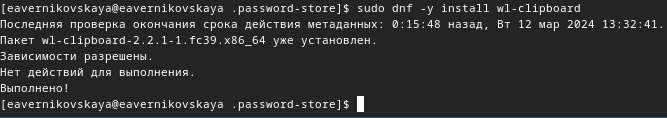


Рис. 26: Установка wl-clipboard

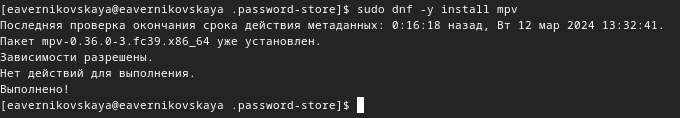


Рис. 27: Установка mpv

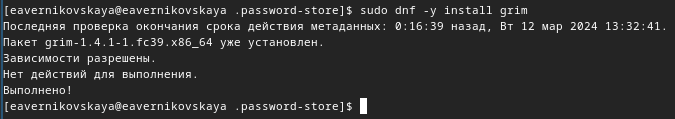


Рис. 28: Установка grim

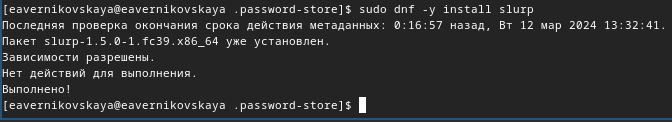


Рис. 29: Установка slurp

Далее устанавливаем шрифты. Сначала подключаем репозиторий, потом ищем нужное название и устанавливаем нужные шрифты (рис. 30), (рис. 31), (рис. 32)

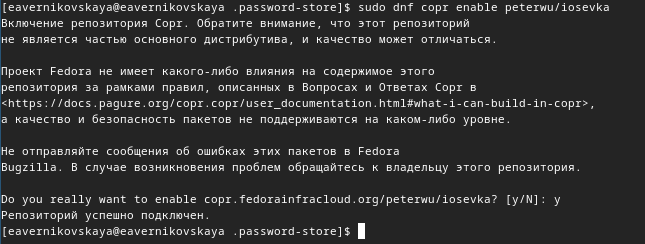


Рис. 30: Подключение репозитория

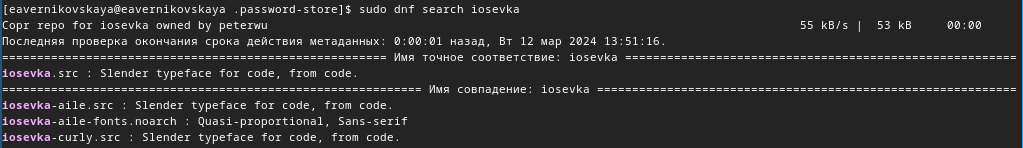


Рис. 31: Поиск iosevka

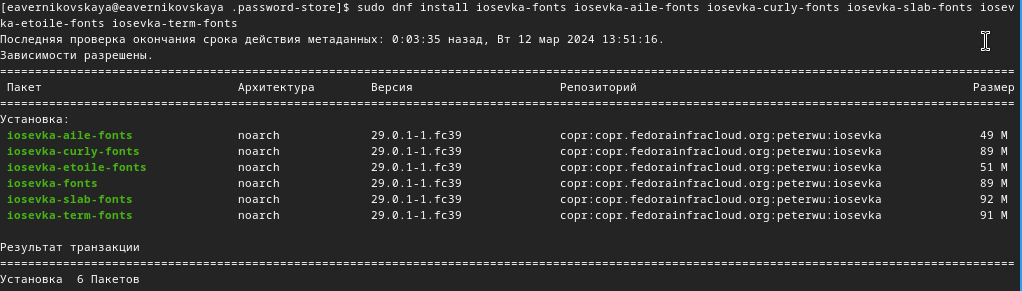


Рис. 32: Установка шрифтов

### 3.2.2 Установка

Устанавливаем chezmoi с помощью команды *sh -c “$(wget -qO- chezmoi.io/get)”* (рис. 33)

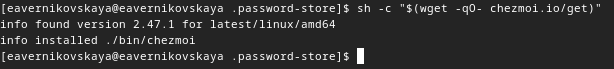


Рис. 33: Установка chezmoi

### 3.2.3 Создание собственного репозитория с помощью утилит

Создаём свой репозиторий для конфигурационных файлов на основе шаблона, с помощью *gh repo create* (рис. 34)

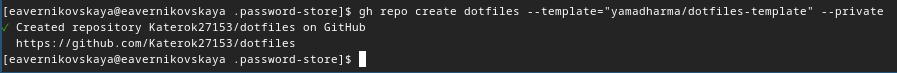


Рис. 34: Создание репозитория на основе шаблона

### 3.2.4 Подключение репозитория к своей системе

Переносим бинарный файл chezmoi в папку ~/usr/lacal/bin (рис. 35)

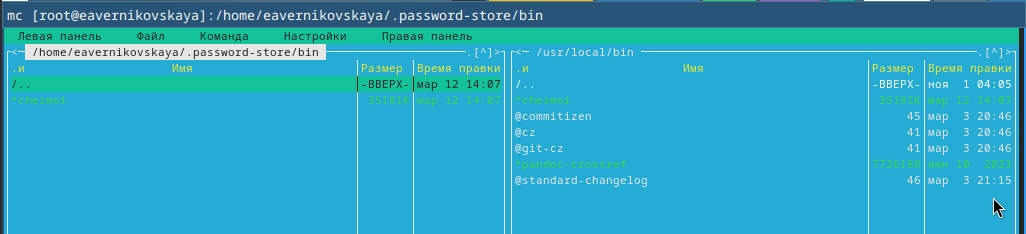


Рис. 35: Перенос файла (1)

