Отчет по лабораторной работе №5

Настройка рабочей среды.

Дагделен Зейнап Реджеповна

Содержание

# 1 Цель работы

Научиться настраивать рабочую среду

# 2 Задание

1. Менеджер паролей pass -Установка -Настройка -Настройка интерфейса с броузером -Сохранение пароля
2. Управление файлами конфигурации
3. Дополнительное программное обеспечение -Установка -Создание собственного репозитория с помощью утилит -Подключение репозитория к своей системе -Использование chezmoi на нескольких машинах -Настройка новой машины с помощью одной команды -Ежедневные операции c chezmoi

# 3 Теоретическое введение

## 3.1 Менеджер паролей pass

* Менеджер паролей pass — программа, сделанная в рамках идеологии Unix.
* Также носит название стандартного менеджера паролей для Unix (The standard Unix password manager).

## 3.2 Основные свойства

* Данные хранятся в файловой системе в виде каталогов и файлов.
* Файлы шифруются с помощью GPG-ключа.

## 3.3 Структура базы паролей

* Структура базы может быть произвольной, если Вы собираетесь использовать её напрямую, без промежуточного программного обеспечения. Тогда семантику структуры базы данных Вы держите в своей голове.
* Если же необходимо использовать дополнительное программное обеспечение, необходимо семантику заложить в структуру базы паролей.

## 3.4 Семантическая структура базы паролей

Рассмотрим пользователя user в домене example.com, порт 22. Отсутствие имени пользователя или порта в имени файла означает, что любое имя пользователя и порт будут совпадать:

example.com.pgp

Соответствующее имя пользователя может быть именем файла внутри каталога, имя которого совпадает с хостом. Это полезно, если в базе есть пароли для нескольких пользователей на одном хосте:

example.com/user.pgp

Имя пользователя также может быть записано в виде префикса, отделенного от хоста знаком @:

user@example.com.pgp

Соответствующий порт может быть указан после хоста, отделённый двоеточием (:):

example.com:22.pgp example.com:22/user.pgp user@example.com:22.pgp

Эти все записи могут быть расположены в произвольных каталогах, задающих Вашу собственную иерархию.

# 4 Выполнение лабораторной работы

## 4.1 Установка

Установлю pass и gopass на Ubuntu (рис. 1 - рис. 2).

