Автор:
Магда Матвій Олексійович
студент 42ІПЗ групи
Науковий керівник:
кандидат фізико-математичних наук,
доктор педагогічних наук,
доцент кафедри комп'ютерної та програмної інженерії,
Малежик Петро Михайлович

ПРОГРАМНИЙ КОМПЛЕКС КЕРУВАННЯ ОБЛІКОВИМИ ЗАПИСАМИ КОРИСТУВАЧА

Анотація. Останні роки загрози для побутових і корпоративних користувачів тільки збільшуються. Особливо це стосується облікових даних та паролів до них. Для комфортної адаптації у світі кібербезпеки необхідно інтегрувати рішення та інструменти для забезпечення безпеки облікових даних серед масового користувача.
Ключові слова: пароль, ключ, програма, менеджмент, кібербезпека.

Вступ. Рівні загроз у кібербезпеці еволюціонують із кожним днем. Так само і розвиваються відповідні до них заходи та інструменти з безпеки. Однією з найважливіших проблем у кібербезпеці є нехтування правилами та рекомендаціями щодо створення, зберігання та управління критично важливими обліковими та іншими даними, такими як: дані облікових записів та їх паролі, ключі доступу до ресурсів та інше. Ця проблема є актуальною як в побутовому так і в корпоративному та урядовому сегментах. Її наслідками можуть бути втрати критично важливих даних, що може спричинити величезні збитки та безповоротну втрату даних. Для часткового вирішення цієї проблеми достатньо дотримуватись простих рекомендацій. Для повного вирішення необхідно використовувати спеціально призначені для цього сервіси, так звані менеджери паролів або секретів.

Мета роботи полягає в наступному: створити програмний комплекс для створення, безпечного зберігання та керування обліковими даними.

Виклад основного матеріалу. Проблему створення, або ж іншими словами – генерації паролів або ключів можна частково вирішити ручним способом, дотримуючись широковідомих рекомендацій щодо генерації ключів: використовувати як великі, так і маленькі літери, використовувати цифри, спеціальні символи та інакші ніж латинські літери, наприклад кирилицю. Також довжина ключа має бути достатньо великою. На сьогоднішній день 32 символи є достатній рівнем для довжини паролю, проте методи атак зловмисників еволюціонують із кожним днем, тому ця довжина постійно збільшується. Проте, навіть дотримання всіх цих правил не забезпечує стовідсоткову надійність згенерованому ключу, бо як на своєму досвіді показує соціальна інженерія – у багатьох випадках ключ, згенерований людиною, можна відтворити, використовуючи методи соціальної інженерії. Якщо ж ключ буде генерувати програма за допомогою криптографічно-стійкого алгоритму – ймовірність його відтворення буде близькою до нуля. Навіть якщо проблема створення генерації ключів та паролів вирішена, проблема їх зберігання нікуди не зникає. Яким би надійним не був згенерований ключ – якщо недбало відноситися до його зберігання, він легко може бути отриманий зловмисниками. На сьогоднішній день самий надійний спосіб зберегти ключ або пароль – це запам'ятати його і ніколи нікому не повідомляти і ніде не записувати. Проте хоч цей спосіб і є самим надійним, він не дозволить навіть побутовому користувачу користуватися ним у повній мірі. По правилам кібербезпеки не можна використовувати один і той самий ключ або пароль у декількох місцях, бо при компрометації ключа в одному ресурсі – він може бути використаний для доступу до іншого. Таким чином для кожного окремого ресурсі завжди

має бути унікальний ключ або пароль. Додаючи до цього ще й вищезазначені рекомендації з генерування ключів щодо їх довжини та різноманіття символів можна дійти до висновку що навіть середньостатистичному побутовому користувачу неможливо тримати своїй у пам’яті всі паролі та ключі доступу, так як на сьогоднішній день кількість різноманітних сервісів якими люди користуються щодня досить велика і лише збільшується. Через це, навіть якщо ми маємо надійні ключі або паролі, ми маємо їх десь зберегти. Зазвичай люди записують їх на папері, або ж у нотатки на комп’ютері або телефоні, але очевидно, що це ненадійно, бо при отриманні зловмисником доступу до девайсу він отримає всі дані. Можна використовувати більш надійні сховища для критичних даних, наприклад зберігати їх у зашифрованому сховищі для розшифровки якого необхідно використовувати спеціальний ключ, який для найбільшої надійності користувач має запам’ятати і ніде не записувати.

Система керування обліковими записами користувача об’єднує у собі генерування та зберігання ключів і паролів, а також дозволяє зберігати інші облікові дані та надає інструменти для керування. Її також часто називають менеджером паролів, менеджером ключів, або менеджером секретів. Необхідно створити програмну систему, що дозволяє генерувати унікальні паролі або ключі для облікових записів. Зберігати всі необхідні логіни, паролі та інші дані, що відносяться до облікових записів для подальшого їх використання користувачем. Система має дозволяти керувати обліковими записами (оновлення, видалення, пошук і т.д.). Програма повинна мати режими формування паролів за бажанням користувача, а саме: паролі із набору лише цифр, з набору лише літер, лише символів або самий захищений варіант - змішаний. Також необхідна функція зберігання вже введених чи використаних даних облікових записів користувачів.

