# Brief Overview of Project :

The main aim of this project was to create a new method of encryption based on the existing algorithms, but in our own way. We wanted to prepare a program that could encrypt text or text files into a secure format, which is not decryptable by hackers.

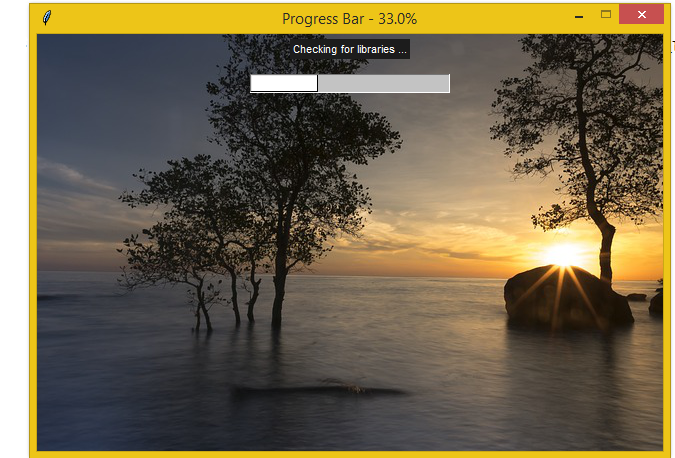
The software created by us requires a username and password to login and use our services. The password is hashed before it is stored in an online MySQL database along with the username. Once the user has logged in successfully, he/she can begin using the software.

The encryption algorithm used makes sure that the returned encrypted text is always of a fixed length, however big the input may be. This makes sure that the message’s length cannot be interpreted from the encrypted text. Moreover various techniques using matrices and random number generator have been adopted in our algorithm to make it secure.

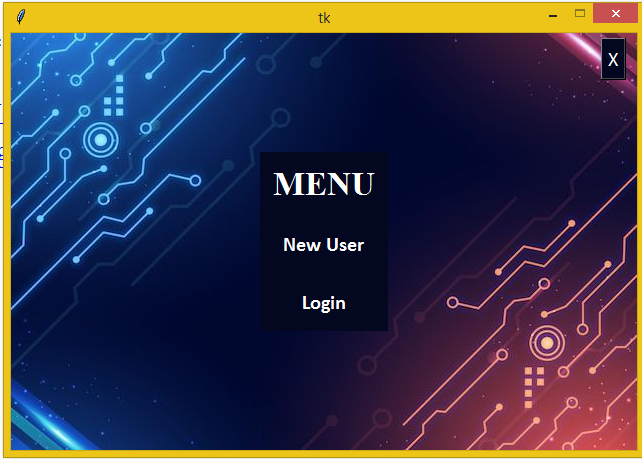
The software has a GUI based on Tkinter and also has embedded music with it. It was built on Python 3.1x and incorporates various libraries like pyperclip.

# Output:

Screen 1: Loading screen which checks if device is connected to the internet and whether required libraries are installed.



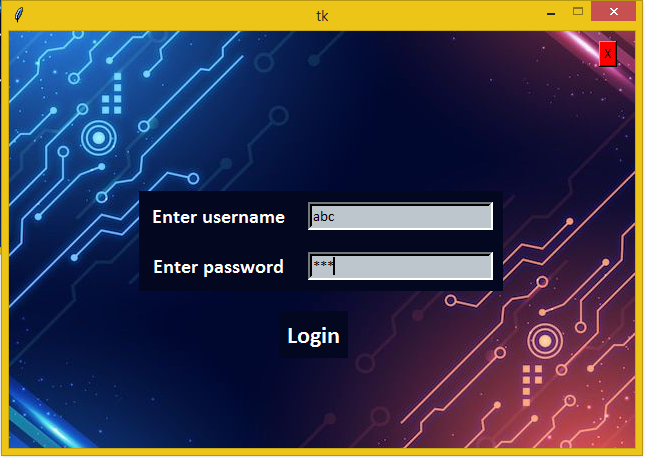
Screen 2: Login / New user signup screen.