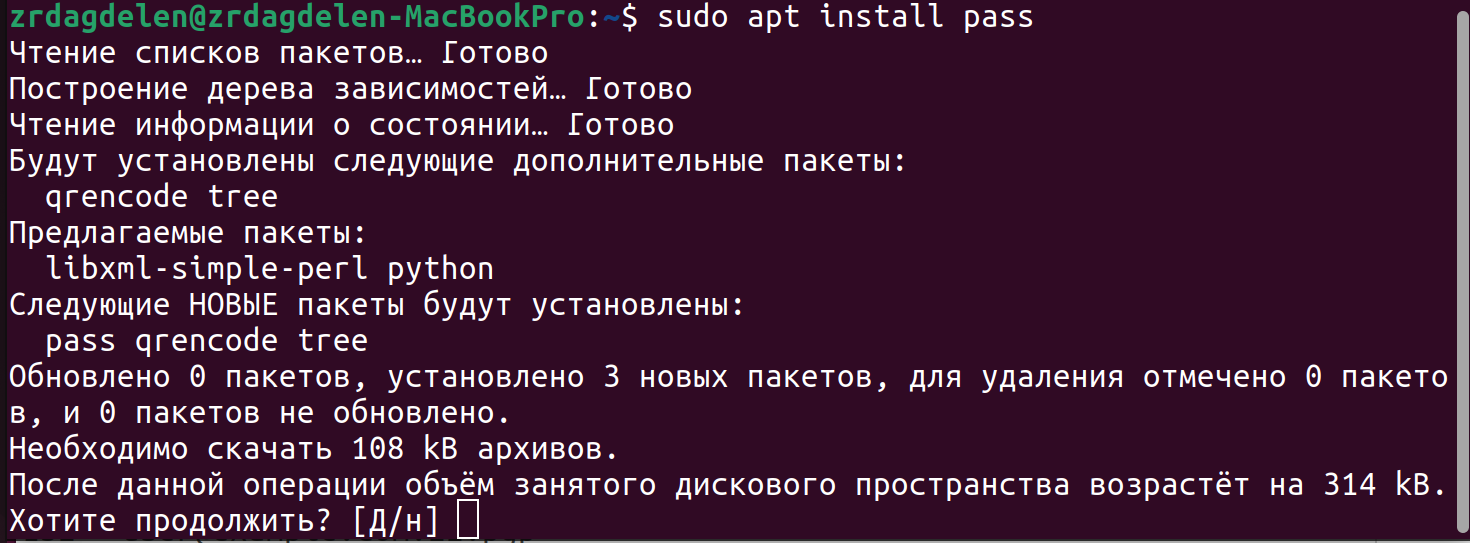


Рис. 1: Установка pass

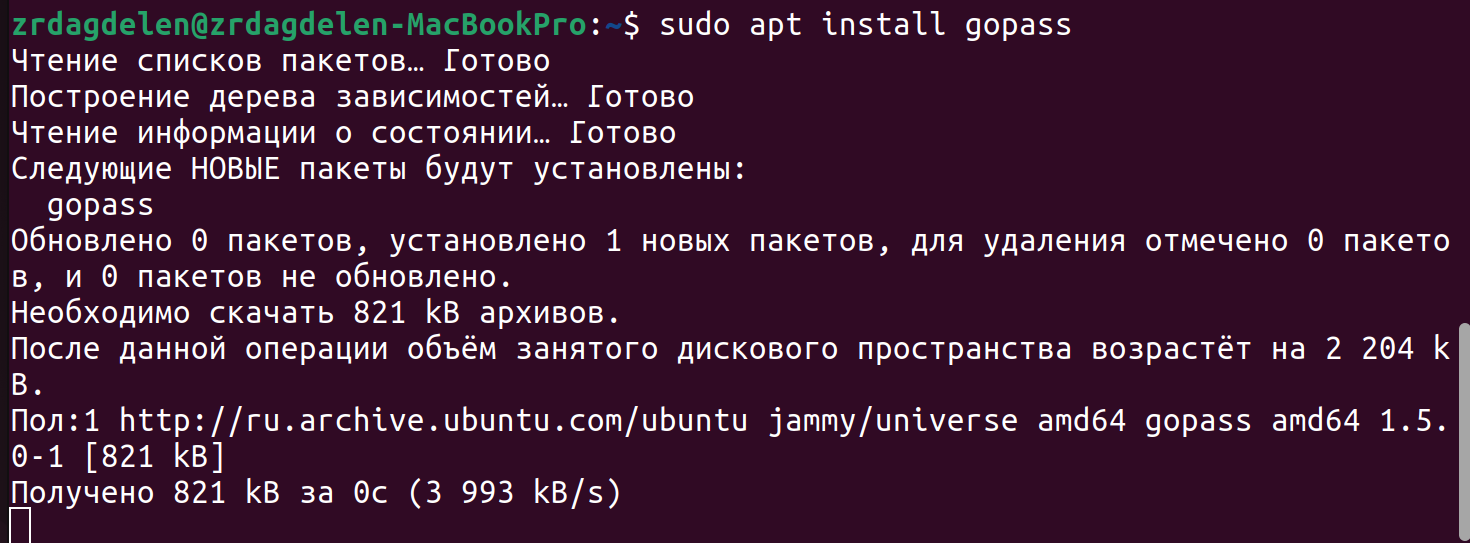


Рис. 2: Установка gopass

## 4.2 Настройка

Просмотрю список ключей GPG (gpg –list-secret-keys) и инициализируем хранилище (рис. 3), перейду в каталог work с помощью сd.

