

ELL 783/ ELL 405

Assignment 2

DISTRIBUTED SECURE FILE SYSTEM USING FUSE

9th March 2022

1. Distributed User-Level File System

You need to implement a distributed file system (DFS) using FUSE - a tool that allows us to run file system code as a regular program (in userspace). When a FUSE program is run, it creates a file system at the mount point provided by the user and then makes its virtual file system appear within that mount point.

For this assignment, your distributed file system must support the following functions:

- Create and remove a directory.
- Create, read and write to a file.
- Delete an existing file.
- Appending to and truncating a file.
- Access, modify, and status change updates (essentially, update the times).
- Open and close a file.

The communication layer between the client and the server can be implemented using the following protocols: TCP, UDP, and RPC. Also, the data stored in the file system should be **encrypted using public-key cryptography (RSA)**.

You have to create more than two client machines and a server machine. The server machine stores the files created by the client machines. The files are stored in encrypted form. Whenever a client wants to access the file's content, it first needs to decrypt the files then perform the specific operation. The file has the following permissions i.e., user-level and group level. For example: if client 1 creates a file that has read, write permissions for the user (client 1) and read-only permission for the group (i.e., other clients on the network). Then other group machines will not be able to write or execute the file created by client 1. You will have to implement all such types of scenarios in which a client can read/write files created by

himself/another client. The constraint is that only the owner of the file will be able to delete that file.

Encryption and decryption scenario:-

Assume that a particular group of users can access each file. They have already established a group key using the Diffie Hellman key exchange protocol. Each user has a table in which he had stored two information, i.e., Group ID and group key. The communication between user and server is secured using RSA protocol.

1.1 Submitting Your Assignment

- Create a folder named <EntryNumber>, containing the modified files. Compress the folder and submit only the <EntryNumber>.zip archive on Moodle. For example, if your entry number is 2017EE51010, then submit as 2017EE51010.zip. No other format will be accepted.

To summarize, you have to do the following steps:

1. Create a folder named <EntryNumber>.
2. Copy the files into this folder.
3. Create the .zip file
 - For Windows: Select the files, right-click and point to "Send to" and then select the Compressed (zipped) folder. Name the folder as <EntryNumber>.zip
 - For Linux: Select the files, right-click and point to "Compress" and then create the zipped folder with the name <EntryNumber>.zip
4. Submit this zip file on Moodle.
 - **The deadline is 4th April 2022, 23:59 hours.**
 - **This assignment has to be done in groups of 2, and you will be evaluated based on the demos.**

1.2 Grading Scheme

1. Marks division: Total = 50 marks
 - Correct implementation of DFS: 35 marks
 - Correct implementation of encryption-decryption : 15 marks

2. Late policy: 20% penalty per day (rounded to the next day, i.e., 3 hours late implies 1 day late).

3. Plagiarism: We will run MOSS over all submissions. If any similarity to any other source is found, you will get a zero.