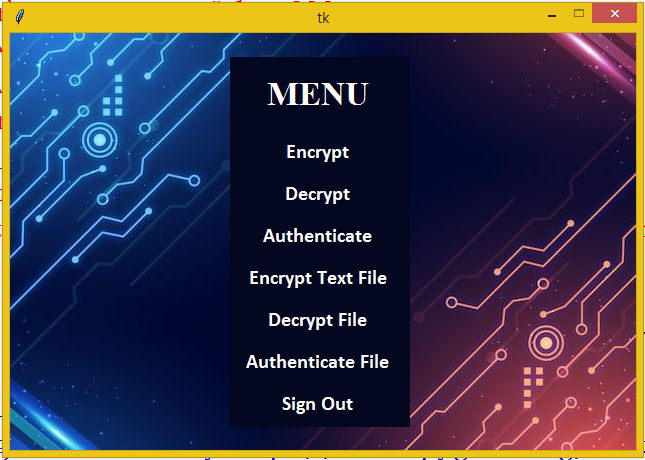
Screen 3(a): New user sign up.



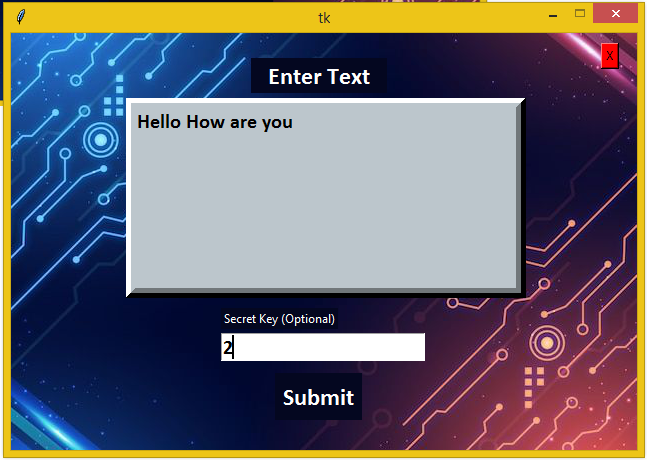
Screen 3(b): Existing user login screen.



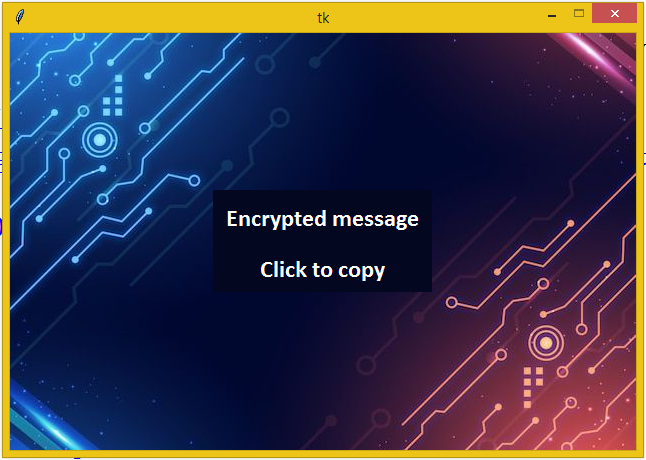
Screen 4 : Encryption option screen.



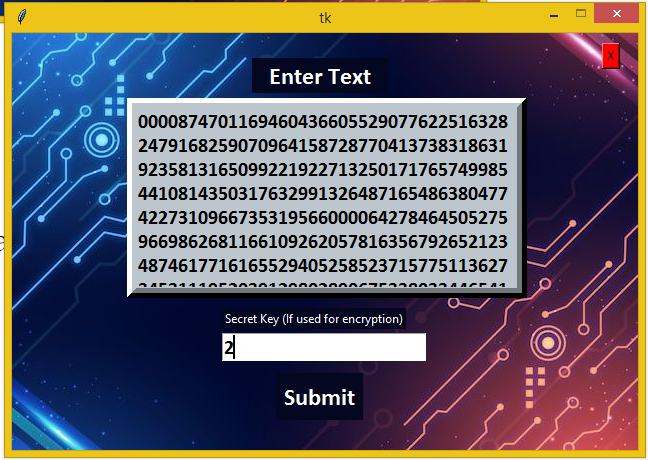
Screen 5(a) : Encrypt a message.



