Менеджер паролей

‘hush’

Подготовили студенты группы Б9123-01.03.02:

- Куторгин Руслан Алексеевич
- Соколовский Роман Вадимович

г. Владивосток
2026

$ cat проблема.txt

[✗] Современные менеджеры паролей перегружены функциями

[✗] Облачная синхронизация = потенциальные риски безопасности

[✗] Слишком много зависимостей и сложности

[✓] Нужно простое, локальное и безопасное решение

> основные_функции

- Зашифрованная локальная база данных (приватные файлы)
- Аутентификация аппаратным ключом (блокировка USB-устройства)
- Автоочистка буфера обмена (таймаут 10 секунд)
- Индикатор надежности пароля
- Избранные записи со звездочкой
- Функции поиска и фильтрации

интерфейс.fltk

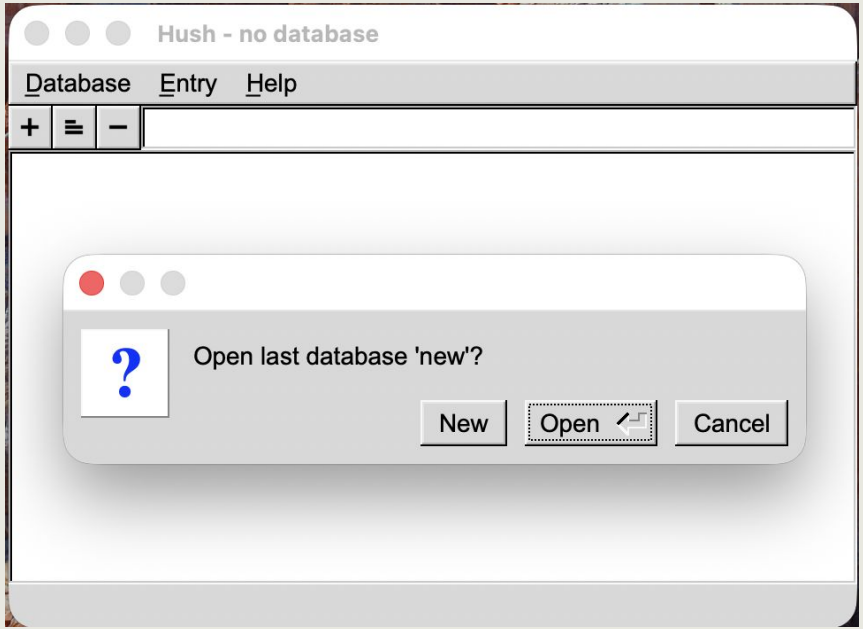
- Слева: список паролей с записями через моноширинный пробел
- Строка меню: ⌘+N ⌘+O ⌘+S
- Справа: панель редактирования с полями ввода
- Строка состояния: обратный отсчет таймера буфера

// база_данных.cpp

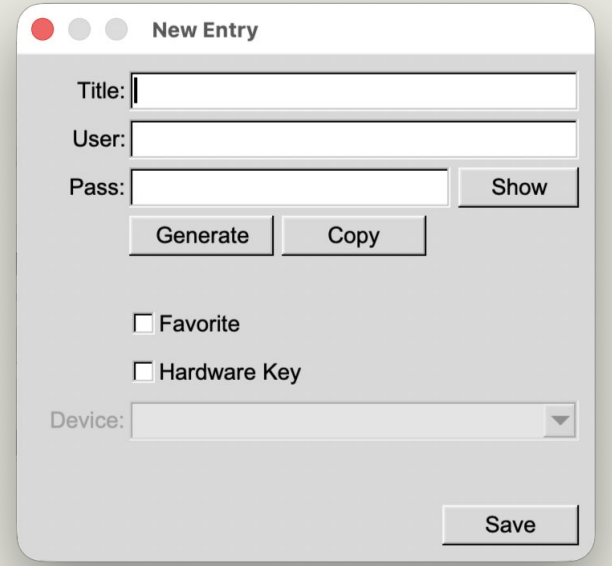
База данных Hush — это зашифрованный файл на вашем компьютере.
Никакого облака, никакой синхронизации через интернет.

Вы создаете мастер-пароль, который генерирует ключ шифрования.
Файл начинается с заголовка "HUSH", затем соль, затем зашифрованные данные.
Трехпроходное XOR-шифрование без внешних
библиотек — простая, понятная криптография.

Вы полностью контролируете свои данные.



$ создать_секрет.cpp



Создание записи в Hush предельно простое.

Введите название сайта, логин и пароль. Приложение оценит надежность пароля: красный — слабый, желтый — средний, зеленый — сильный.

После сохранения можно пометить запись звездочкой для быстрого доступа. При копировании пароля буфер обмена автоматически очищается через 10 секунд.

