



# The Mooltipass Open Source Hardware Authentication Ecosystem

... from the electronics to the high-level software

Mathieu Stephan

# Hello!

## I am Mathieu Stephan

- Embedded systems engineer
- Former writer for Hackaday
- [www.limpkin.fr](http://www.limpkin.fr)
- Mooltipass project founder



# The Mooltipass Ecosystem

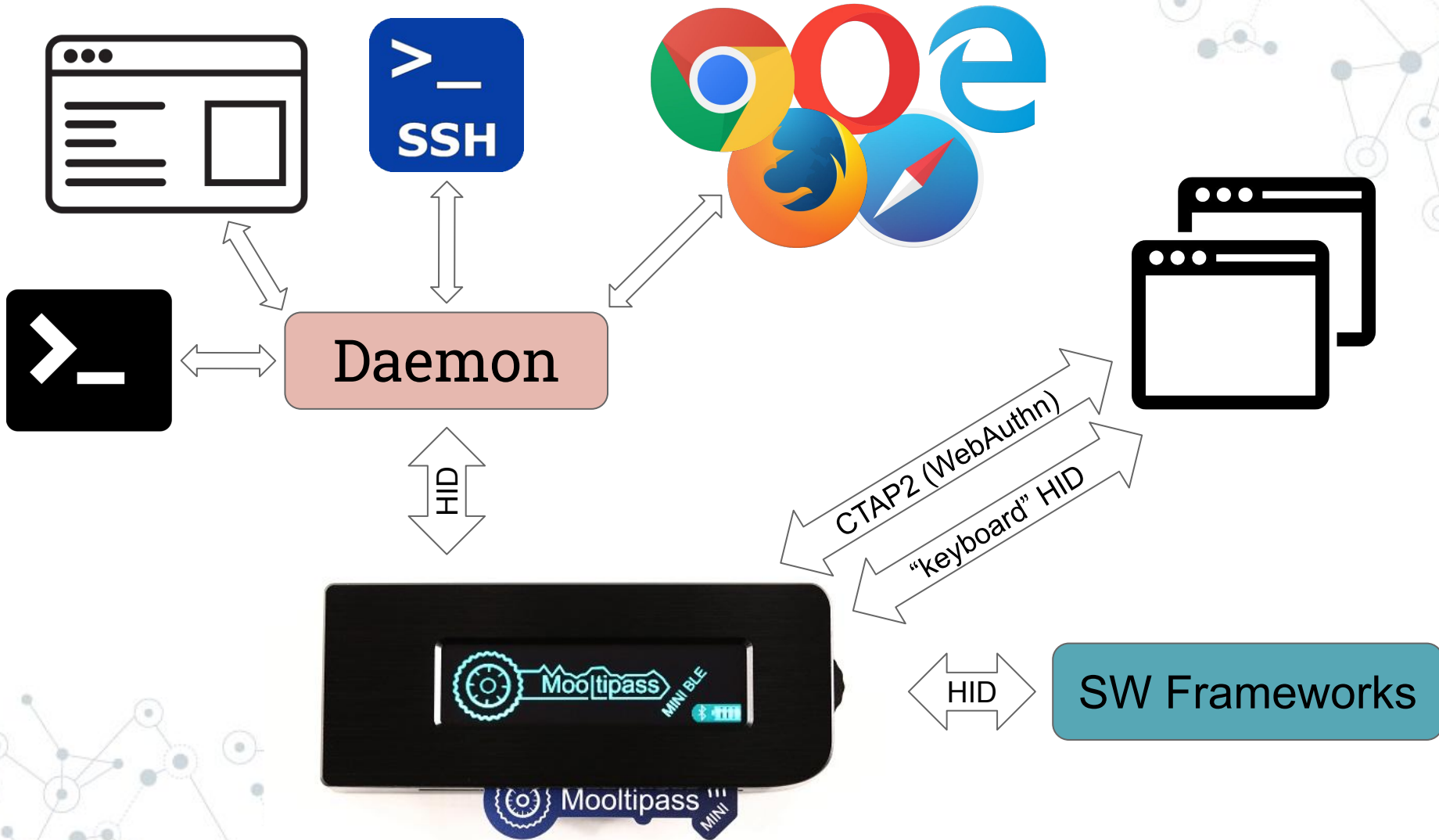
- Credentials, notes & files storage
- SSH & WebAuthn authentication
- TOTP support
- Multiple users
- Cross platform (tools)
- Native browser integration
- Open software & hardware



