Шаблон отчёта по лабораторной работе

Простейший вариант

Хань Цзянтао

Содержание

# 1 Цель работы

Знаком Менеджер паролей pass

# 2 Задание

1. Менеджер паролей pass
2. Управление файлами конфигурации
3. Общая информация
4. Выполнение лабораторной работы

# 3 Теоретическое введение

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*).

Основные свойства

Данные хранятся в файловой системе в виде каталогов и файлов.

Файлы шифруются с помощью GPG-ключа.

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.

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 @:

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

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

# 4 Выполнение лабораторной работы

### 4.0.1 **4.1Менеджер паролей pass**

### 4.0.2 Установка

Fedora

pass

dnf install pass pass-otp

|  |
| --- |
| 01 |

01

gopass

dnf install gopass

|  |
| --- |
| 02 |

02

### 4.0.3 Настройка

1.Ключи GPG

​ Просмотр списка ключей:

gpg --list-secret-keys

|  |
| --- |
| 03 |

03

Если ключа нет, нужно создать новый:

gpg --full-generate-key

|  |
| --- |
| 04 |

04

2.Инициализация хранилища

​ Инициализируем хранилище:

pass init <gpg-id or email>

|  |
| --- |
| 05 |

05

3.Синхронизация с git

​ Создадим структуру git:

pass git init

|  |
| --- |
| 06 |

06

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

pass git remote add origin git@github.com:<git\_username>/<git\_repo>.git

|  |
| --- |
| 07 |

07

Для синхронизации выполняется следующая команда:

pass git pull  
pass git push

​ 1.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

|  |
| --- |
| 08 |

08

Проверить статус синхронизации модно командой

pass git status

|  |
| --- |
| 09 |

09

### 4.0.4 4.2Настройка интерфейса с броузером

* Для взаимодействия с броузером используется интерфейс *native messaging*.
* Поэтому кроме плагина к броузеру устанавливается программа, обеспечивающая интерфейс *native messaging*.

1. Плагин [browserpass](https://github.com/browserpass/browserpass-extension)
   * 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

|  |
| --- |
| 10 |

10

|  |
| --- |
| 11 |

11

### 4.0.5 4.3Сохранение пароля

​ 1.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.

|  |
| --- |
| 12 |

12

​ Отобразите пароль для указанного имени файла:

pass [OPTIONAL DIR]/[FILENAME]

|  |
| --- |
| 13 |

13

​ Замените существующий пароль:

pass generate --in-place FILENAME

### 4.0.6 4.4Управление файлами конфигурации

#### 4.0.6.1 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

|  |
| --- |
| 14 |

14

​ 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

# 5 Выводы

Благодаря данной лабораторной работе я приобрела навыки работы в Midnight

Commander и освоила инструкции языка ассемблер mov и int, что поможет мне

при выполнении последующих лабораторных работ.

|  |
| --- |
| 17 |

17

### 5.0.1 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)"

|  |
| --- |
| 18 |

18

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

|  |
| --- |
| 19 |

19

Подключение репозитория к своей системе Инициализируйте chezmoi с вашим репозиторием dotfiles:

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

|  |
| --- |
| 20 |

20

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

|  |
| --- |
| 21 |

21

Если вас устраивают изменения, внесённые chezmoi, запустите:

chezmoi apply -v

|  |
| --- |
| 22 |

22

#### 5.0.1.1 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

### 5.0.2 4.5Настройка 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

### 5.0.3 Ежедневные операции c chezmoi

1. Извлеките последние изменения из репозитория и примените их
   * Можно извлечь изменения из репозитория и применить их одной командой:
   * chezmoi update

|  |
| --- |
| 23 |

23

# Список литературы

::: {#refs}

1. GDB: The GNU Project Debugger. — URL: https://www.gnu.org/software/gdb/.
2. GNU Bash Manual. — 2016. — URL: https://www.gnu.org/software/bash/manual/.
3. Midnight Commander Development Center. — 2021. — URL: https://midnight

commander. org/.

1. NASM Assembly Language Tutorials. — 2021. — URL: https://asmtutor.com/.
2. Newham C. Learning the bash Shell: Unix Shell Programming. — O’Reilly

Media, 2005. — 354 с. — (In a Nutshell). — ISBN 0596009658. — URL:

http://www.amazon.com/Learningbash-Shell-Programming-Nutshell/dp/0596009658.

1. Robbins A. Bash Pocket Reference. — O’Reilly Media, 2016. — 156 с. — ISBN

978-1491941591.

1. The NASM documentation. — 2021. — URL: https://www.nasm.us/docs.php.
2. Zarrelli G. Mastering Bash. — Packt Publishing, 2017. — 502 с. — ISBN
3. Колдаев В. Д., Лупин С. А. Архитектура ЭВМ. — М. : Форум, 2018.
4. Куляс О. Л., Никитин К. А. Курс программирования на ASSEMBLER. — М. :

Солон-Пресс, 2017.

1. Новожилов О. П. Архитектура ЭВМ и систем. — М. : Юрайт, 2016.
2. Расширенный ассемблер: NASM.— 2021.—URL: https://www.opennet.ru/docs/RUS/nasm/.
3. Робачевский А., Немнюгин С., Стесик О. Операционная система UNIX. — 2-е

изд. — БХВПетербург, 2010. — 656 с. — ISBN 978-5-94157-538-1.

1. Столяров А. Программирование на языке ассемблера NASM для ОС Unix.— 2-

е изд.—М. : МАКС Пресс, 2011.—URL: http://www.stolyarov.info/books/asm\_unix.:::