Project for AA2021-2022

Computer Systems and Programming

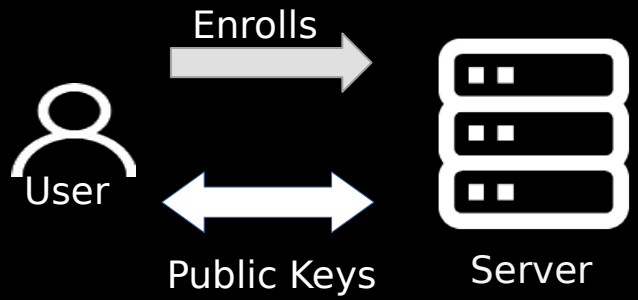
Project 2021-2022

- The purpose of the project is to write a simple “Secure Document Store”
- When a client “enrolls” with the service for the first time, the client generates a public/private keypair and exchange them with the server
- Files uploaded to and downloaded from the server are encrypted using public keys

Project 2021-2022

- A client can:
  - Register (and generate PubK/PrvK)
  - Login
  - Upload a new file, encrypting it with the server’s PubK.
  - Delete a file, if owned by himself
  - List which files are available for downloading
  - Request one file. The server will then:
    - Decrypt the file using his PrvK
    - Re-encrypt it using the PubK the requester.
    - Send the encrypted file to requester.

Example

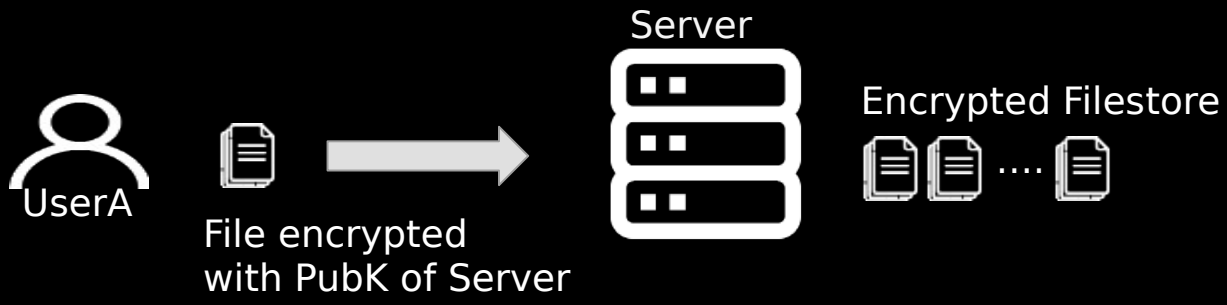


Generate:

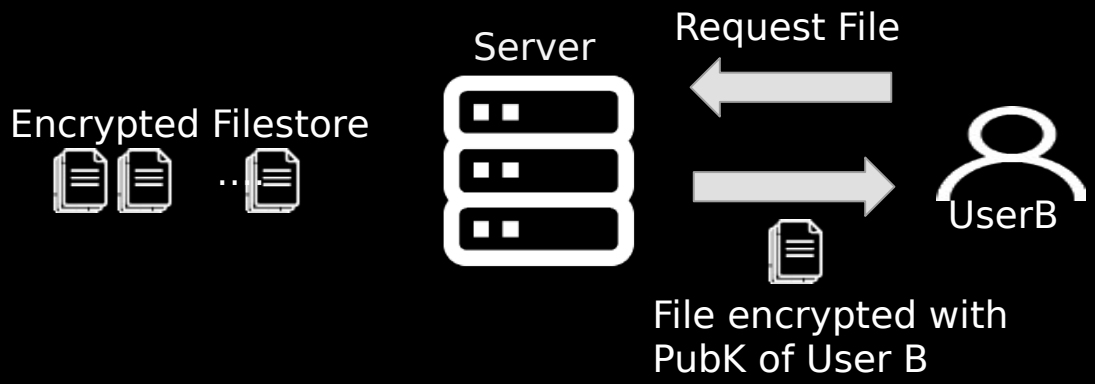
- Private Key
- Public Key

- Keeps PrvK
- Exchange PubK

Example


Server
Encrypted Filestore
UserA
File encrypted
with PubK of Server

Example


Server
Request File
Encrypted Filestore
UserB
File encrypted with
PubK of User B

Architecture

- The server must be implemented by means of multiple independent/concurrent processes sharing common data structures and coordinating through atomic primitives.
- E.g:
  - One process accepts a client request to upload files, and stores them in the shared data structure memory buffer.
  - Another process gets requests to download files and performs decryption+encryption.
- Communication between clients and server processes must be implemented using sockets.
  - Depending on how you implement the the client, there could be multiple options, e.g:
    - A master/slave architecture (a single process forks childs to perform appropriate actions)
    - Use different port numbers to connect directly to permanent processes which perform specific functions (login/admin, sender, receiver)
- Consider contentions: i.e. a client uploads a newer version of a file, or deletes it, while the same file is being downloaded by another client.

More

- OpenSSL (www.openssl.org), included in all major Linux distribution, provides libraries and tools to:
  - Generate key pairs
  - Handle encryption and decryption of files and buffers
- Don’t reinvent the wheel
- Start defining:
  - the overall architecture
  - the flow of control&data requests
  - First document and then code