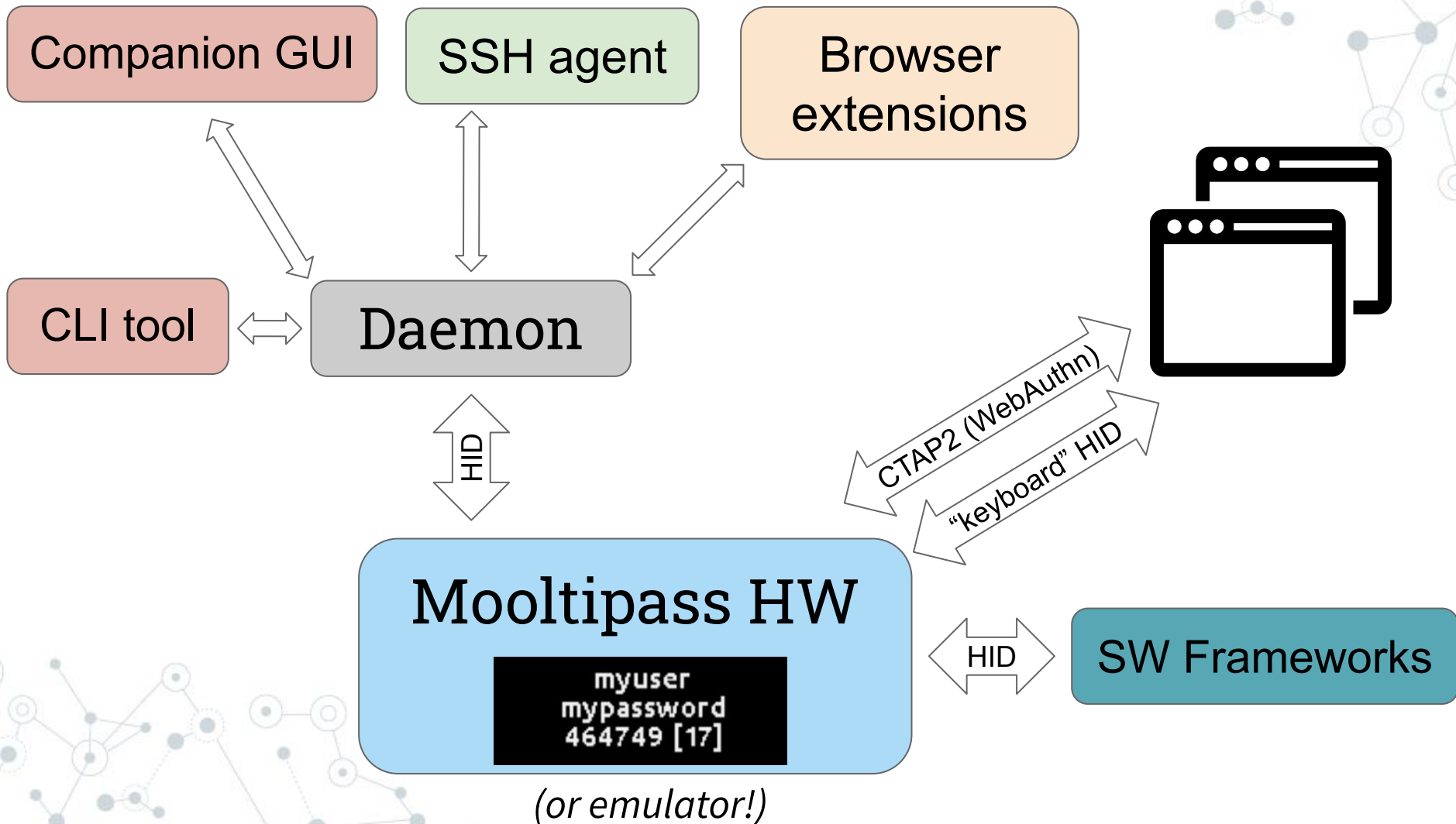
# Globally Distributed Contributors



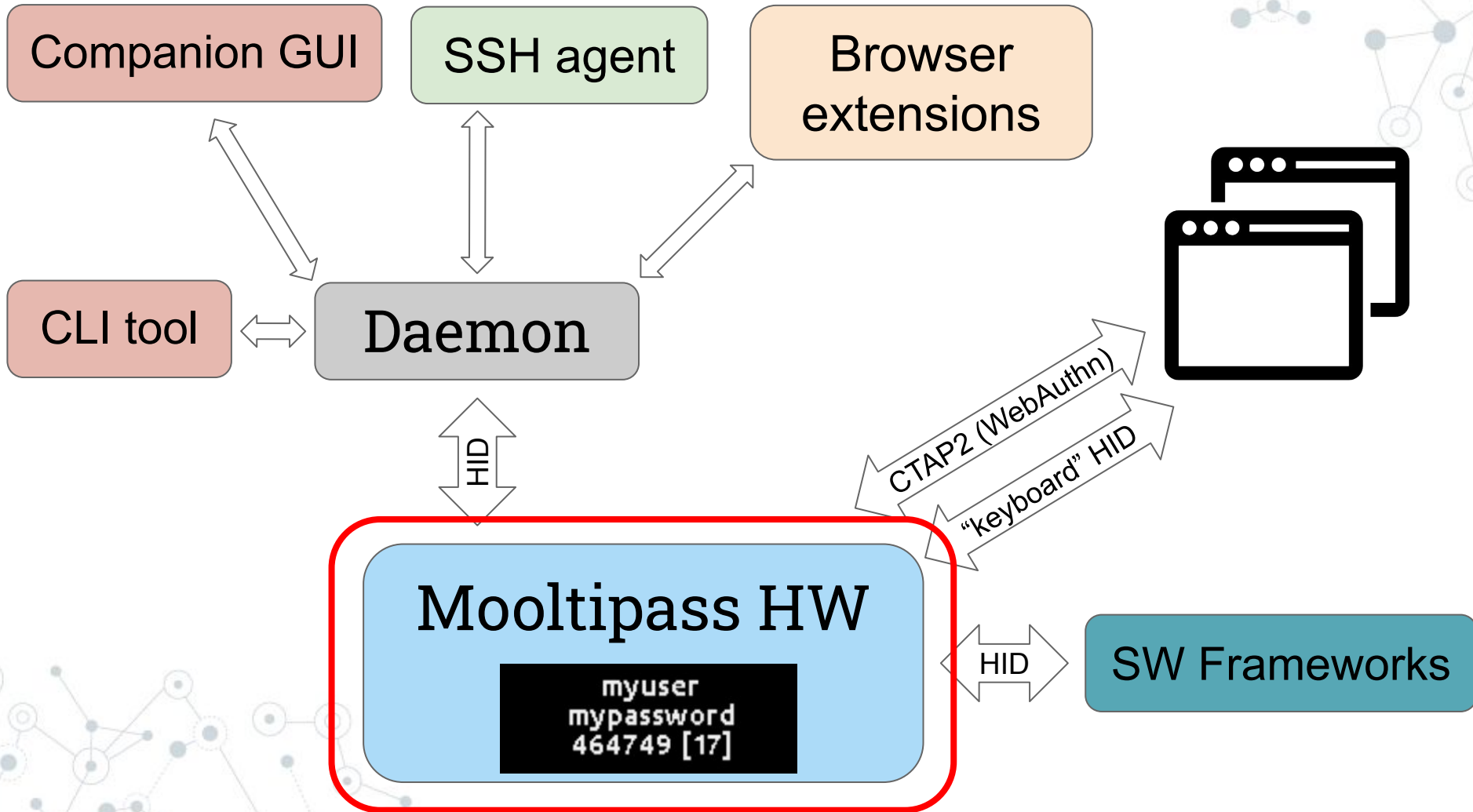
# The Mooltipass Ecosystem



# The Mooltipass Ecosystem



# The Mooltipass Hardware



# Three Hardware Devices



*Dec. 2014*



*Nov. 2016*



*May 2021*

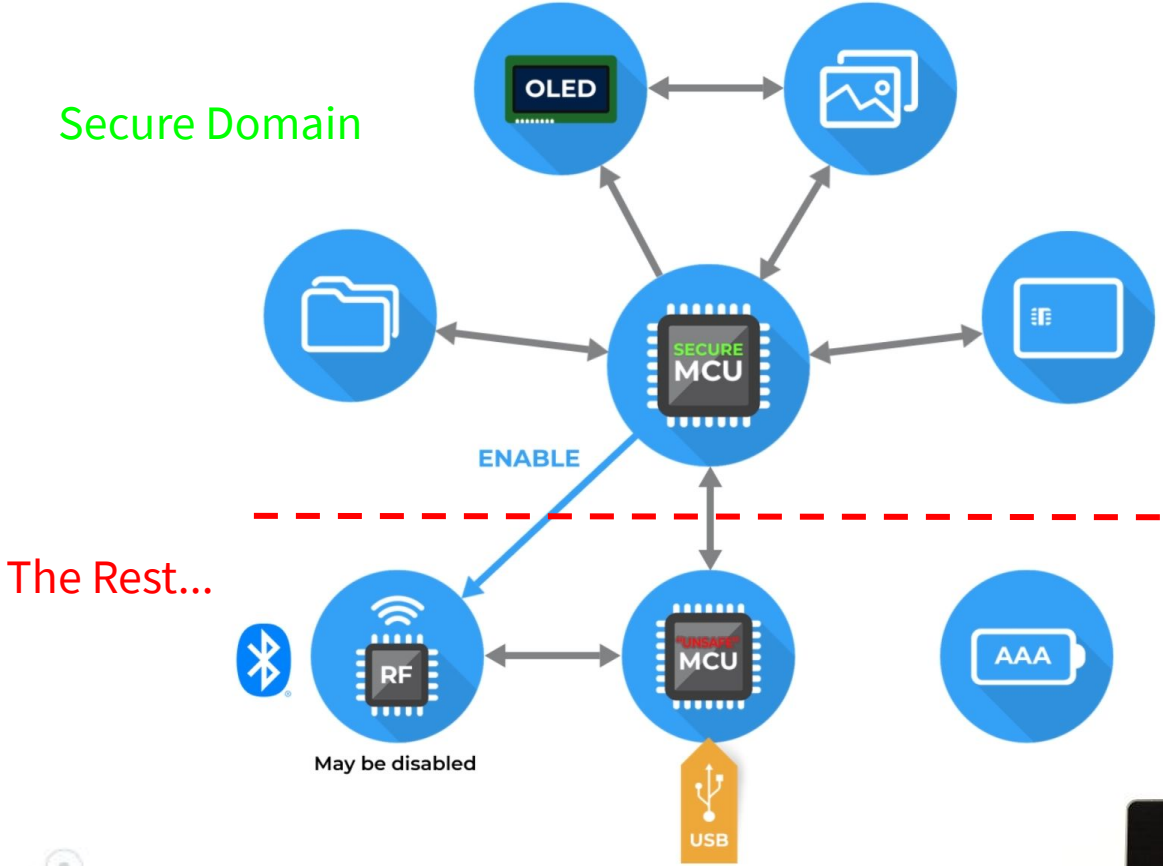


# Made in Aluminum



*...designed to be tamper-evident*

# Its Latest Model: the Mini BLE



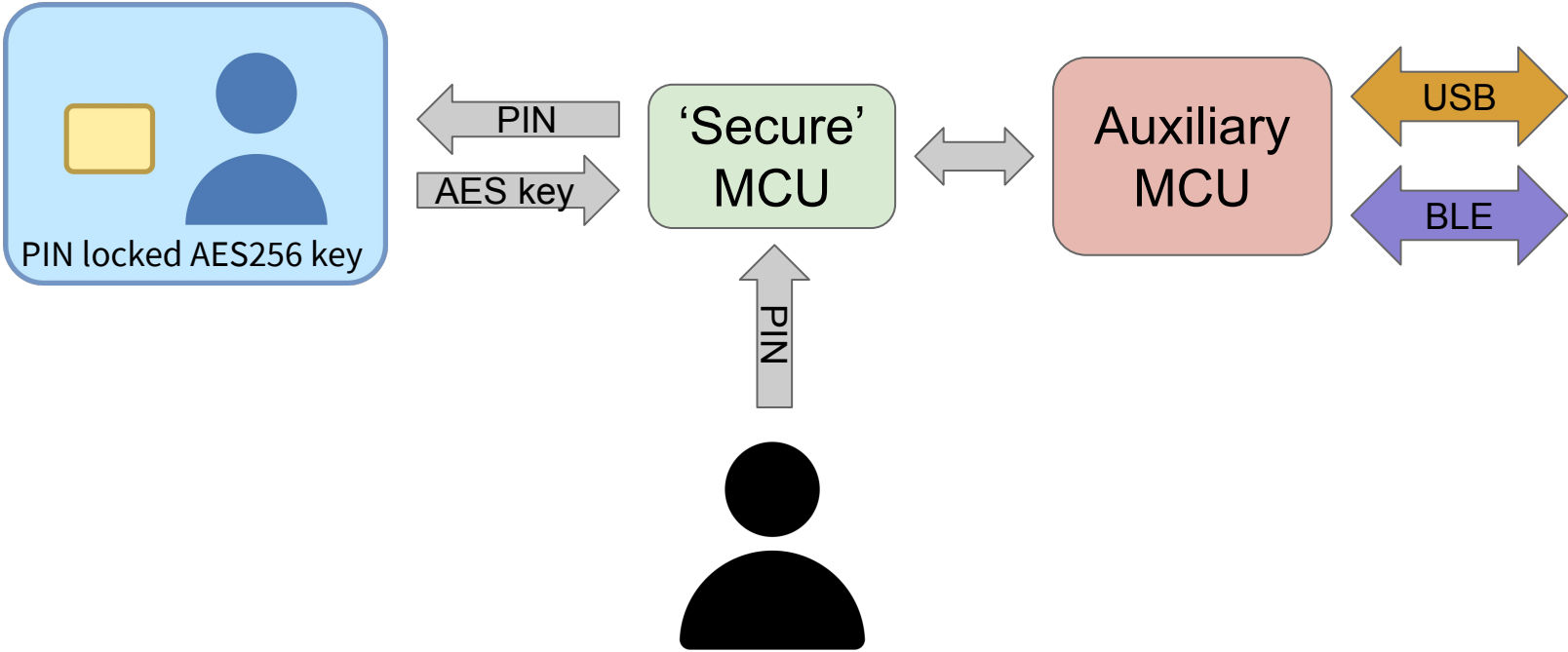
# Dual Microcontroller Design

- One MCU dedicated to secure operations
- Another for communicating with the outside world
- ‘Old school’ fixed frame length UART link between them

... so that:

- Protocol stacks run in the non secure MCU
- Complex libraries (CBOR) run in the non secure MCU

# Data Decryption Flow



# The Microcontrollers

- Both ATSAM21
- Internally store:
  - Unique encryption & signing keys
  - User profile data
- Can be updated through uniquely signed firmware updates

