AES Encrypt/Decrypt Java Application (CA2 – 30%)

This assignment should be completed in groups of two, except by prior agreement with the lecturer. This assignment is worth 30% of the overall grade for this module.

Objectives

Build an application that will encrypt and decrypt a file using AES encryption.

Task 1: Create a Menu System

Create an initial menu system that will allow the user to:

1. Encrypt a File (Task 2)
2. Decrypt a File (Task 3)
3. Quit the application

After each task is finished the menu should be presented to the user until the user specifies the quit the application menu option.

Task 2: Encrypt a File

When the user selects encrypt a file, they should be prompted to enter the filename. The application should generate a random key to encrypt the file. The application should then encrypt the file using the random key. The encrypted data should be written to a file ciphertext.txt, the key should be written to the screen along with a notice regarding the encrypted text file.

Task 3: Decrypt a File

When the user selects decrypt a file, they should be prompted to enter the filename. Then they should be prompted to enter a valid key. The application should then decrypt the file using the key. The decrypted data should be written to a file plaintext.txt and a notice regarding the location of the decrypted data should be written to the screen

Additional Restrictions and Guidance

The solution must include appropriate exception handling and validation of external input. You must gracefully handle a scenario where a user enters an invalid menu option, invalid filename, or invalid key. Your application should not terminate abruptly.

Assessment Criteria

1. Use of exception handling and validation of external input (20%)
2. Implementation of AES encryption process (15%)
3. Implementation of AES decryption process (15%)
4. **Individual** (i.e. each student – not per group) document discussion of AES best practice, guidance on key lengths, appropriate usage. Word count: 500 (+/-10%), must include references. (20%)
5. Code quality, abstraction, and reuse (20%)
6. Presentation and professionalism of terminal interface (10%)

Upload a single ZIP file (**per person**) to Moodle by Thursday, December 15th at 23.55pm, containing the following:

1. The JAVA PROJECT that contains your solution.
2. A link to a screencast demonstrating the operation of your application.
3. A document or text file containing your discussion of AES best practice (as detailed above) plus references for any work which is not your own, or has not been created specifically for this project, or has been used as part of any other deliverable. You should also include links to any third-party tutorials you follow while creating your solution.

The assignment must be entirely the work of each student group. Students are not permitted to share any pseudocode or source code from their solution with any other group in the class. Students may not distribute the source code of their solution to any other student in any format (i.e. electronic, verbal, or hardcopy transmission). Any suspected plagiarism will be investigated, pursued, and reported to the Plagiarism Committee.

The institute’s standard policy on marking of late submissions will be applied:

* Submissions received up to 1 week late will have a 20% grade reduction applied.
* Submissions received up to 2 weeks late will have the above penalty applied and will be capped at 40%.
* Submissions received more than 2 weeks late cannot be graded.