Homework 2 – Encrypted FS on FUSE (Due by **2024/12/26**)

**Objective**: This assignment aims to deepen your understanding of file system operations and encryption mechanisms by building a simple in-memory file system using the FUSE (Filesystem in Userspace) framework, followed by integrating AES-256 encryption to ensure data security.

**Resources**:

1. Less Simple, Yet Stupid Filesystem (Using FUSE): https://github.com/MaaSTaaR/LSYSFS
2. In Storage Filesystem (ISFS) Using FUSE: https://github.com/yttty/isfs

**Part 1 (15%)**: Setting Up FUSE environment: Install the FUSE library if not already installed. Use the package manager of your Linux distribution, for example, **sudo apt-get install fuse libfuse-dev**

**Part 2 (25%)**: Building a Basic In-Memory File System with FUSE: Using the FUSE framework, create a simple in-memory file system. This file system should support basic operations such as:

* Create, read, and write files.
* Open and close files.
* Create and remove directories.
* List directory contents.

**Part 3 (20%)**: Integrating AES-256 Encryption: Extend your file system to encrypt file data using AES-256 encryption before writing to memory and decrypt data when reading from memory. Utilize a cryptographic library such as OpenSSL for implementing AES encryption.

**Part 4 (10%)**: Encryption Key Management: Implement a mechanism for managing encryption keys, ensuring that each file can be encrypted with a *different key*. Design the system so that the encryption key must be supplied to open a file.

**Part 5 (10%)**: File Operations with Encryption: Ensure all file operations (read, write, etc.) handle encrypted data correctly.

**Part 6 (10%)**: Testing and Validation: Conduct comprehensive tests to verify the functionality with encrypted files. Ensure encrypted files are unreadable without the correct decryption key and readable with it.

**Part 7 (10%)**: Demonstration and Presentation: Make appointment with TA to demonstrate your developed encrypted FS on FUSE. You should detail the project architecture, encountered challenges, implemented solutions, and key learnings.

**Deadline**: All materials, including the report and modified code, must be uploaded to the EE-Class platform before the due date.