ONE MANAGER : SECRETS MANAGEMENT SYSTEM

Introduction:

Secrets management refers to the tool for managing digital authentication credentials (secrets), including passwords, keys, APIs, and tokens for use in applications, services, privileged accounts and other sensitive parts of the IT ecosystem.

Secrets include:

1. User or auto-generated passwords
2. API and other application keys/credentials (including within containers)
3. SSH Keys
4. Database and other system-to-system passwords.
5. Private encryption keys for systems like PGP

The objective of this project is to implement an application to manage the secrets securely.It stores your secrets securely by ciphering it using one the most used and advanced cryptographic algorithms.

Present System:

People use methods like storing secrets in a file or hard code it in some application or writing it somewhere etc which are not the best way of storing the secrets because it can be accessed by anyone easily, thus getting access to your personal information.

Major Limitations of Present System :

- ☐ It stores secrets in plain text format
- ☐ It allows anyone to access the data

Proposed System:

This system will store the secrets by ciphering the secrets in the best possible way,it also imposes access rules ,thus allowing only the creators to access the secret.

Advantages of Proposed System:

- ☐ Secured Secrets
- ☐ Authorised Access

Hardware Requirements:

- CPU with clock speed 2.0GHz & above
- 2 Gigabyte RAM and above

Software Requirements:

- Windows 10 & above
- Python 3.9 & above
- PostgreSQL 14.0 & above

Packages Used:

- Rich (https://github.com/Textualize/rich):
  Rich is a Python library for rich text and beautiful formatting in the terminal.
- Psycopg (https://www.psycopg.org/psycopg3/):
  New generation PostgreSQL database adapter for the Python programming language
- Pycryptodomex (https://www.pycryptodome.org/) :
  A self-contained cryptographic library for Python
- Winreg (https://docs.python.org/3/library/winreg):
  Windows registry access
- Proto-plus (https://github.com/googleapis/proto-plus-python):
  Beautiful, idiomatic protocol buffers in Python
- Pyperclip(https://github.com/asweigart/pyperclip):
  Python module for cross-platform clipboard functions.

CryptoGraphical Overview:

In One Manager we have used XChaCha Poly1305 (https://en.wikipedia.org/wiki/ChaCha20-Poly1305#XChaCha20-Poly1305_-_Extended_Nonce_Variant) which is faster and could be more secure than the industry standard AES (https://nordpass.com/blog/xchacha20-encryption-vs-aes-256/). Major companies nowadays preferred XChaCha over AES like Google , Nordpass, In TLS v1.3 .

For cryptographic keys used for ciphering the plain text is derived by Scrypt (https://en.wikipedia.org/wiki/Scrypt) which is a Key Derivation Function. It derives a cryptographic key based on a passphrase .The main advantage of scrypt is it generates a unique key every time for the same passphrase .Thus if a keys get leaked only a single secrets get leaked but other secrets remain safe

Setting Up the Application:

1. Get the source code from Github (https://github.com/Shahprogrammer/OneManager)
2. Install the dependencies
3. Set the environment variables:
    a. OneManager_DB_HOST ⇛ Host address of database
    b. OneManager_DB_PORT ⇛ Port of database
    c. OneManager_DB_NAME ⇛ Name of database
    d. OneManager_DB_USERNAME ⇛ Username of database
    e. OneManager_DB_PASSWORD ⇛ Password of database
4. Open terminal in source code directory
5. Run command '`python3 -m one_manager.app --setup`'

Running the Application

1. Set up the environment variable as above shown
2. Open terminal in source code directory
3. Run command '`python3 -m one_manager.app`'

OUTPUT

➳ Login:-

➳ Login(Error):-

➳ Register :-

➳ Register(Error):

➳ Register (Error):

➳ Main Menu:

➳ Store a secret

➳ Store a secret(Error):

➳ View a Secret (Copy to Clipboard):

➳ View a Secret:

➳ View a Secret(Error):

➳ Update a Secret

➳ Update a Secret(Error):

➳ Delete a Secret:

➳ Delete a Secret(Error):

➳ Generate Random Password & Auto Save:

➳ Generate Password:

➳ Generate Password (Copy to Clipboard):

➽ Exit


ONE MANAGER
MENU
1) Store a secret
2) View a secret
3) Update a secret
4) Delete a secret
5) Generate random password
6) Exit
Please select a choice [1/2/3/4/5/6]: 6
Good bye
Press any key to exit ...

Bibliography:

- https://github.com/Textualize/rich
- https://www.psycopg.org/psycopg3/
- https://www.pycryptodome.org/
- https://docs.python.org/3/library/winreg
- https://docs.python.org/3/library/uuid
- https://github.com/googleapis/proto-plus-python
- https://github.com/asweigart/pyperclip