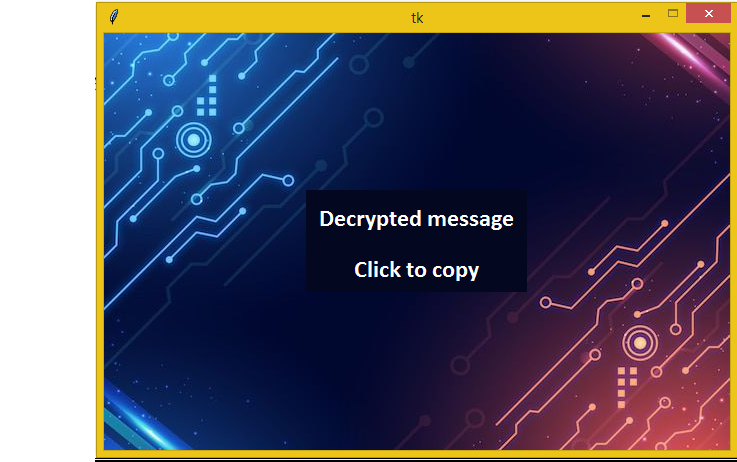
Screen 5(b): Copying the encrypted text to your clipboard.



Screen 6(a) : Decrypt a message.



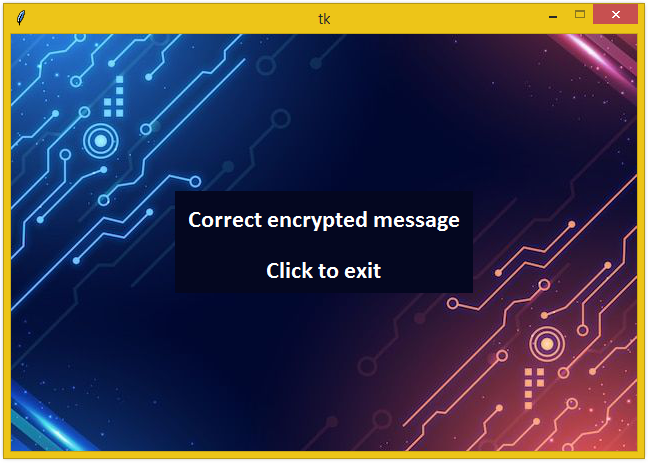
Screen 6(b): Copying the decrypted text to your clipboard.



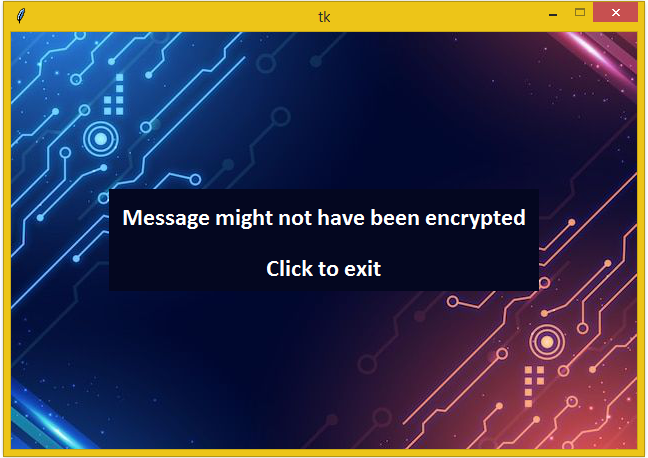
Screen 7(a): Authenticate a encrypted message.



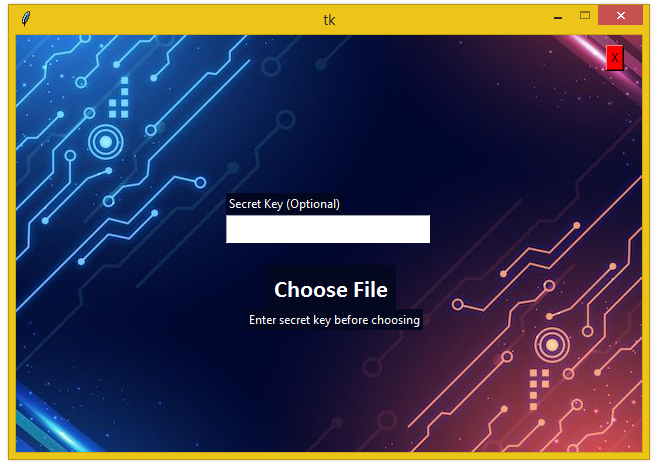
Screen 7(b): Authentication successful.



