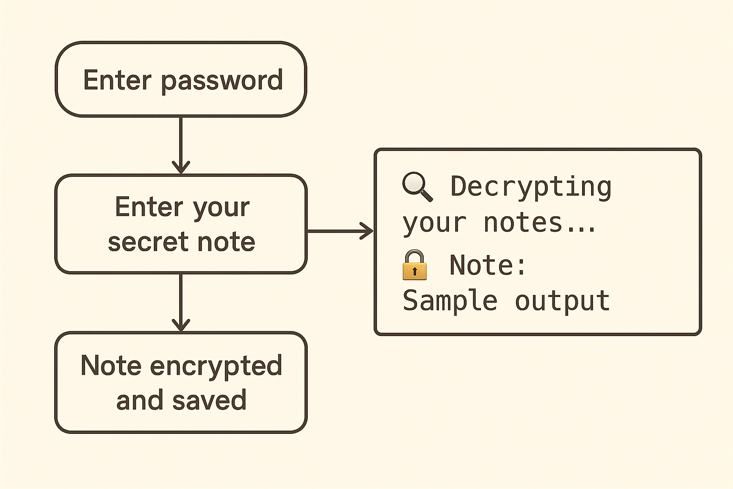
# Encrypted Note Storage - C++ Application (AES-256)

This document explains the functionality and structure of the C++ program that securely encrypts and decrypts text notes using AES-256-CBC with OpenSSL.

## 1.Overview

This C++ application. This allows users to Add a secret note which is encrypted using a password, View saved notes after decrypting them using the correct password and Store all notes securely in a file named 'notes.db'.



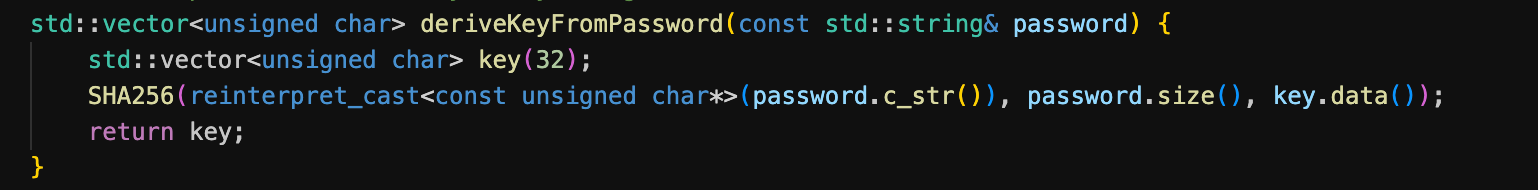
## 2. Cryptographic Setup

The application uses AES-256-CBC mode for symmetric encryption. Keys are derived from a user-provided password using SHA-256 hashing. Random Initialization Vectors (IVs) are generated using OpenSSL's RAND\_bytes for each note.

