

Отчет по лабораторной работе №5

Дисциплина: Операционные системы

Беспутин Глеб Антонович

Содержание

1	Цель работы	6
2	Задание	7
3	Теоретическое введение	8
4	Выполнение лабораторной работы	9
5	Выводы	17
	Список литературы	18

Список иллюстраций

4.1	9
4.2	9
4.3	9
4.4	9
4.5	10
4.6	10
4.7	10
4.8	10
4.9	10
4.10	11
4.11	11
4.12	11
4.13	11
4.14	11
4.15	12
4.16	12
4.17	12
4.18	12
4.19	12
4.20	13
4.21	13
4.22	13
4.23	13
4.24	13
4.25	13
4.26	14
4.27	14
4.28	14
4.29	14
4.30	14
4.31	15
4.32	15
4.33	15
4.34	15
4.35	15
4.36	16
4.37	16

4.38	16
4.39	16

Список таблиц

1 Цель работы

Научиться работать с менеджером паролей pass.

2 Задание

- 1) Установить и настроить менеджер паролей pass.
- 2) Научиться управлять файлами конфигурации

3 Теоретическое введение

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

4 Выполнение лабораторной работы

Установка(рис. [4.32]).

```
[glebbesputin@glebbesputin ~]$ dnf install pass pass-otp
```

Рис. 4.1:

```
[glebbesputin@glebbesputin ~]$ dnf install gopass
```

Рис. 4.2:

Просмотр списка ключей(рис. [4.3]).

```
Выполнено!  
[glebbesputin@glebbesputin ~]$ gpg --list-secret-keys
```

Рис. 4.3:

Инициализация хранилища(рис. [4.4]).

```
pass init glebb2005@mail.ru
```

Рис. 4.4:

Создадим структуру git(рис. [4.5]).

```
[glebbesputin@glebbesputin ~]$ pass git init
```

Рис. 4.5:

Также можно задать адрес репозитория на хостинге (репозиторий необходимо предварительно создать)(рис. [4.6]).

```
git remote add origin git@github.com:<git_username>/<git_repo>.git
```

Рис. 4.6:

Для синхронизации выполняется следующая команда(рис. [4.7]).

```
pass git pull
```

Рис. 4.7:

Если изменения сделаны непосредственно на файловой системе, необходимо вручную закоммитить и выложить изменения(рис. [4.8]).

```
cd ~/.password-store/
```

Рис. 4.8:

```
git add .
```

Рис. 4.9:

```
git add .  
git commit -am 'edit manually'
```

Рис. 4.10:

```
git push
```

Рис. 4.11:

Проверить статус синхронизации можно командой(рис. [4.12])

```
pass git status
```

Рис. 4.12:

Для взаимодействия с браузером используется интерфейс native messaging.(рис. [4.13])

```
[glebbesputin@glebbesputin .password-store]$ dnf copr enable maximbaz/browserpass  
ss
```

Рис. 4.13:

```
[glebbesputin@glebbesputin .password-store]$ dnf install browserpass
```

Рис. 4.14:

Добавить новый пароль(рис. [4.15])

```
pass insert [OPTIONAL DIR]/[FILENAME]
```

Рис. 4.15:

Отобразите пароль для указанного имени файла(рис. [4.16])

```
pass [OPTIONAL DIR]/[FILENAME]
```

Рис. 4.16:

Замените существующий пароль(рис. [4.17])

```
pass generate --in-place FILENAME
```

Рис. 4.17:

Установите дополнительное программное обеспечение(рис. [4.18])

```
[glebbesputin@glebbesputin .password-store]$ sudo dnf -y install \
dunst \
fontawesome-fonts \
powerline-fonts \
light \
fuzzel \
swaylock \
kitty \
waybar swaybg \
wl-clipboard \
mpv \
grim \
glances
```

Рис. 4.18:

Установите шрифты(рис. [4.19])

```
[glebbesputin@glebbesputin .password-store]$ sudo dnf copr enable peterwu/iosev
ka
```

Рис. 4.19:

```
[glebbesputin@glebbesputin .password-store]$ sudo dnf search iosevka
```

Рис. 4.20:

```
[glebbesputin@glebbesputin .password-store]$ sudo dnf install iosevka-fonts ios  
evka-aile-fonts iosevka-curly-fonts iosevka-slab-fonts iosevka-etoile-fonts ios  
evka-term-fonts
```