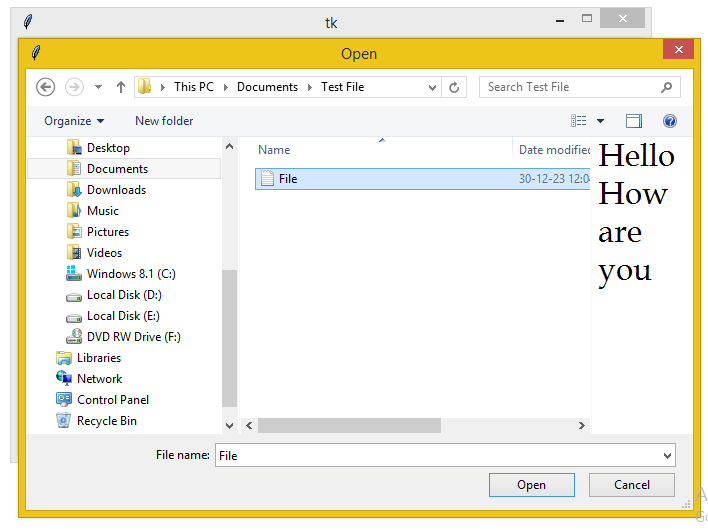
Screen 7(c): Authentication failed.



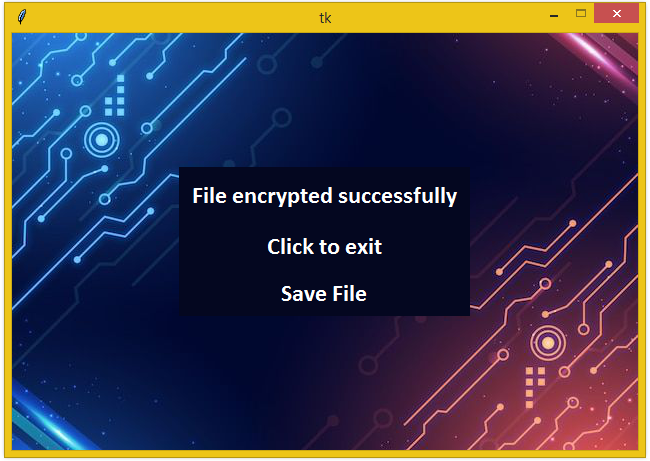
Screen 8(a): Encrypt a text file.

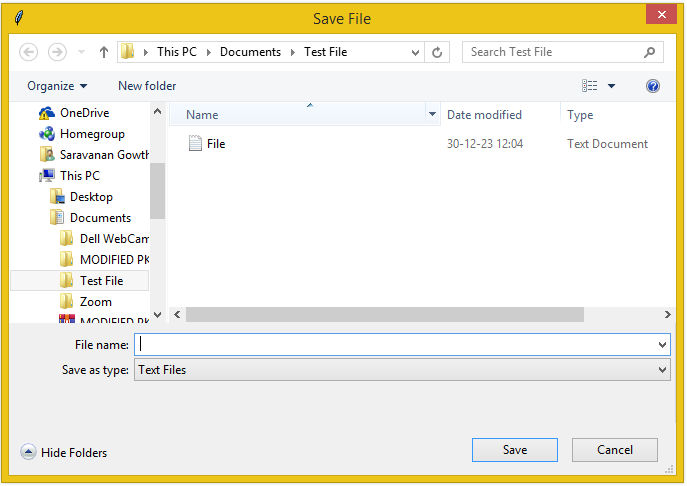


Screen 8(b): Choosing the file to encrypt.

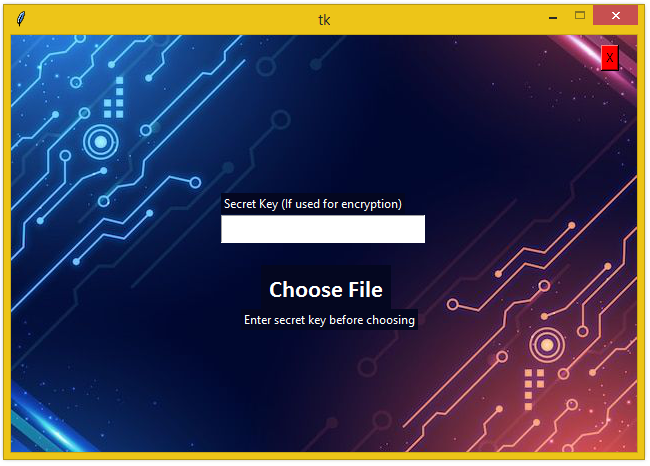


Screen 8(c): Choosing the location to save the encrypted file.