## 3. Functions

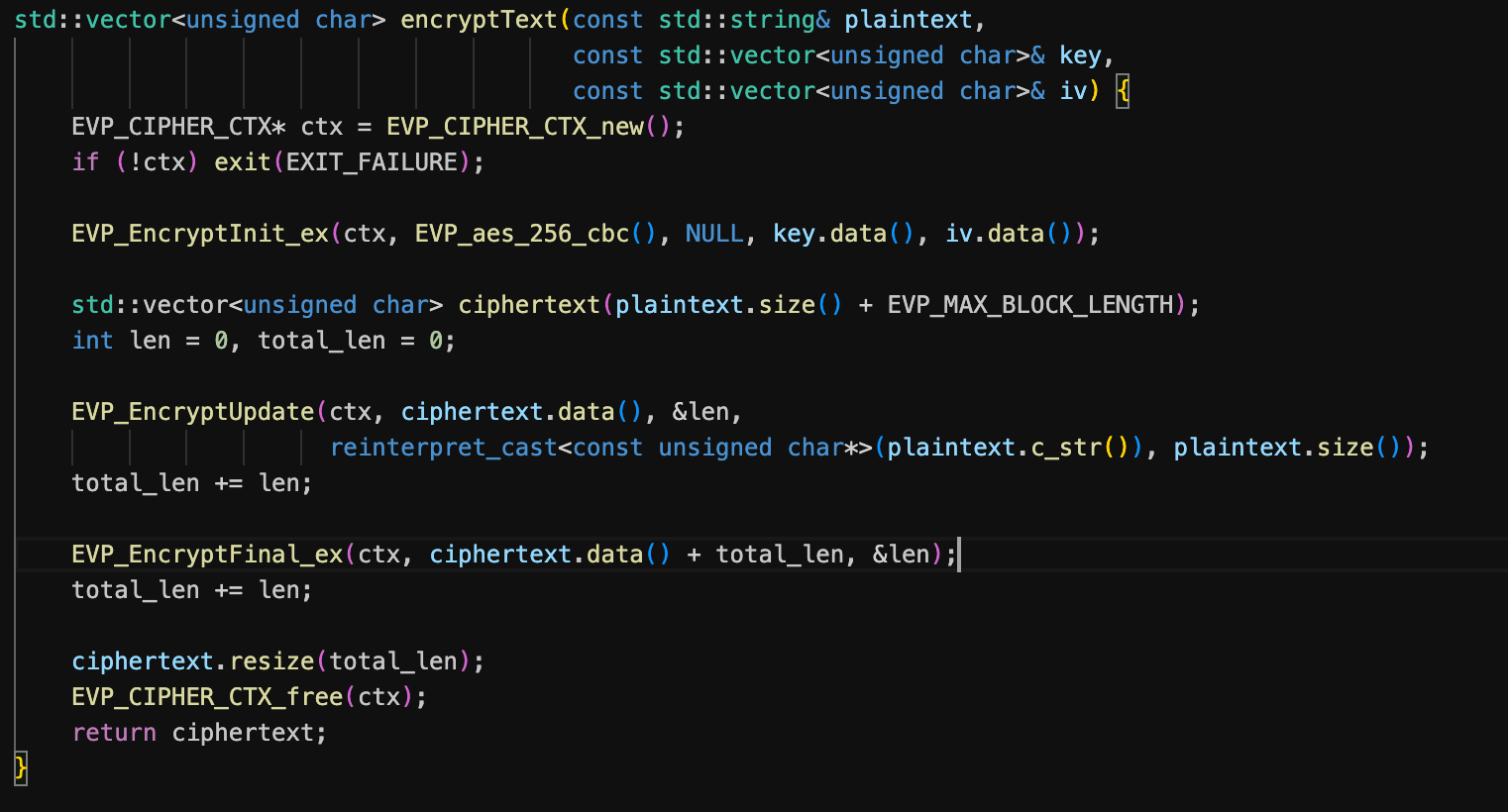
### 3.1 deriveKeyFromPassword

Generates a 256-bit key from the user's password using SHA-256.