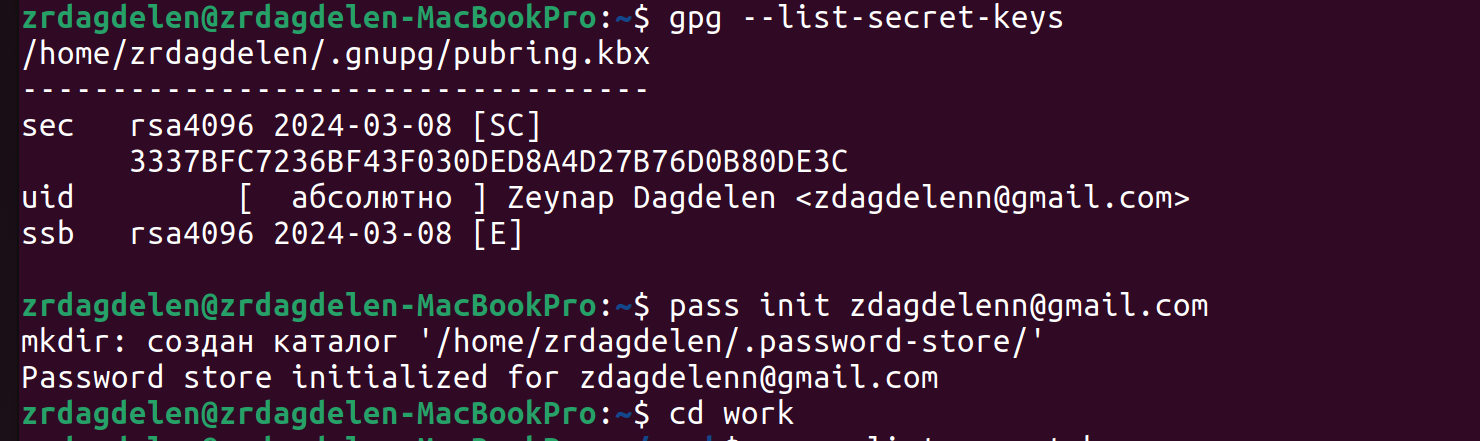


Рис. 3: Просмотр списка ключей и инициализация хранилища

Синхронизация с git. Создам структуру git (pass git init)(рис. 4)

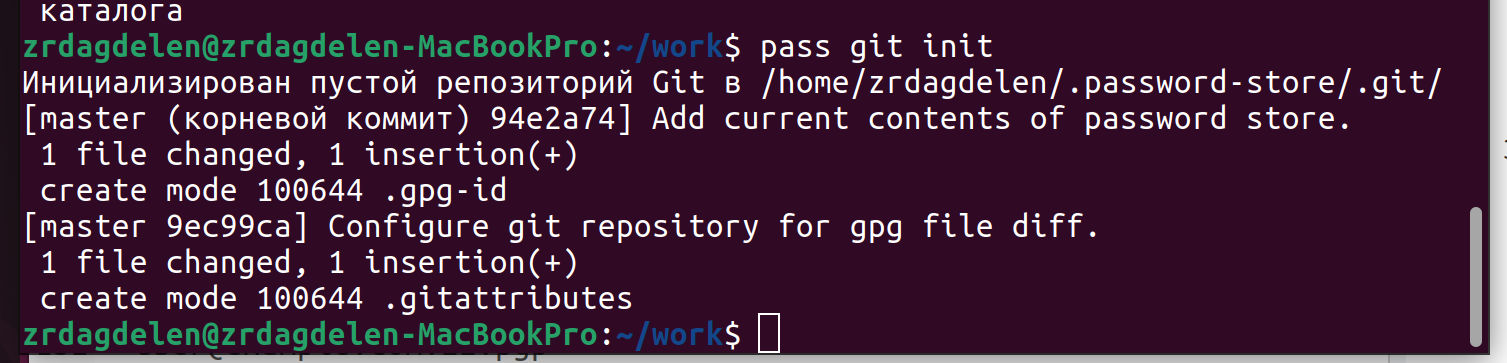


Рис. 4: Создание структуры git

Также можно задать адрес репозитория на хостинге (репозиторий необходимо предварительно создам)(рис. 5 - рис. 6).