# Flashing the Firmware



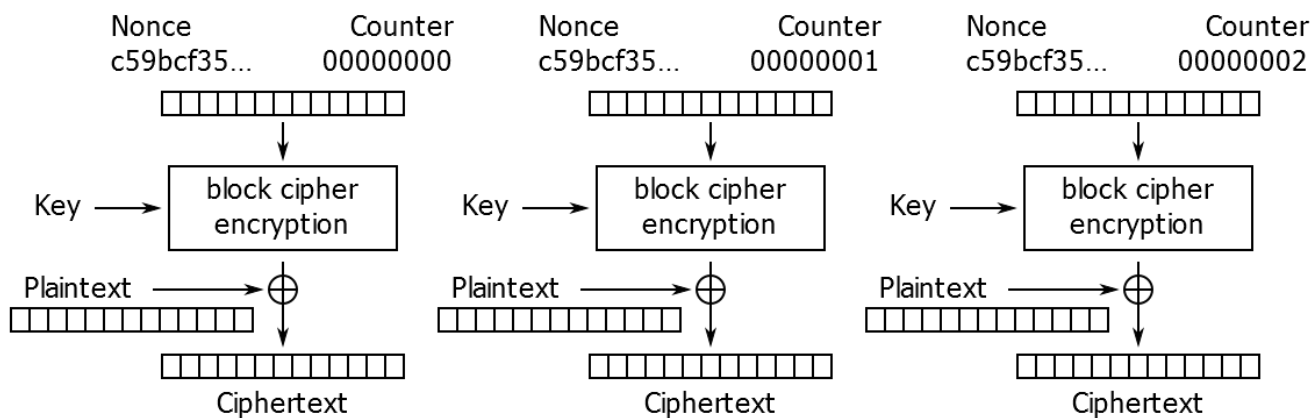
*Custom-made programming jig*



# Firmware Features

# Firmware - AES Encryption

- Using BearSSL, CTR mode
- Checked against NESSIE vector sets



Counter (CTR) mode encryption



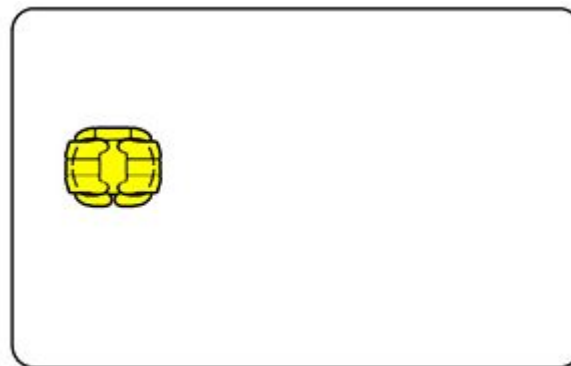
# Firmware - Encrypted Storage

- Dedicated flash memory used for storage
- 2 types of data
  - Credentials (logins/passwords, WebAuthn data)
  - Encrypted blobs (small files, notes)
- Sorted linked list data structure
- Multiple databases for >100 users



# Firmware - Smartcard Use

- Ubiquitous form of read-protected memory
- 16-bit PIN access (“0000” to “FFFF”)
- Permanently locked after 4 incorrect PINs
- Cheap (<\$1) in volume




# Firmware - RNG

- Using the onboard accelerometer
- Using the 2LSb of the 16bits output word
- Checked using dieharder tests

# Firmware - USB & Bluetooth HID

A decorative network diagram in the top right corner, consisting of various sized circles (nodes) connected by thin lines (edges). Some nodes are solid grey, while others are hollow with a grey outline. The connections form a complex, interconnected web.

- HID keyboard and ‘proprietary’
  - Proprietary channel for integration plugins
  - Keyboard channel for manual password recall
- 
- A decorative network diagram in the bottom left corner, similar to the one in the top right. It features a cluster of nodes connected by lines, with some nodes highlighted in solid grey and others as hollow circles.

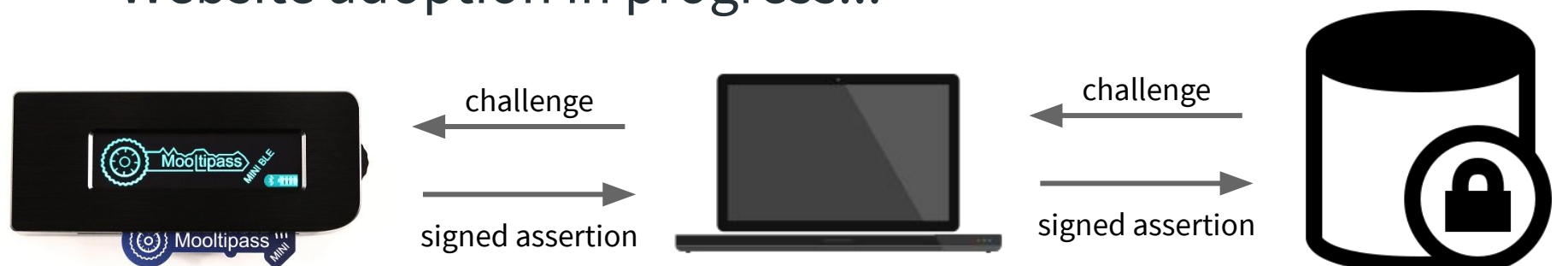
# Manual Credential Recall



- Using HID, key presses are sent to type credentials
- > 50 layouts currently supported
- Look-up tables generated by parsing the Unicode Common Locale Data Repository
- Allows credential typing on any platform / application

# Firmware - WebAuthn

- Support added by a single contributor
- Essentially a private / public key registration
- Also called FIDO2, successor of FIDO U2F
- Website adoption in progress...



# Firmware - Graphics Library



- Designed from the ground up
- Optimized for speed
- RLE compression for bitmaps
- Support for multiple fonts
- Support for non contiguous character blocks

# Firmware - Data Storage Flash

- Designed a simple read-only file system to store:
  - Generic and language-specific bitmaps
  - Character to HID key presses look up tables
  - Strings for each supported language
  - Firmware updates
  - Fonts