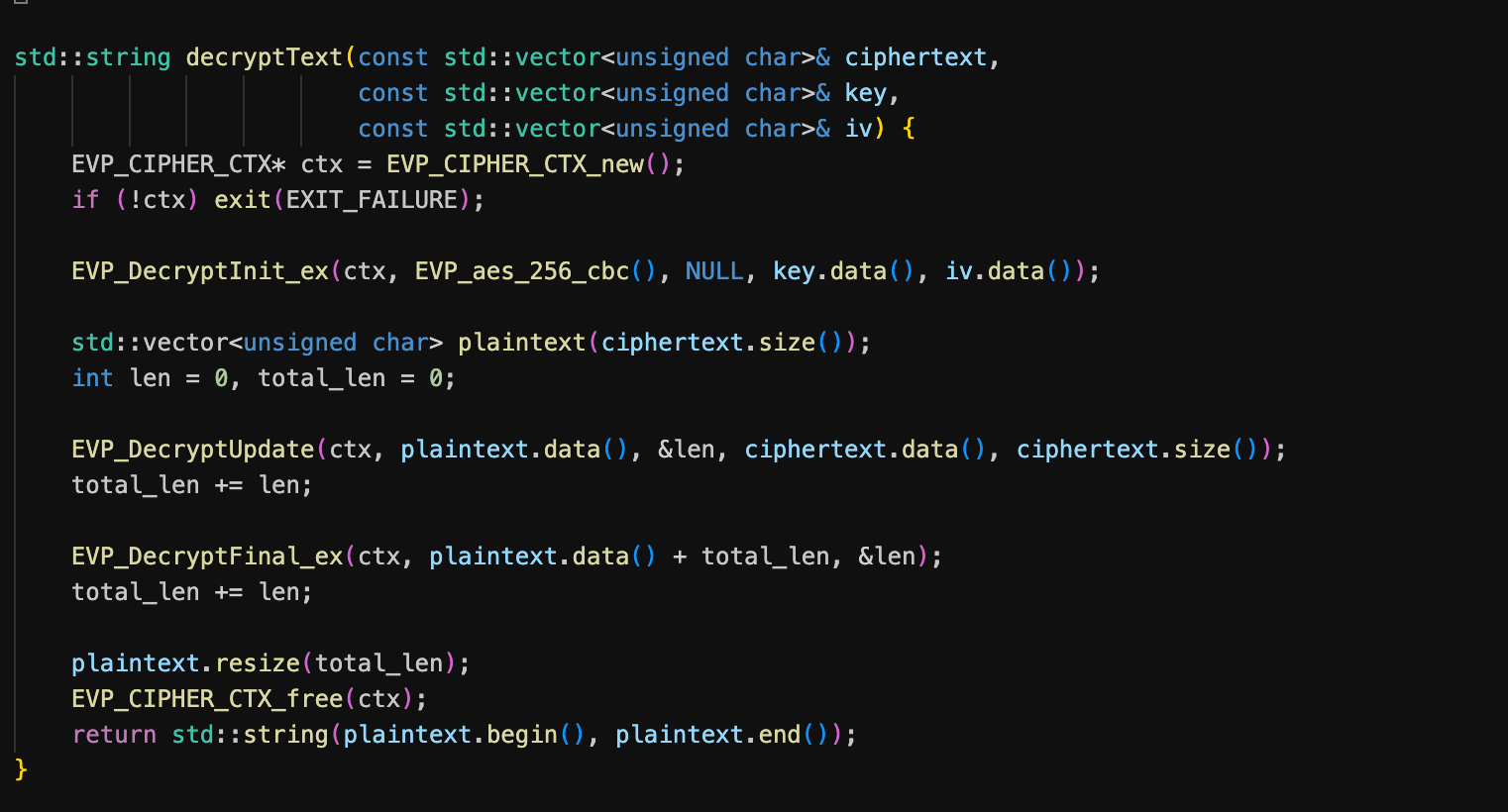
### 3.2 encryptText

Encrypts a note using the derived key and a random IV. The encrypted data and IV are stored.



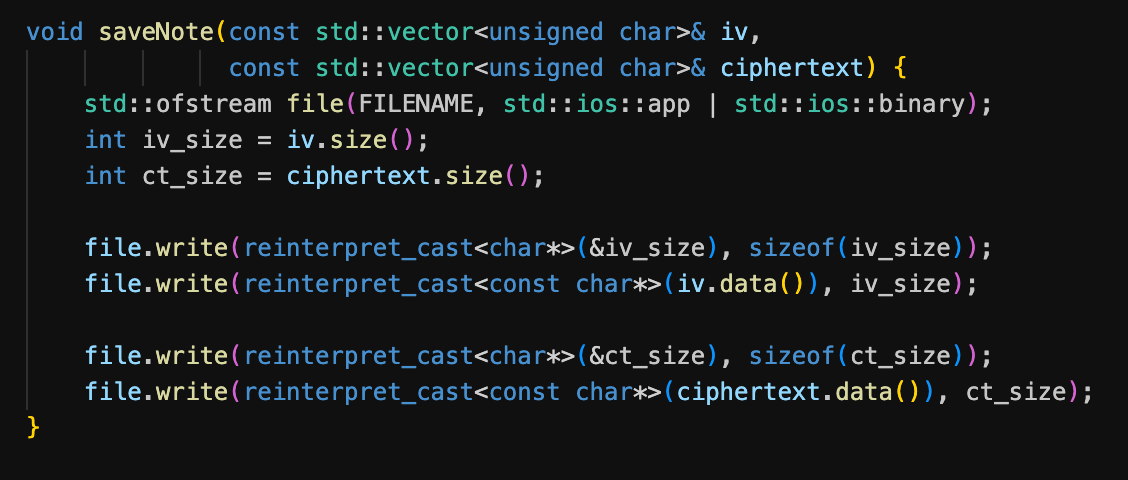
### 3.3 decryptText

Decrypts the encrypted note using the same key and the IV that was saved with it.



### 3.4 saveNote

Appends the encrypted note and IV to 'notes.db'.



### 3.5 readNotesFromFile

Reads all notes from 'notes.db' and attempts decryption using the entered password.



### 3.6 main

Main application logic with a text-based menu for users to add or view notes.



## 4. Security Considerations

* Each note is encrypted with a unique IV.
* Passwords are not stored; keys are derived from the password.
* Uses OpenSSL library for cryptographic operations.

## 5. Sample Output

Complie and Run from this Comands:

**g++ aes\_note\_vault.cpp.cpp -o aes\_notes**

**./ aes\_notes**

A screenshot of a computer

Description automatically generated