Рис. 4.21:

Установка бинарного файла. Скрипт определяет архитектуру процессора и операционную систему и скачивает необходимый файл(рис. [4.22])

```
[glebbesputin@glebbesputin .password-store]$ sh -c "$(wget -qO- chezmoi.io/get  
)"
```

Рис. 4.22:

Создадим свой репозиторий для конфигурационных файлов на основе шаблона:(рис. [4.23])

```
[glebbesputin@glebbesputin .password-store]$ gh repo create dotfiles --templat  
e="yamadharmadotfiles-template" --private
```

Рис. 4.23:

Инициализируйте chezmoi с вашим репозиторием dotfiles(рис. [4.24])

```
[glebbesputin@glebbesputin .password-store]$ chezmoi init git@github.com:<user  
name>/dotfiles.git
```

Рис. 4.24:

Проверьте, какие изменения внесёт chezmoi в домашний каталог, запустив(рис. [4.25])

```
[glebbesputin@glebbesputin .password-store]$ chezmoi diff
```

Рис. 4.25:

Если вас устраивают изменения, внесённые chezmoi, запустите(рис. [4.26])

```
[glebbesputin@glebbesputin .password-store]$ chezmoi apply -v
```

Рис. 4.26:

На второй машине инициализируйте chezmoi с вашим репозиторием dotfiles(рис. [4.27])

```
[glebbesputin@glebbesputin .password-store]$ chezmoi init https://github.com/<username>/dotfiles.git
```

Рис. 4.27:

Или через ssh(рис. [4.28])

```
[glebbesputin@glebbesputin .password-store]$ chezmoi init git@github.com:<username>/dotfiles.git
```

Рис. 4.28:

Проверьте, какие изменения внесёт chezmoi в домашний каталог, запустив(рис. [4.29])

```
bash: chezmoi: нет такого файла или каталога
[glebbesputin@glebbesputin .password-store]$ chezmoi diff
bash: chezmoi: команда не найдена
```

Рис. 4.29:

Если вас устраивают изменения, внесённые chezmoi, запустите(рис. [4.30])

```
[glebbesputin@glebbesputin .password-store]$ chezmoi apply -v
bash: chezmoi: команда не найдена...
```

Рис. 4.30:

Если вас не устраивают изменения в файле, отредактируйте его с помощью(рис. [4.31])

```
[glebbesputin@glebbesputin .password-store]$ chezmoi edit file_name
bash: chezmoi: команда не найдена...
```

Рис. 4.31:

Также можно вызвать инструмент слияния, чтобы объединить изменения между текущим содержимым файла, файлом в вашей рабочей копии и измененным содержимым файла(рис. [??])

```
[glebbesputin@glebbesputin .password-store]$ chezmoi merge file_name
```

Рис. 4.32:

При существующем каталоге chezmoi можно получить и применить последние изменения из вашего репозитория(рис. [4.33])

```
[glebbesputin@glebbesputin .password-store]$ chezmoi update -v
```

Рис. 4.33:

Можно установить свои dotfiles на новый компьютер с помощью одной команды(рис. [4.34])

```
[glebbesputin@glebbesputin .password-store]$ chezmoi init --apply https://github.com/<username>/dotfiles.git
```

Рис. 4.34:

Через ssh(рис. [4.35])

```
[glebbesputin@glebbesputin .password-store]$ chezmoi init --apply git@github.com:<username>/dotfiles.git
```

Рис. 4.35:

Можно извлечь изменения из репозитория и применить их одной командой(рис. [4.36])

```
[glebbesputin@glebbesputin .password-store]$ chezmoi update
bash: chezmoi: команда не найдена...
```

Рис. 4.36:

Извлеките последние изменения из своего репозитория и посмотрите, что изменится, фактически не применяя изменения (рис. [4.37])

```
[glebbesputin@glebbesputin .password-store]$ chezmoi git pull -- --autostash --rebase && chezmoi diff
```

Рис. 4.37:

Если вы довольны изменениями, вы можете применить их(рис. [4.38])

```
[glebbesputin@glebbesputin .password-store]$ chezmoi apply
```

Рис. 4.38:

Автоматически фиксируйте и отправляйте изменения в репозиторий(рис. [4.39])

```
[git]
autoCommit = true
autoPush = true
```

Рис. 4.39:

5 Выводы

Я научился работать с менеджером паролей pass.

Список литературы