

# Отчёт по лабораторной работе №5

## Операционные системы

Лисенков Е.Р.

### Содержание

Цель работы .....	1
Задание.....	1
Теоретическое введение .....	1
Выполнение лабораторной работы .....	1
Выводы .....	6

### Цель работы

Приобретение практических навыков взаимодействия пользователя с системой посредством командной строки.

### Задание

1. Установить необходимое программное обеспечение.
2. Изучить принцип работы менеджера паролей pass.
3. Установить дополнительное программное обеспечение.

### Теоретическое введение

Менеджер паролей pass создан в рамках идеологии Unix. В нём данные хранятся в файловой системе в виде каталогов и файлов. Файлы шифруются с помощью GPG-ключа. Структура базы паролей может быть произвольной. Если необходимо использовать дополнительное программное обеспечение, семантику необходимо заложить в структуру базы паролей. На данный момент существует 2 основных реализации: pass и gopass.

### Выполнение лабораторной работы

Выполним команду для просмотра списка ключей (рис.1).

```
erlisenkov@dkin22 ~ $ gpg --list-secret-keys
/afs/.dk.sci.pfu.edu.ru/home/e/r/erlisenkov/.gnupg/pubring.kbx
-----
sec   rsa2048 2023-10-12 [SCEA]
      47C3B92253CCBD48C23E566343F48217E8A497E3
uid   [ абсолютно ] erlisenkov (nkabd-03-23) <egorlis13@mail.ru>
ssb   rsa2048 2023-10-12 [SEA]

sec   rsa4096 2024-02-28 [SC]
      1D21AA6FDB473A9571234A1825ED28145B7677F8
uid   [ абсолютно ] Egorr <1132232881@rudn.ru>
ssb   rsa4096 2024-02-28 [E]

erlisenkov@dkin22 ~ $
```

*Просмотрим список ключей*

Инициализируем хранилище (рис.2).

```
erlisenkov@dkin22 ~ $ pass init 1D21AA6FDB473A9571234A1825ED28145B7677F8
mkdir: создан каталог '/afs/.dk.sci.pfu.edu.ru/home/e/r/erlisenkov/.password-store/'
Password store initialized for 1D21AA6FDB473A9571234A1825ED28145B7677F8
erlisenkov@dkin22 ~ $
```

*Инициализируем хранилище*

Синхронизация с git (рис.3 рис.4)

Создадим структуру git:

1. pass git init

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

2. pass git remote add origin git@githubcom:<git\_username>/<git\_repo>.git

Для синхронизации выполняется следующая команда:

3. pass git pull

4. pass git push

```
erlisenkov@dk1n22 ~ $ pass git init
Инициализирован пустой репозиторий 6
[master (корневой коммит) 0a8bc3e] A
1 file changed, 1 insertion(+)
create mode 100644 .pgp-id
[master d80778d] Configure git repos
1 file changed, 1 insertion(+)
create mode 100644 .gitattributes
erlisenkov@dk1n22 ~ $ pass git remot
erlisenkov@dk1n22 ~ $ pass git pull
```

*Синхронизация с git*

```
erlisenkov@dk1n22 ~/work/study/2022-2023/Операционные системы/os-intro $ pass git push
```

*Синхронизация с git*

Скачаем интерфейс для взаимодействия с браузером (native messaging) (рис.5).

Репозиторий: <https://github.com/browserpass/browserpass-native>

Gentoo:

emerge www-plugins/browserpass

```
erlisenkov@dk1n22 ~ $ emerge www-plugins/browserpass
```

*Синхронизация с git*

Сохранение пароля (рис.6 рис.8)

1.Добавим новый пароль:

pass insert [OPTIONAL DIR]/[FILENAME]

2.Отобразите пароль для указанного имени файла:

pass [OPTIONAL DIR]/[FILENAME]

3.Замените существующий пароль:

pass generate --in-place FILENAME

```

erlisenkov@dkin22 ~ $ pass insert pass/password
mkdir: создан каталог '/afs/.dk.sci.pfu.edu.ru/home/erlisenkov/pass'
Enter password for pass/password:
Retype password for pass/password:
[master 7430a0e] Add given password for pass/password
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 pass/password.gpg
erlisenkov@dkin22 ~ $ pass pass/password
12345

```

*Сохранение пароля*

```

erlisenkov@dkin22 ~ $ pass generate --in-place password

```

*Сохранение пароля*

Установим бинарный файл. Скрипт определяет архитектуру процессора и операционную систему и скачивает необходимый файл (рис.8).

```

erlisenkov@dkin22 ~ $ sh -c "$(wget -qO- chezmoi.io/get)"
info found version 2.47.1 for latest/linux/amd64