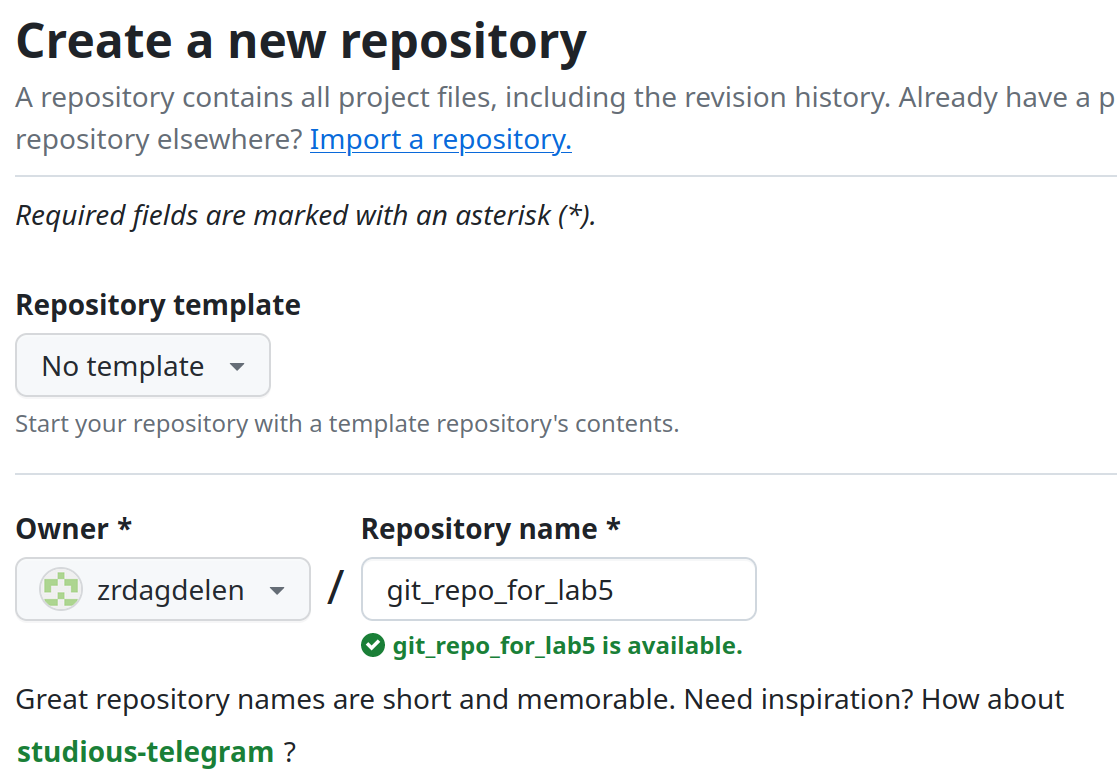


Рис. 5: Создание репозитория

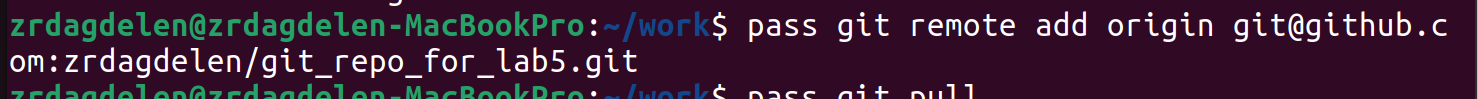


Рис. 6: Задание адреса репозитория на хостинге

Для синхронизации выполняю следующую команду

pass git pull  
pass git push

Прямые изменения

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

Если изменения сделаны непосредственно на файловой системе, необходимо вручную закоммитить и выложить изменения:

cd ~/.password-store/  
 git add .  
 git commit -am 'edit manually'  
 git push

Проверить статус синхронизации командой

pass git status

Настройка интерфейса с броузером

Для взаимодействия с броузером используется интерфейс native messaging.  
Поэтому кроме плагина к броузеру устанавливается программа, обеспечивающая интерфейс native messaging.  
  
Плагин browserpass  
 Репозиторий: https://github.com/browserpass/browserpass-extension  
 Плагин для брoузера  
 Плагин для Firefox: https://addons.mozilla.org/en-US/firefox/addon/browserpass-ce/.  
 Плагин для Chrome/Chromium: https://chrome.google.com/webstore/detail/browserpass-ce/naepdomgkenhinolocfifgehidddafch.  
  
 Интерфейс для взаимодействия с броузером (native messaging)  
 Репозиторий: https://github.com/browserpass/browserpass-native  
  
 Gentoo:  
  
 emerge www-plugins/browserpass  
  
 Fedora  
  
 dnf copr enable maximbaz/browserpass  
 dnf install browserpass

Сохранение пароля

Добавить новый пароль  
  
 Выполните:  
  
 pass insert [OPTIONAL DIR]/[FILENAME]  
  
 OPTIONAL DIR: необязательное имя каталога, определяющее файловую структуру для вашего хранилища паролей;  
 FILENAME: имя файла, который будет использоваться для хранения пароля.  
  