Инициализируем chezmoi с нашим репозиторием dotfiles (рис. 36)

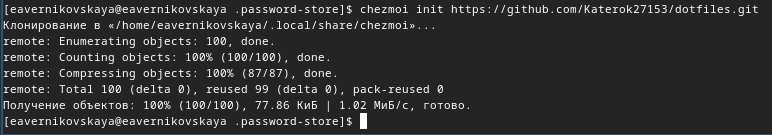


Рис. 36: Инициализация chezmoi с dotfiles (1)

Проверяем какие изменение внёс chezmoi в домашний каталог, введя *chezmoi diff* (рис. 37)

Ввод команды chezmoi diff (1)

Рис. 37: Ввод команды chezmoi diff (1)

Так нас устраивает большое количество внесённых изменений мы запускаем команду *chezmoi apply -v* (рис. 38)

Ввод команды chezmoi apply -v (1)

Рис. 38: Ввод команды chezmoi apply -v (1)

### 3.2.5 Использование chezmoi на нескольких машинах

В качестве второй виртуальной машины я выбрала kali. Сначала устанавливаем на kali chezmoi (рис. 39)

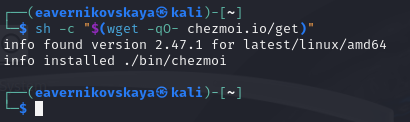


Рис. 39: Установка chezmoi на kali

Снова переносим бинарный файл chezmoi в папку ~/usr/lacal/bin (рис. 40)

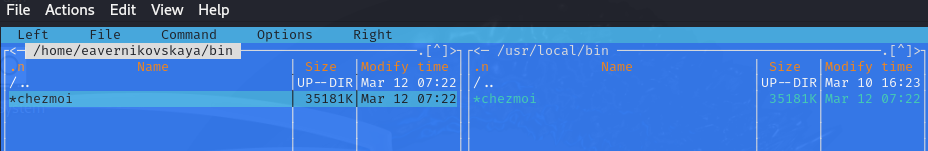


Рис. 40: Перенос файла (2)

На второй машине инициализируем chezmoi с нашим репозиторием dotfiles (рис. 41)

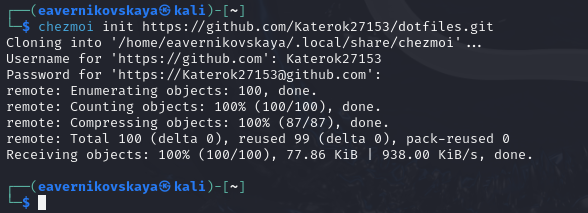


Рис. 41: Инициализация chezmoi с dotfiles (2)

Снова проверяем какие изменения внесёт chezmoi командой *chezmoi diff*, и если изменения устраивают вводим команду *chezmoi apply -v* (рис. 42), (рис. 43)

Инициализация chezmoi с dotfiles (2)

Рис. 42: Инициализация chezmoi с dotfiles (2)

Ввод команды chezmoi apply -v (2)

Рис. 43: Ввод команды chezmoi apply -v (2)

Выполняем команду *chezmoi update -v* - получаем и применяем последние изменения из нашего репозитория (рис. 44)

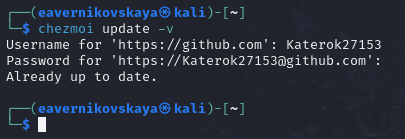


Рис. 44: Ввод команды chezmoi update -v

### 3.2.6 Настройка новой машины с помощью одной команды

Настраиваем новую машину с помощью одной команды *chezmoi init –apply https://github.com//dotfiles.git* (рис. 45)



Рис. 45: Настройка с помощью одной команды

### 3.2.7 Ежедневные операции c chezmoi

Извлекаем изменения из репозитория и применяем их одной командой *chezmoi update* (рис. 46)

Ввод команды chezmoi update

Рис. 46: Ввод команды chezmoi update

Далее выполняем *chezmoi git pull – –autostash –rebase && chezmoi diff* (рис. 47)

Извлечение последних изменений

Рис. 47: Извлечение последних изменений

Так как мы довольны изменениями то применяем их введя команду *chezmoi apply* (рис. 48)

Применение изменений

Рис. 48: Применение изменений

Включем функцию автоматического фиксирования и отправки изменений в репозиторий добавив в файл конфигурации ~/.config/chezmoi/chezmoi.toml следующее: - [git] autoCommit = true autoPush = true (рис. 49)

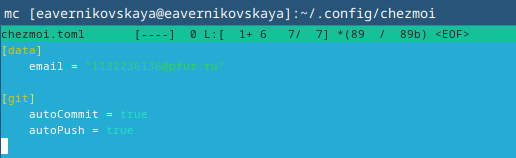


Рис. 49: Редоктирование файла

# 4 Выводы

В ходе выполнения лабораторной рбаоты мы настроили рабочую среду и получили навыки работы с менеджером паролей pass и с chezmoi

# 5 Список литературы

1. Лаборатораня работа №5 [Электронный ресурс] URL: https://esystem.rudn.ru/mod/page/view.php?id=1098939#org2695679
2. Плагин для Firefox[Электронный ресурс] URL: https://addons.mozilla.org/en-US/firefox/addon/browserpass-ce/.