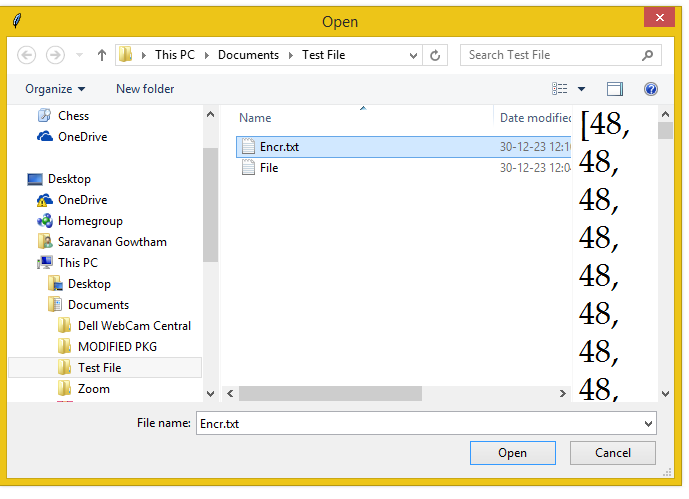




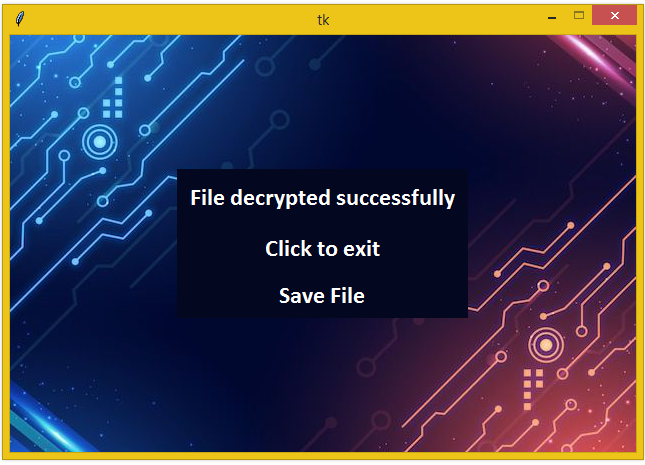
Screen 9(a): Decrypt a encrypted file.

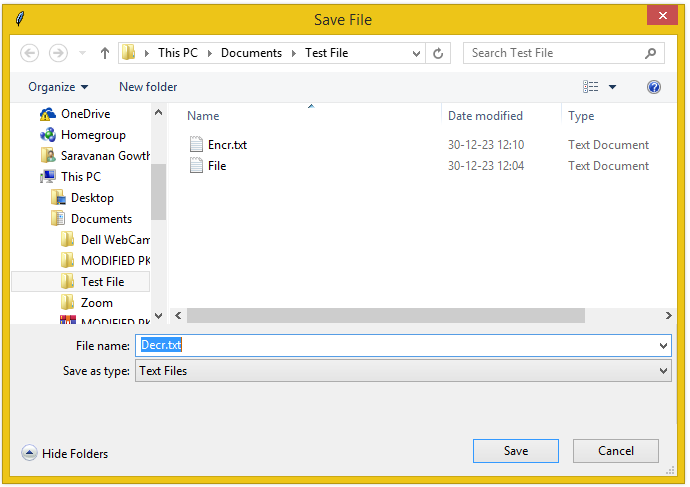


Screen 9(b): Choosing the file to decrypt.

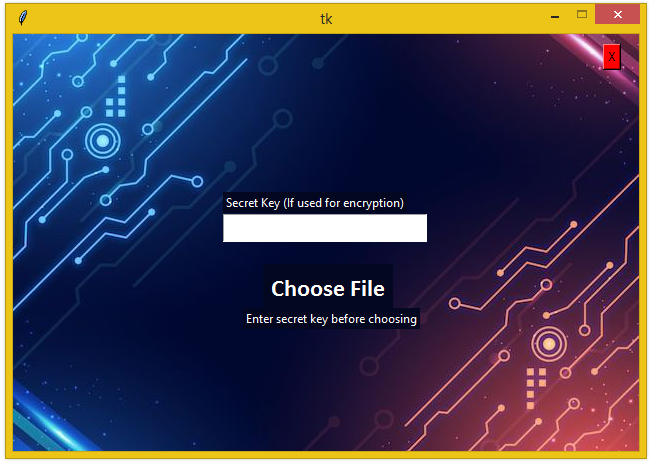


Screen 9(c): Choosing a location to save the decrypted file.