 Отобразите пароль для указанного имени файла:  
  
 pass [OPTIONAL DIR]/[FILENAME]  
  
 Замените существующий пароль:  
  
 pass generate --in-place FILENAME

Управление файлами конфигурации

Дополнительное программное обеспечение

Установите дополнительное программное обеспечение:  
  
sudo dnf -y install \  
 dunst \  
 fontawesome-fonts \  
 powerline-fonts \  
 light \  
 fuzzel \  
 swaylock \  
 kitty \  
 waybar swaybg \  
 wl-clipboard \  
 mpv \  
 grim \  
 slurp  
  
Установите шрифты:  
  
sudo dnf copr enable peterwu/iosevka  
sudo dnf search iosevka  
sudo dnf install iosevka-fonts iosevka-aile-fonts iosevka-curly-fonts iosevka-slab-fonts iosevka-etoile-fonts iosevka-term-fonts

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)"

Создание собственного репозитория с помощью утилит

Будем использовать утилиты командной строки для работы с github.  
  
Создадим свой репозиторий для конфигурационных файлов на основе шаблона:  
  
gh repo create dotfiles --template="yamadharma/dotfiles-template" --private

Подключение репозитория к своей системе

Инициализируйте chezmoi с вашим репозиторием dotfiles:  
  
chezmoi init git@github.com:<username>/dotfiles.git  
  
Проверьте, какие изменения внесёт chezmoi в домашний каталог, запустив:  
  
chezmoi diff  
  
Если вас устраивают изменения, внесённые chezmoi, запустите:  
  
chezmoi apply -v

Использование chezmoi на нескольких машинах

На второй машине инициализируйте chezmoi с вашим репозиторием dotfiles:  
  
chezmoi init https://github.com/<username>/dotfiles.git  
  
Или через ssh:  
  
chezmoi init git@github.com:<username>/dotfiles.git  
  
Проверьте, какие изменения внесёт chezmoi в домашний каталог, запустив:  
  
chezmoi diff  
  
Если вас устраивают изменения, внесённые chezmoi, запустите:  
  
chezmoi apply -v  
  
Если вас не устраивают изменения в файле, отредактируйте его с помощью:  
  
chezmoi edit file\_name  
  
Также можно вызвать инструмент слияния, чтобы объединить изменения между текущим содержимым файла, файлом в вашей рабочей копии и измененным содержимым файла:  
  
chezmoi merge file\_name  
  
При существующем каталоге chezmoi можно получить и применить последние изменения из вашего репозитория:  
  
chezmoi update -v

Настройка новой машины с помощью одной команды

Можно установить свои dotfiles на новый компьютер с помощью одной команды:  
  
chezmoi init --apply https://github.com/<username>/dotfiles.git  
  
Через ssh:  
  
chezmoi init --apply git@github.com:<username>/dotfiles.git

Ежедневные операции c chezmoi

Извлеките последние изменения из репозитория и примените их  
  
 Можно извлечь изменения из репозитория и применить их одной командой:  
  
 chezmoi update  
  
 Это запускается git pull --autostash --rebase в вашем исходном каталоге, а затем chezmoi apply.  
  
Извлеките последние изменения из своего репозитория и посмотрите, что изменится, фактически не применяя изменения  
  
 Выполните:  
  
 chezmoi git pull -- --autostash --rebase && chezmoi diff  
  
 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:  
  
 chezmoi apply  
  
Автоматически фиксируйте и отправляйте изменения в репозиторий  
 Можно автоматически фиксировать и отправлять изменения в исходный каталог в репозиторий.  
 Эта функция отключена по умолчанию.  
  
 Чтобы включить её, добавьте в файл конфигурации ~/.config/chezmoi/chezmoi.toml следующее:  
  
 [git]  
 autoCommit = true  
 autoPush = true  
  
 Всякий раз, когда в исходный каталог вносятся изменения, chezmoi фиксирует изменения с помощью автоматически сгенерированного сообщения фиксации и отправляет их в ваш репозиторий.  
 Будьте осторожны при использовании autoPush. Если ваш репозиторий dotfiles является общедоступным, и вы случайно добавили секрет в виде обычного текста, этот секрет будет отправлен в ваш общедоступный репозиторий.

# 5 Выводы

Я настроила рабочую среду

# Список литературы

[Операционные системы](https://esystem.rudn.ru/mod/page/view.php?id=1098796)