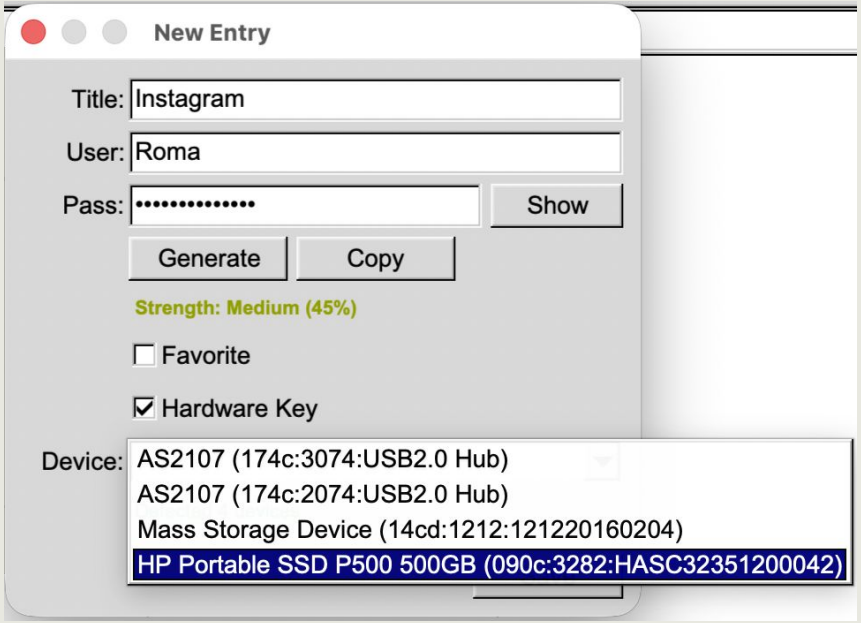
аппаратный_ключ.h

Физический ключ — дополнительная защита для критических паролей.

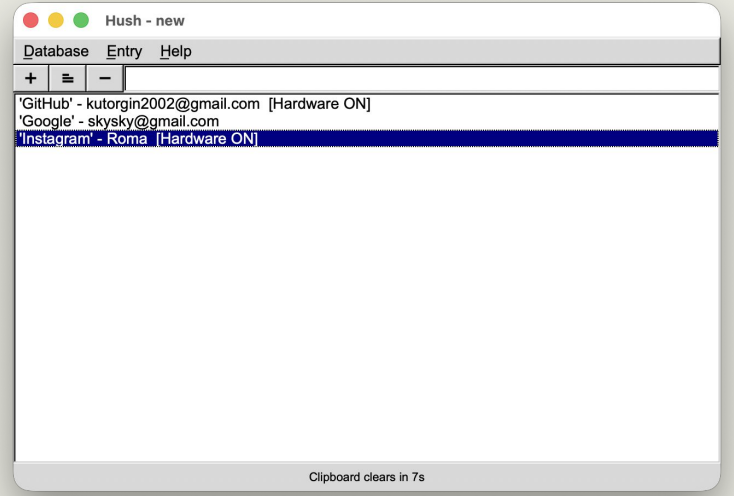
Привяжите запись к USB-устройству.
Hush запоминает уникальный отпечаток: ID производителя, ID продукта, серийный номер.

Запись доступна только при подключенном ключе. Без него — доступ заблокирован. Индикатор показывает статус: [Hardware ON] или [Hardware OFF].

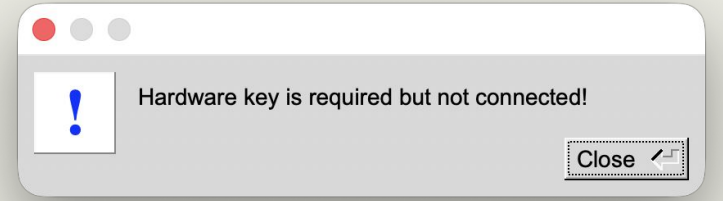
Даже если кто-то узнает ваш мастер-пароль, без физического USB-устройства он не откроет защищенные записи.



аппаратный_ключ.h



Попытка скопировать пароль
при ПОДКЛЮЧЕННОЙ флешке



Попытка скопировать пароль
при ВЫКЛЮЧЕННОЙ флешке

$ cat технический_стек.txt

```
[CORE]
- C++20: современный синтаксис
- stdlib only, zero bloat

[GUI]
- FLTK 1.4.4: 1990s эстетика при очень высокой производительности
- binary size: ~500KB (к примеру приложения на Electron весят 50MB+)

[CRYPTO]
- специальный трехпроходный шифр XOR
- прочитать исходник, понять код

[HARDWARE]
- IOKit: низкоуровневый доступ к USB-устройствам
- обфускация строк во время компиляции
- никаких сетевых вызовов, никогда

$ 
```

заключение.md

---

hush - менеджер паролей

для разработчиков
созданный разработчиками
которые ценят софт,
делающий одно дело хорошо.

```
$ git clone
https://github.com/r0manch1k/hus
h
$ make
$ ./build/hush

$ 
```