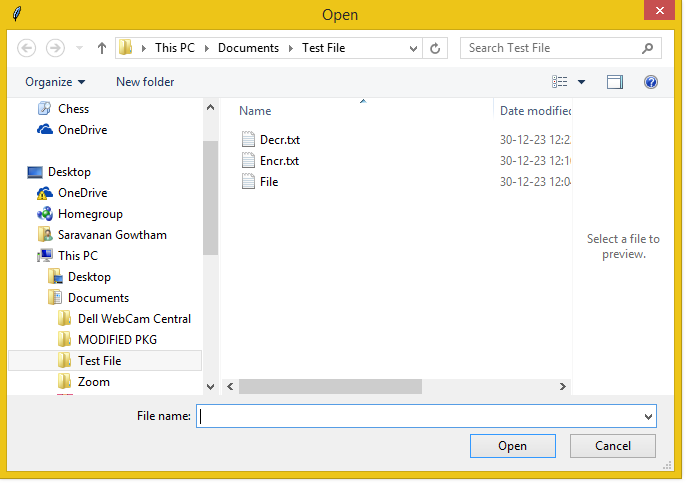




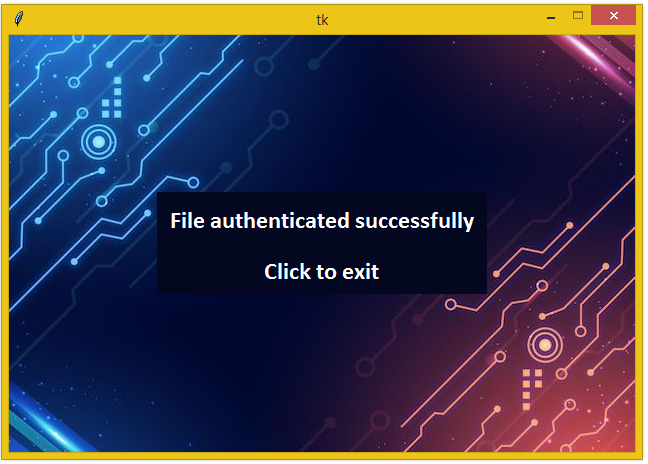
Screen 10(a): Authenticate a encrypted file.



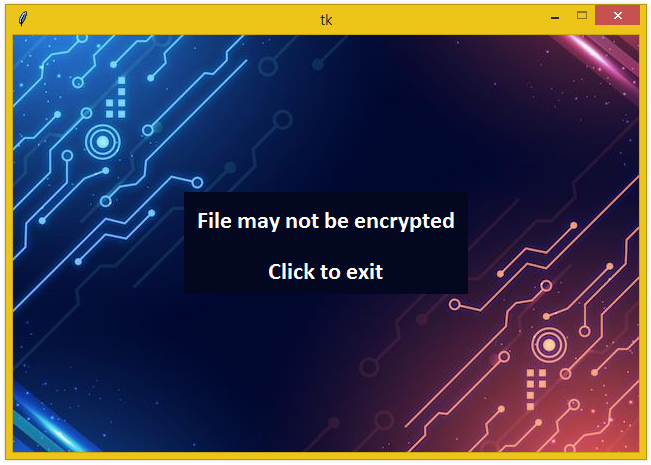
Screen 10(b): Choosing the file to authenticate.



Screen 10(c): Authentication successful.



Screen 10(d): Authentication failed.



Future Enhancements:

* The MySQL server can be upgraded to a paid version to host more data regarding the user.
* A settings menu could be developed to control the volume of the music and to delete the account of a user.
* An option to save user specific keys to the MySQL database could be implemented in the future.
* An option to use our services without login , that is, as a anonymous user could be implemented.
* A back button could be created to go back to the previous screen.
* Additional algorithms could be added to enhance security.
* Additional encryption services could be added.