... CRC-checked and CBCMAC checked

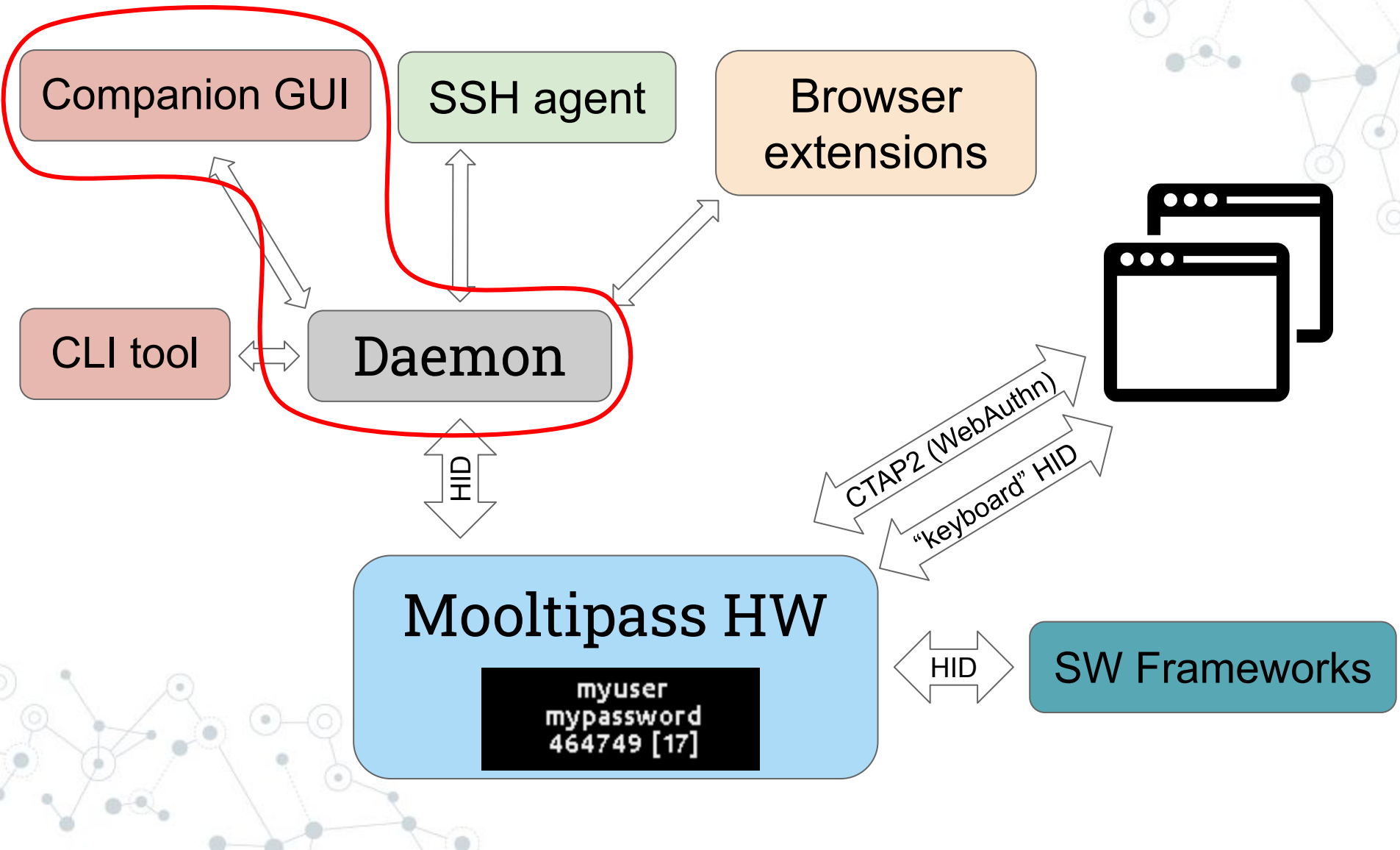


# Firmware - noteworthy features

- On device language switching
- Customizable behavior (prompts/notifications...)
- On device card & user profile operations
- Credential categories with selection filter
- Custom NiMH charging algorithm
- Custom SoC algorithm



# The Mooltipass Software



# Moolticute (Daemon)

**Moolticute**

Device Settings | Credentials | Notes | Files | Fido2 | Synchronization | Settings | About | 0% | Mooltipass

## Device Settings

These settings change parameters on your Mooltipass device. For browser specific settings, go to the options page of the extension.  
Hint: keep your mouse positioned over an option to get more details.

**Keyboard Output**

Device Default Language: English US

Current User Language: English US

Bluetooth Keyboard Layout: Belgium French  Enforce at connection

USB Keyboard Layout: Belgium French  Enforce at connection

For slow computers: wait 15 ms after each key press

By default, send Tab after each login output

By default, send Enter after each password output

**Inactivity**

Cancel credentials request after 15 seconds

Inactivity Timer: No Inactivity

**Advanced Settings (See User Manual)**

Lock Unlock Feature: Login + Password  No Password Input Prompt

Knock Detection Sensitivity: Medium

Information Time Delay: 3 seconds

Screensaver: None

Display the security hash before & after the card is unlocked

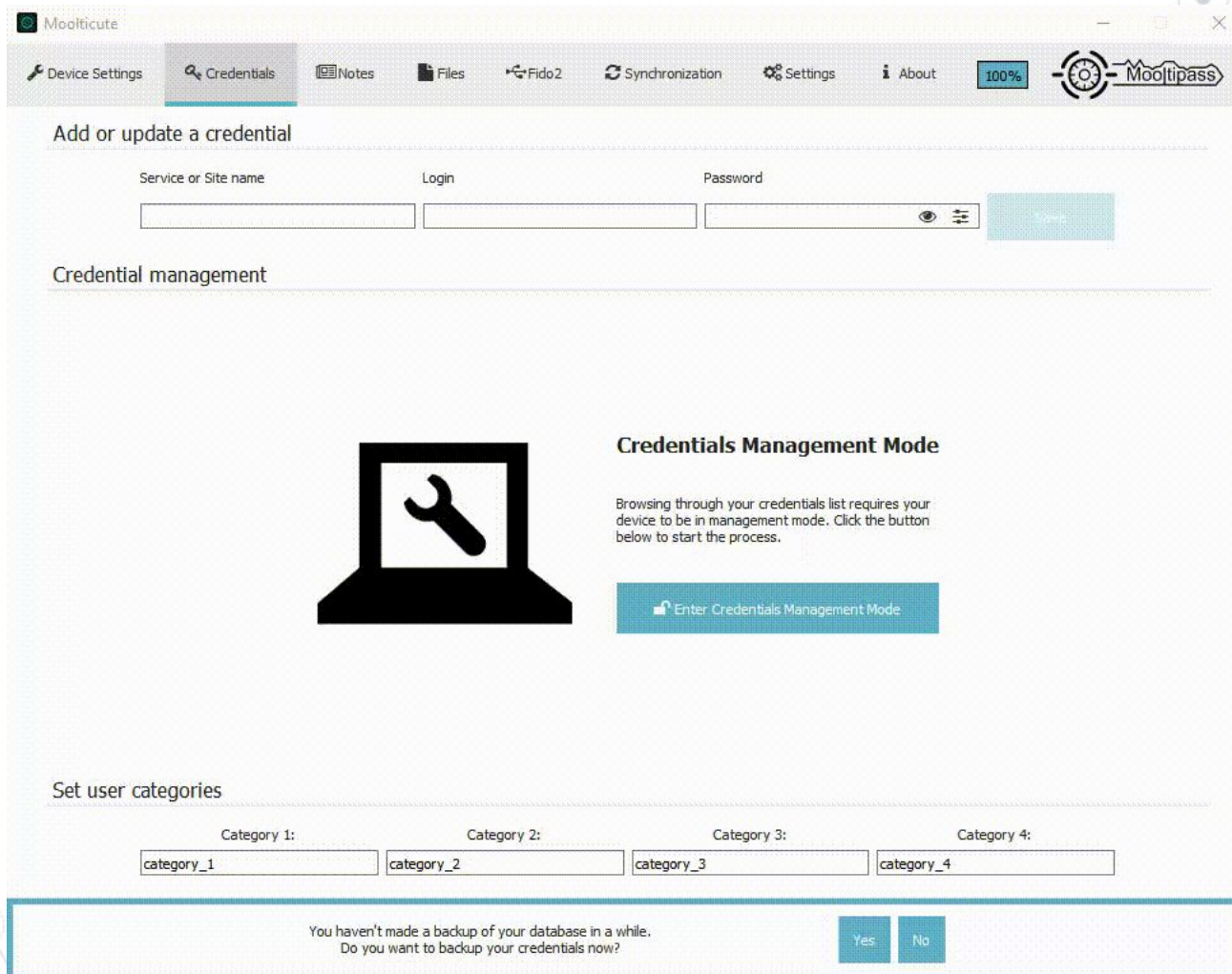
Display Pin on Back Pressed

Reset default values

Credentials on the device are more recent.  
Do you want to export your database?