```

*Установим бинарный файл*

Установим соединение с git (рис.9)

```

erlisenkov@dkin22 ~/work $ gh auth login
? What account do you want to log into? GitHub.com
? What is your preferred protocol for Git operations on this host? HTTPS
? Authenticate Git with your GitHub credentials? Yes
? How would you like to authenticate GitHub CLI? Login with a web browser

! First copy your one-time code: E988-8F2E
Press Enter to open github.com in your browser...
kdeinit5: preparing to launch '/usr/lib64/libexec/kf5/klauncher'
kdeinit5: Launched KLauncher, pid = 5281, result = 0
Connecting to deprecated signal QDBusConnectionInterface::serviceOwnerChanged(QString,QString,QStri
kdeinit5: opened connection to :0
kdeinit5: Got SHELL 'konqueror' from wrapper.
kdeinit5: preparing to launch '/usr/bin/konqueror'
Launched ok, pid = 5288
KTranscript: Loaded module: /usr/share/locale/ru/LC_SCRIPTS/ki18n5/ki18n5.js

```

*Установим соединение с git*

Создам собственный репозиторий с помощью утилит

Буду использовать утилиты командной строки для работы с github.

Создам свой репозиторий для конфигурационных файлов на основе шаблона:

```
gh repo create dotfiles --template="yamadharma/dotfiles-template" --private
```

Подключу репозитория к своей системе

Инициализирую chezmoi с вашим репозиторием dotfiles:

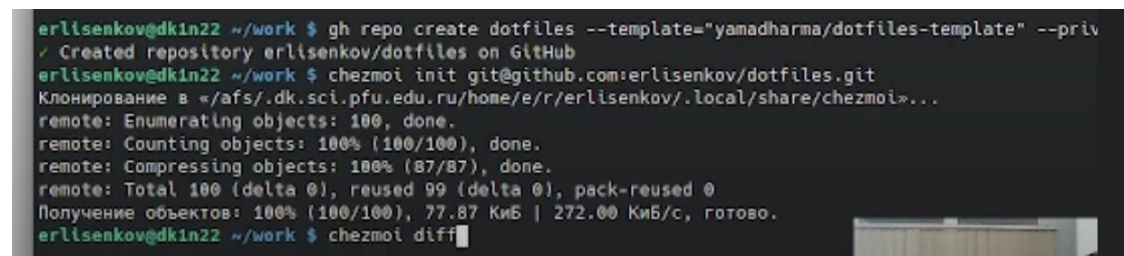
```
chezmoi init git@github.com:<username>/dotfiles.git
```

Проверю, какие изменения внесёт chezmoi в домашний каталог, запустив:

```
chezmoi diff
```

Если устраит изменения, внесённые chezmoi, запустите:

```
chezmoi apply -v
```

A terminal window showing the process of initializing chezmoi. The user runs 'gh repo create dotfiles --template="yamadharma/dotfiles-template" --priv' to create a repository on GitHub. Then, they run 'chezmoi init git@github.com:erlisenkov/dotfiles.git' to initialize chezmoi with the repository. The terminal output shows progress bars for cloning, enumerating, counting, and compressing objects, and a final message indicating the repository is ready. The user then runs 'chezmoi diff' to check for changes.

```
erlisenkov@dkin22 ~/work $ gh repo create dotfiles --template="yamadharma/dotfiles-template" --priv
✓ Created repository erlisenkov/dotfiles on GitHub
erlisenkov@dkin22 ~/work $ chezmoi init git@github.com:erlisenkov/dotfiles.git
Клонирование в «/afs/dk.sci.pfu.edu.ru/home/e/r/erlisenkov/.local/share/chezmoi»...
remote: Enumerating objects: 100, done.
remote: Counting objects: 100% (100/100), done.
remote: Compressing objects: 100% (87/87), done.
remote: Total 100 (delta 0), reused 99 (delta 0), pack-reused 0
Получение объектов: 100% (100/100), 77.87 КиБ | 272.00 КиБ/с, готово.
erlisenkov@dkin22 ~/work $ chezmoi diff
```

*Создам собственный репозиторий*

Ежедневные операции с chezmoi

Извлеку 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.

Извлеку последние изменения из своего репозитория и посмотрите, что изменится, фактически не применяя изменения

Выполню:

```
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.

применяем их:

```
chezmoi apply
```

```

erlisenkov@dkin22 ~ $ chezmoi update
Уже актуально.
erlisenkov@dkin22 ~ $ chezmoi git pull -- --autostash --rebase && chezmoi diff
Уже актуально.
erlisenkov@dkin22 ~ $ chezmoi apply
erlisenkov@dkin22 ~ $ ls
bin      GNUstep  public   tmp      Видео    Загрузки  Музыка    'Рабочий стол'
GNS3     LICENSE  public_html  work    Документы  Изображения  Общедоступные  Шаблоны
erlisenkov@dkin22 ~ $ mkdr ~/.config/chezmoi/chezmoi.toml

```

*Ежедневные операции с chezmoi*

```

erlisenkov@dkin22 ~ $ cd ~/.config/chezmoi
erlisenkov@dkin22 ~/.config/chezmoi $ mc

```

*Ежедневные операции с chezmoi*

## Выводы

В ходе лабораторной работы я освоила навыки использования менеджера паролей pass.