Secure File Transfer Procotol (SFTP) Using Python

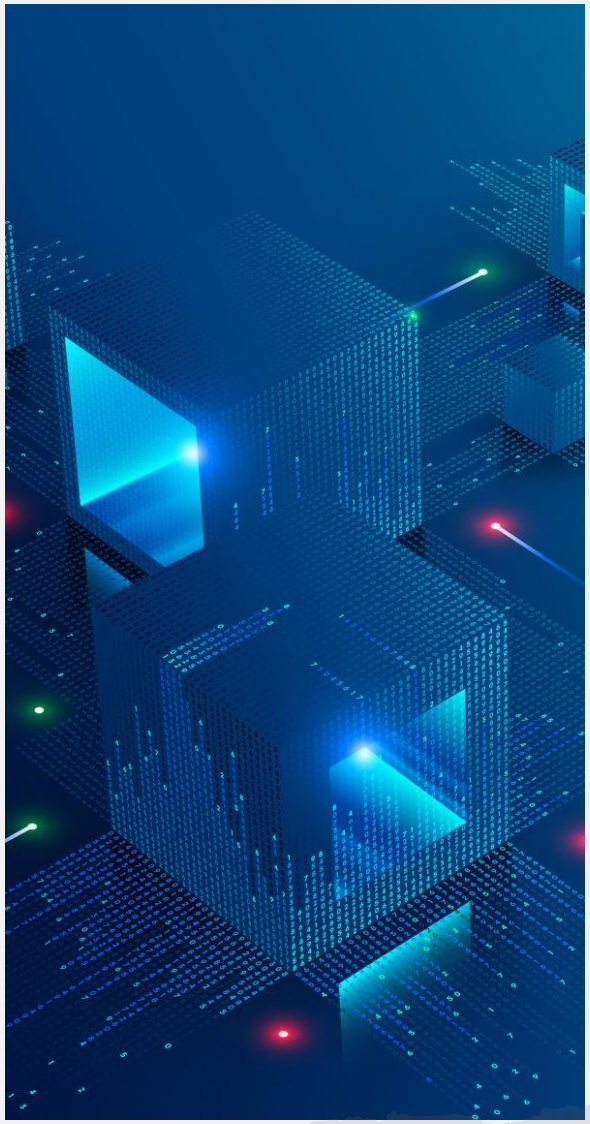
CYSE-250 Basic Cybersecurity Programming and Networking

My Nguyen

11/17/24

Project Overview

- This project aims to build a secure file transfer system using Python.
- The system will use socket programming for communication and encryption to secure files.



Key Features

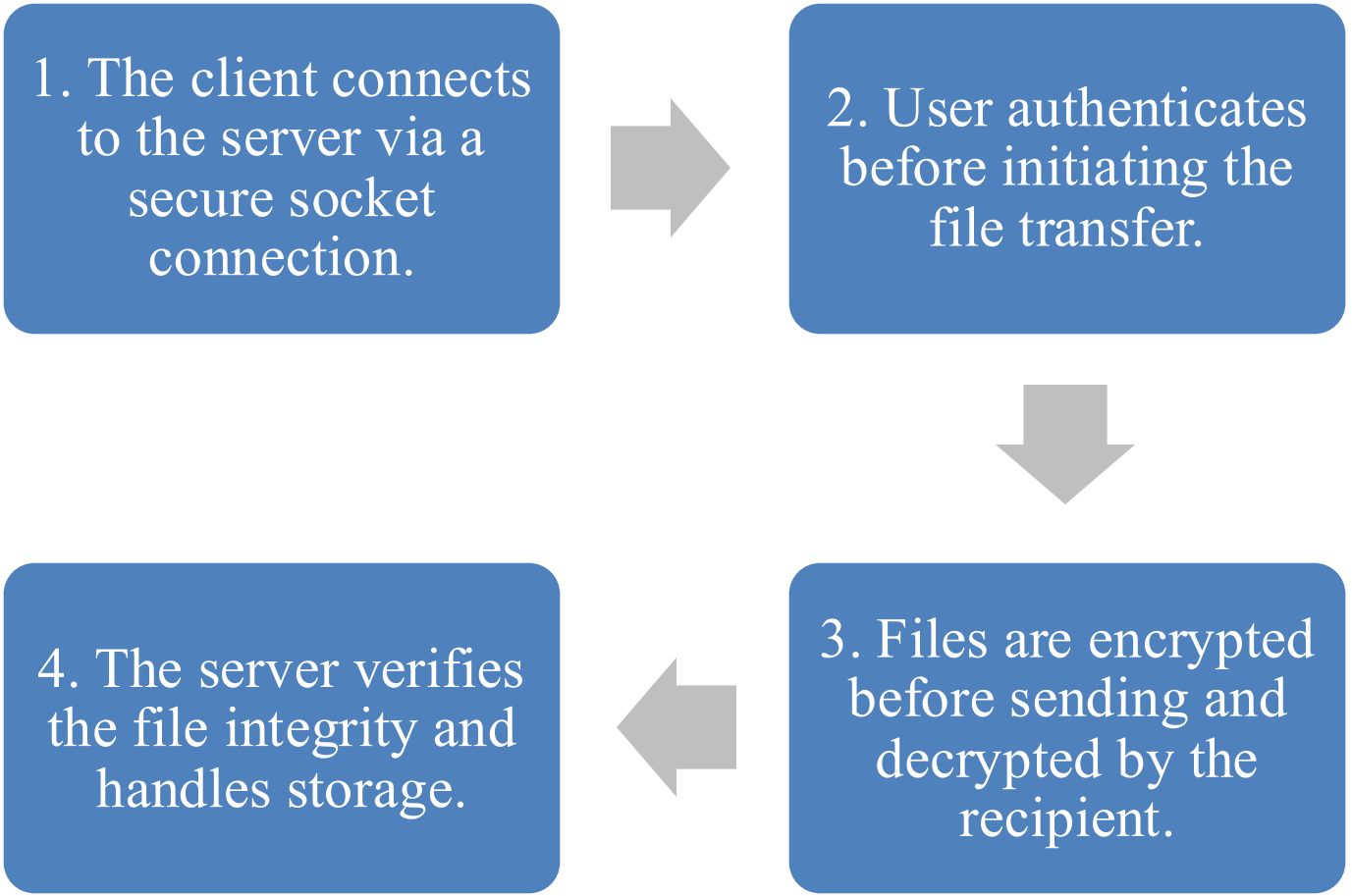
1. **Socket Programming**: Real-time communication between the client and server.