Yes No

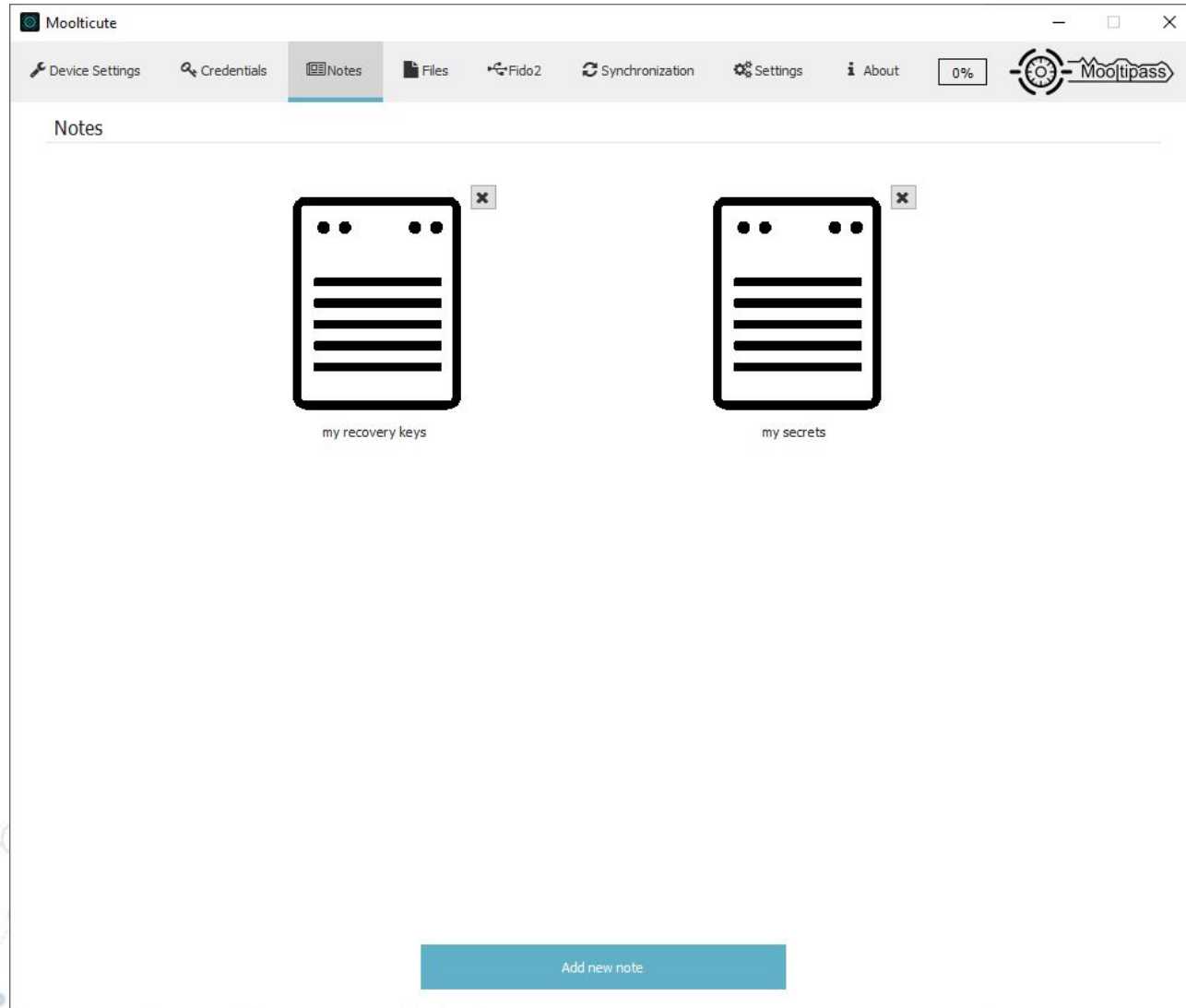
# Moolticute (Daemon)



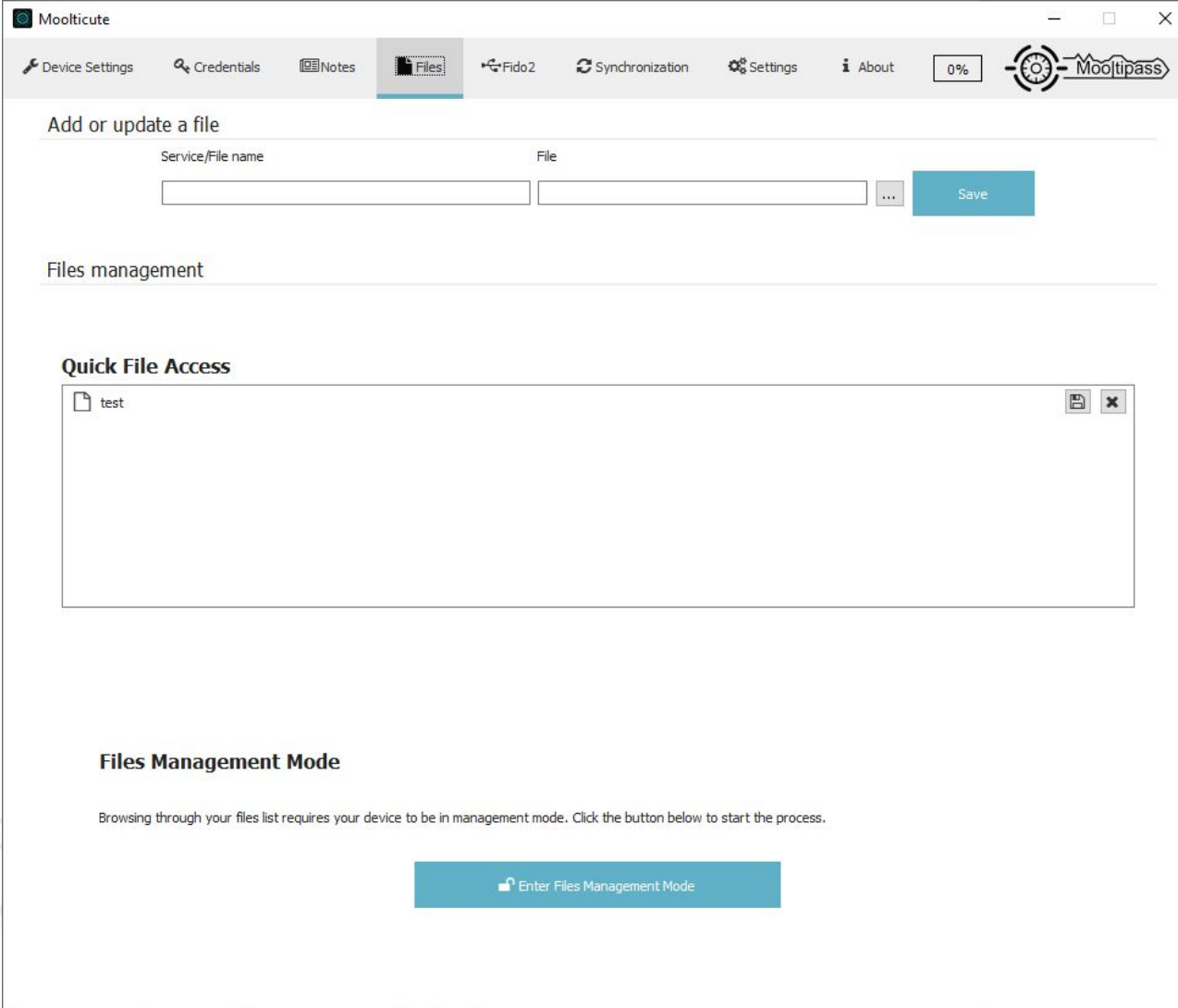
The screenshot shows the Moolticute application window with the following elements:

- Toolbar:** Device Settings, Credentials (selected), Notes, Files, Fido2, Synchronization, Settings, About, 100% zoom, and Mooltipass logo.
- Section: Add or update a credential**
  - Fields: Service or Site name, Login, Password (with eye icon and dropdown menu).
  - Submit button.
- Section: Credential management**
  - Icon of a laptop with a wrench.
  - Credentials Management Mode**
    - Text: "Browsing through your credentials list requires your device to be in management mode. Click the button below to start the process."
    - Button: Enter Credentials Management Mode.
- Section: Set user categories**
  - Category 1: category\_1
  - Category 2: category\_2
  - Category 3: category\_3
  - Category 4: category\_4
- Notification:** "You haven't made a backup of your database in a while. Do you want to backup your credentials now?" with Yes and No buttons.

# Moolticute (Daemon)



# Moolticute (Daemon)



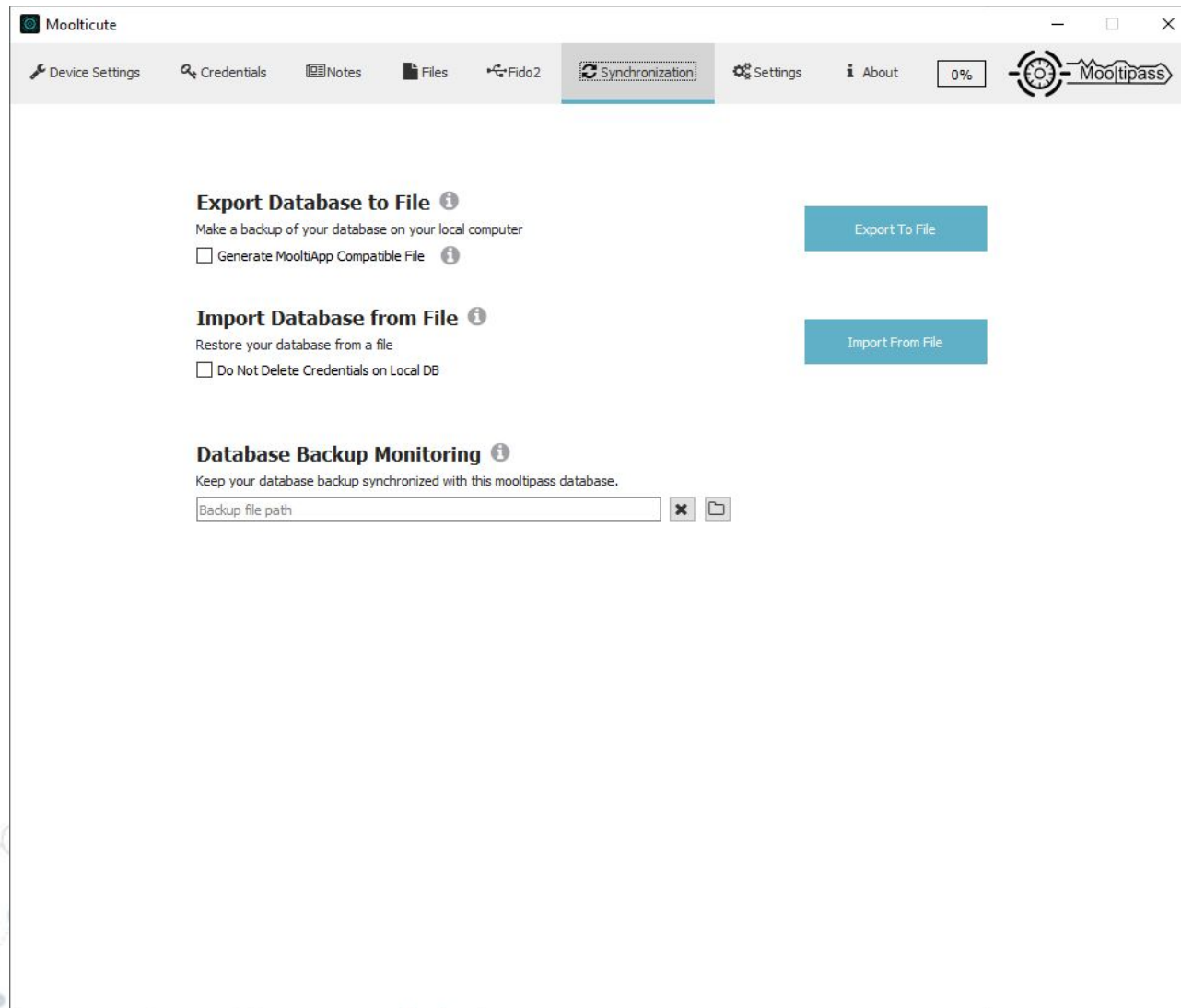
The screenshot displays the Moolticute application window. The title bar reads "Moolticute". The top navigation bar includes "Device Settings", "Credentials", "Notes", "Files" (which is the active tab), "Fido2", "Synchronization", "Settings", "About", a "0%" progress indicator, and the "Mooltipass" logo.

The main content area is titled "Add or update a file". It features two input fields: "Service/File name" and "File". A "Save" button is positioned to the right of the "File" field. Below this section is a "Files management" section.