Програма повинна бути багатомовною, підтримувати англійську та українську мови та дозволяти зручно додавати будь-які мови та редагувати вже існуючі. Дані облікових записів зберігаються на віддаленому сервері та частина з них шифрується двостороннім шифруванням. Програма є серверною та складається з кількох мікросервісів, що взаємодіють між собою. Програма безпосередньо взаємодіє з базою даних, де зберігаються всі дані та має бути розташована на віддаленому сервері або безсерверному середовищі. Вона має містити frontend-частину, яке буде реалізована у вигляді чату з інтуїтивним інтерфейсом за допомогою Telegram Bot API.

Програма повинна бути у вигляді файлу набору .dll файлів програмного середовища .NET. Програма має запускатися на ОС Linux. Для розгортання програми було обране безсерверне середовище .NET 6 на хмарній платформі Amazon Web Services на сервісі AWS Lambda. У якості бази даних було обрано Microsoft SQL Server, що буде знаходитись на AWS RDS – сервісу реляційних баз даних. Серед основних функцій системи слід зазначити зручний пошук та можливість легкої заміни однієї групи даних на іншу. Можна вручну створювати всі паролі без використання паролевового утворювача. Основною перевагою програми є захищеність всіх даних, а також шифрування паролів, що зменшує ймовірність втручання із сторони. Паролі та ключі шифруються за допомогою ключа шифрування, який користувач має запам'ятати та ніде не зберігати. При втраті ключа відновити зашифровані дані буде неможливо. Для шифрування паролів було обрано найнадійніший сучасний симетричний алгоритм шифрування AES-256. Зручність використання системи за допомогою чату месенджеру Telegram полягає в тому, що користуватися програмою можна з будь-якого пристрою, де встановлений Telegram месенджер, отже вона є кросплатформеною. Користувачами системи можуть бути будь-які побутові користувачі. Реєстрація користувачів програми відбувається системним адміністратором за допомогою інтерфейсу програми.

Загальна характеристика та особливості роботи

Програма має включати в себе такі функціональні можливості:

1. Головне меню, що комбінує собою пошук облікових записів та поле для вводу команд керування програмою;

2. Меню пошуку, що дозволяє переглядати список облікових записів;

3. Меню облікового запису користувача, що дозволяє переглядати дані облікового запису, переглядати пароль, змінювати дані облікового запису користувача, видаляти обліковий запис користувача;

4. Меню-сценарій додавання нового облікового запису користувача дозволяє покроково додавати необхідні дані облікового запису. Необов'язкові кроки дозволяється пропускати;

5. Меню зміни даних облікового запису, що дозволяє змінювати назву облікового запису, посилання на сервіс, видаляти посилання на сервіс, змінювати логін та пароль та повертатися на попереднє меню.

6. Меню налаштувань генератора паролів дозволяє налаштовувати опції генератора паролей, а саме:

6.1. Вмикати або вимикати символи нижнього регістру;

6.2. Вмикати або вимикати символи верхнього регістру;

6.3. Вмикати або вимикати цифри;

6.4. Вмикати або вимикати спеціальні символи;

6.5. Вмикати або вимикати опцію, що примусово ставить першим символом паролю літеру;

6.6. Змінювати довжину паролю від 1 до 2048 символів;

7. Меню видалення облікового запис показує попередження перед видаленням запису для уникнення помилкового видалення;

8. Меню зміни мови дозволяє змінювати мову інтерфейсу програми;

Висновок. Для забезпечення безпеки своїх облікових даних необхідно використовувати рішення та інструменти для менеджменту обліковими записами користувача. Для інтегрування таких рішень у маси можна інтегрувати рішення у соціальні мережі, що дозволить побутовим користувачам простіше адаптуватися до них.

Список використаних джерел

1. Armstrong D. The Quarks of Object-Oriented Development / Deborah J Armstrong., 2006. – 245 с.
2. Paul K. Microsoft open-sources Entity Framework / Krill Paul, 2012., - 54 с.
3. SafetyDetectives. The 20 Most Hacked Passwords in the World: Is Yours Here?. SafetyDetectives. URL: https://www.safetydetectives.com/blog/the-most-hackedpasswords-in-the-world/
4. Kogent Solutions Inc. ASP.NET 3.5 Black Book, 2009., - 134 с.
5. Neumann A. The Integrated Information Environment / Alexander Neumann, 2021. – 45 с.
6. Everett M. Silicon Valley fever: growth of high-technology culture / M. Everett, K. Judith., 1984. – 263 с