2. **Encryption**: Files are encrypted using AES or RSA algorithms for secure transfer.

3. **Authentication**: Users must authenticate to ensure secure access to file transfer.

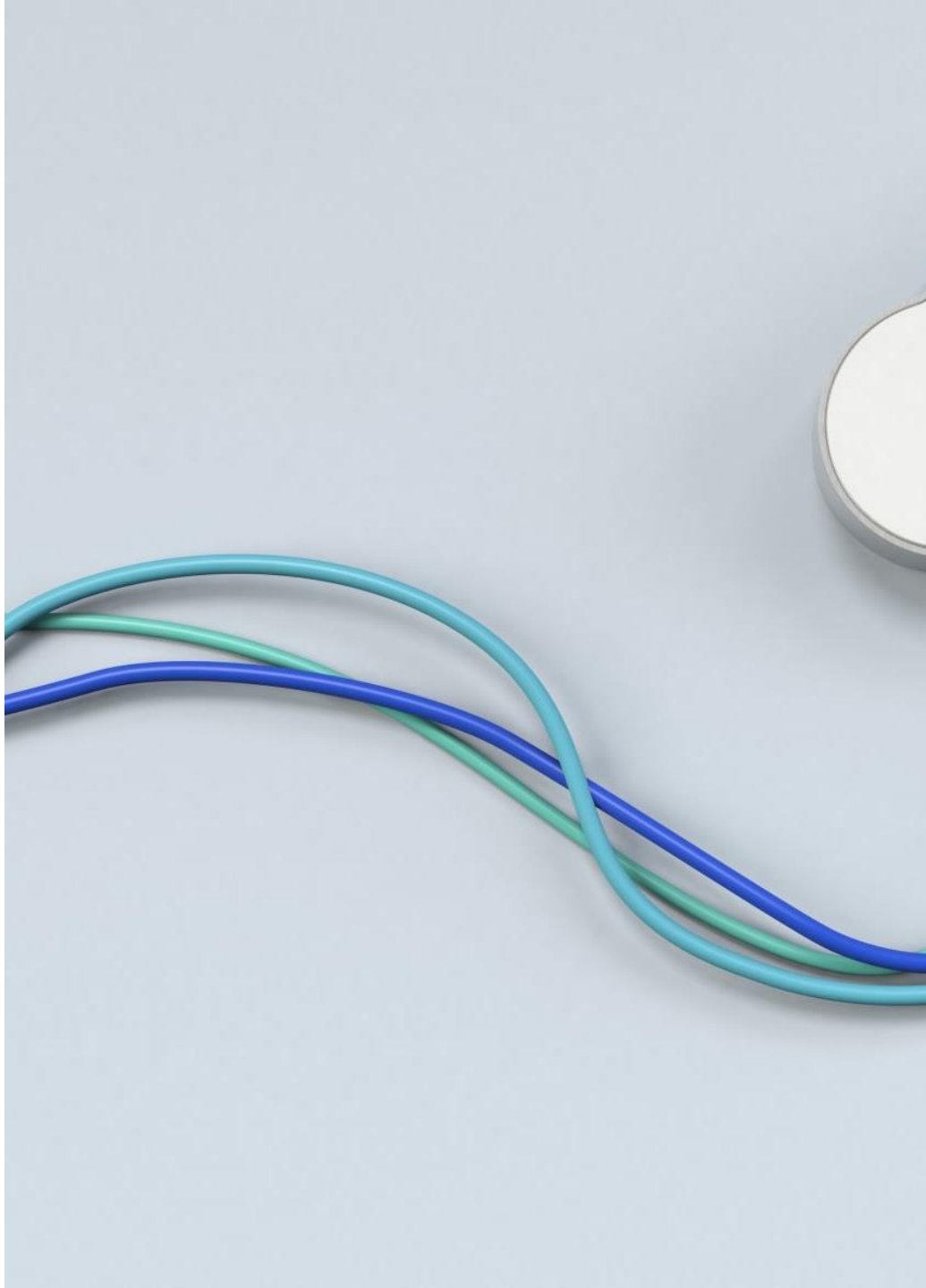
4. **File Transfer**: Allows sending and receiving files securely.

System Design



Project Workflow

- The client connects to the server.
- The client authenticates via username and password.
- The client selects a file to transfer, which is encrypted.
- The server receives the encrypted file, decrypts, and stores it.
- The server sends confirmation back to the client.



Benefits of the Project

- Demonstrates strong cybersecurity practices in file transfer.
- Provides hands-on experience with encryption and socket programming.
- Secure handling of sensitive files through encryption and authentication.

Challenges

Implementing	Implementing strong encryption for both file and communication security.
Handling	Handling file integrity and ensuring no corruption during transfer.
Managing	Managing errors and retries in case of failed transfers.

Conclusion

- This project enables secure file transfer with a focus on encryption and authentication.
- It showcases real-world security practices and socket programming implementation.