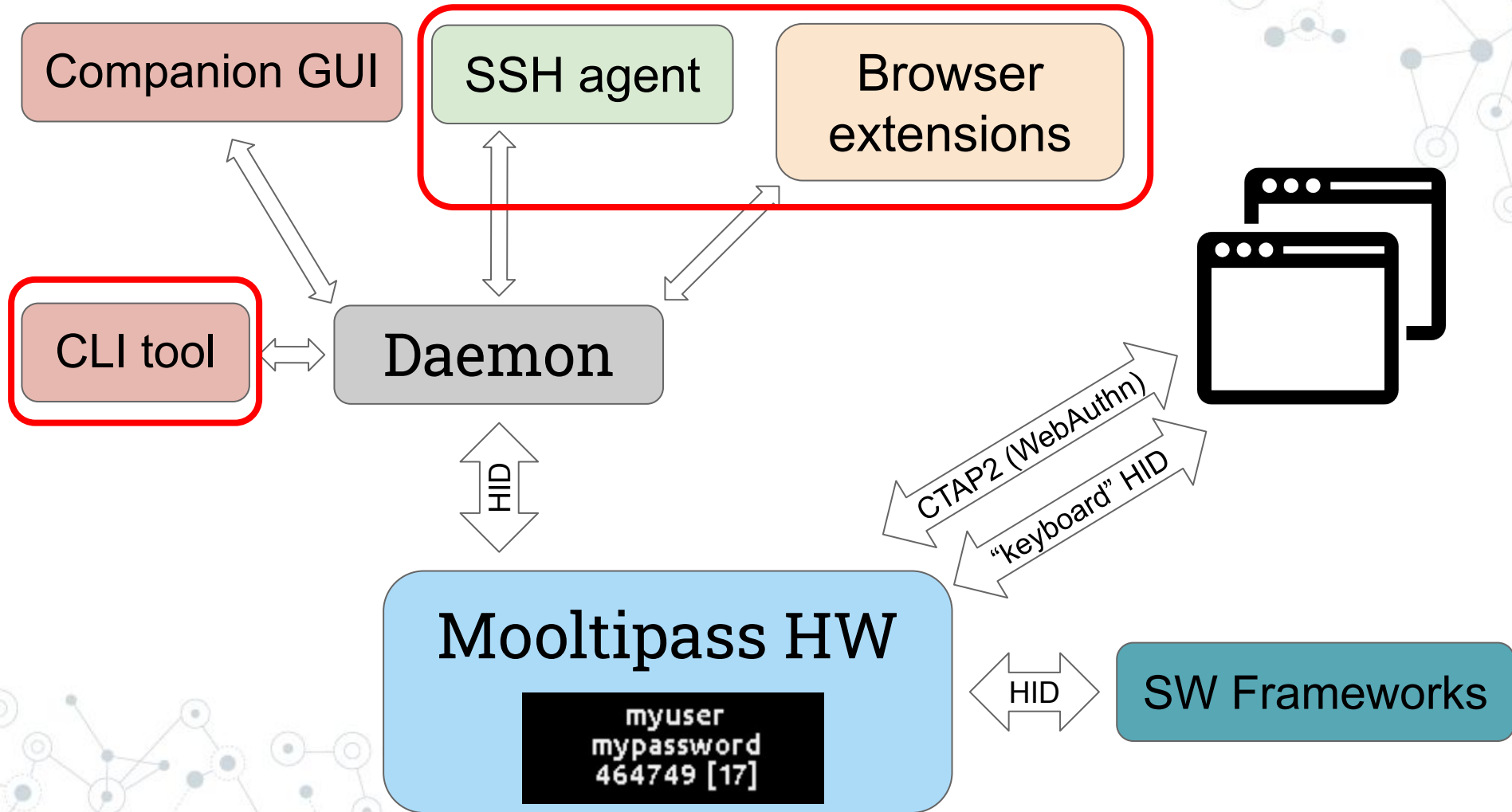
The "Quick File Access" section shows a list of files, currently containing one file named "test". Each file entry has a file icon on the left and "edit" and "delete" icons on the right.

The "Files Management Mode" section contains the following text: "Browsing through your files list requires your device to be in management mode. Click the button below to start the process." Below this text is a large blue button labeled "Enter Files Management Mode".

# Moolticute (Daemon)



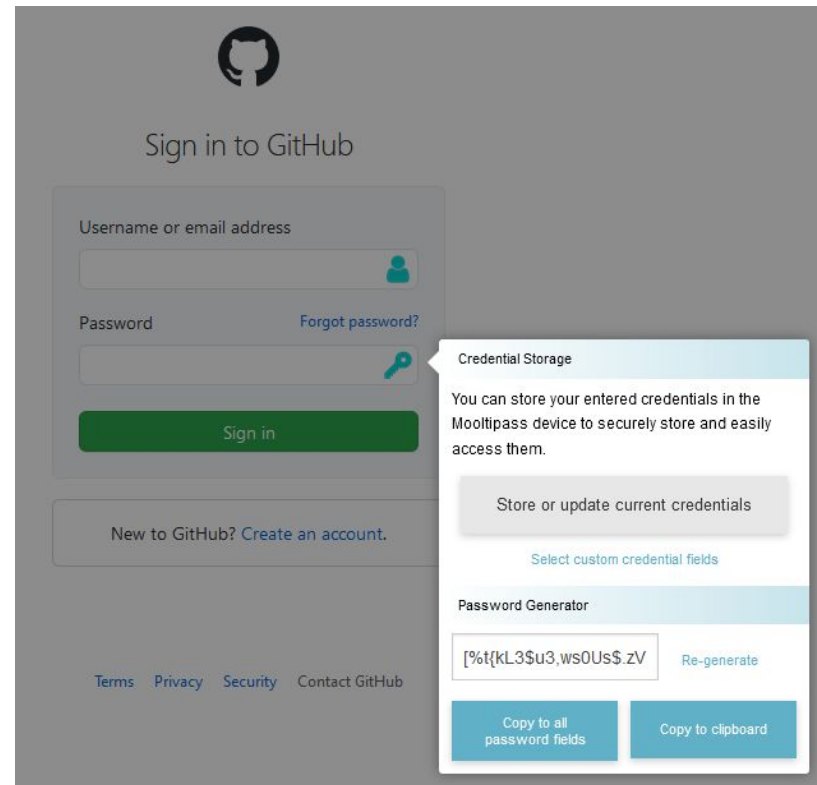
# The Mooltipass Software





# Browser Extensions

- Chrome
- Firefox
- Opera
- Edge
- Safari (beta)



# Command Line Tool

- <https://github.com/raoulh/mc-cli>
- Written in go

```
C:\Users\>mc-cli

Usage: mc-cli COMMAND [arg...]

Command line tool to interact with a mooltipass device through a moolticute daemon

Commands:
  login      Manage credentials stored in the device
  data       Import & export small files stored in the device
  parameters Get/Set device parameters

Run 'mc-cli COMMAND --help' for more information on a command.

C:\Users\>mc-cli login get test mylogin
mypassword
```

# SSH agent

- <https://github.com/raoulh/mc-agent>
- Also written in go
- Retrieves private keys from the device

# Python Framework

- <https://github.com/oSquat/mooltipy> (for the mini)
- scripts/python\_framework (for the mini BLE)

```
C:\Users\python_framework>python mooltipass_tool.py platInfo
USB device found
Received aux MCU ACK
Received main MCU ACK
Connected to device

Platform Info
Aux MCU major: 0
Aux MCU minor: 62
Main MCU major: 0
Main MCU minor: 74
Platform serial: 4294967295
Bundle version: 65535
```

# Contributors Wanted!

Firmware:

- Bluetooth custom service for smartphones
- (Long) list of enhancements on our github

Android / iOS

- Autofill service app





# Thanks!

## Questions?

You can find me at:

limpkin on irc.libera.chat

[mathieu@themooltipass.com](mailto:mathieu@themooltipass.com)

[github.com/mooltipass](https://github.com/mooltipass)

