실행 절차 매뉴얼

김태은

1) 사전 준비

- Python 환경 확인 : Python 3.10 이상이 설치되어 있어야 한다.

- 필요 라이브러리 확인 : tkinter와 cryptography는 기본 내장

2) 서버 실행

서버 프로그램 실행 (터미널 입력하기)

> python server_ui.py

서버 GUI 창이 열리면, 상태 표시줄에 Listening on port 5000문구가 표시된다.

이는 서버가 클라이언트 연결을 대기 중임을 의미한다.

서버는 클라이언트의 접속 요청을 받으면(아래 클라이언트 실행) 자동으로

자신의 RSA 공개키(server_public.pem)를 전송한다.

이후 클라이언트로부터 암호화된 AES 세션키를 수신 후 복호화한다.

3) 클라이언트 실행

클라이언트 프로그램 실행 (터미널 입력하기)

> python client_ui.py

연결 창 (Connect Window)이 나타나면 아래 정보를 입력한다.

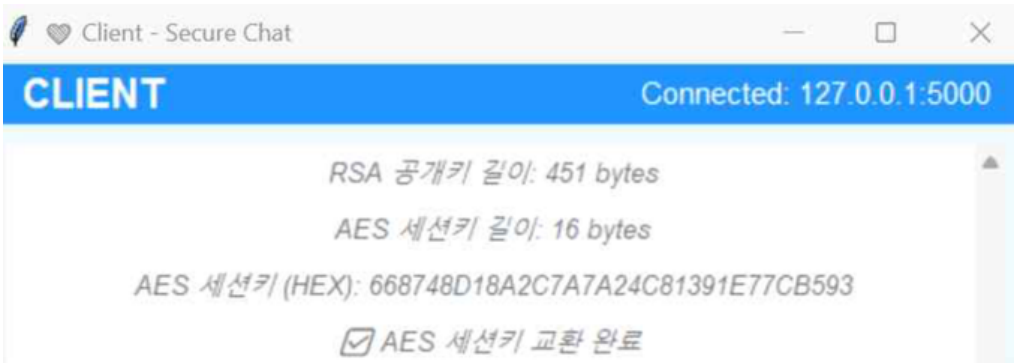
- Server IP:127.0.0.1(로컬 테스트 시 기본값)

- Port:5000

Connect 버튼 클릭

→ 서버와 연결 성공 시 서버와 클라이언트의 상태 표시줄에 Connected: 127.0.0.1:5000 표시

이후 AES 세션키 생성 및 RSA 공개키 암호화 후 서버로 전송, 채팅창 메시지 출력



4) 채팅 기능 테스트

클라이언트 입력창에 임의의 메시지를 입력하고 Send 버튼을 클릭한다.

평문은 AES로 암호화하여 전송되고, 송신 암호문은 HEX 형태로 표시된다.

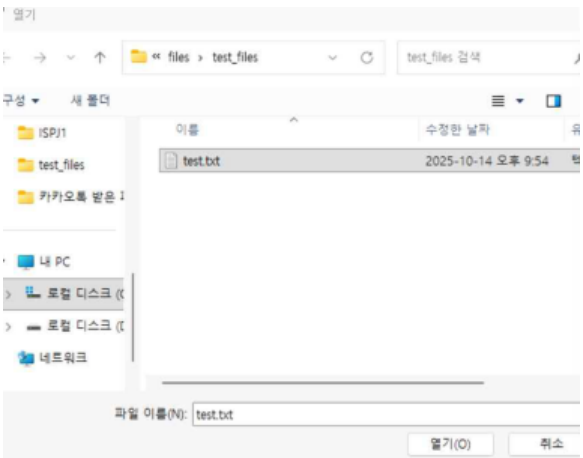
서버 측 화면에도 암호문과 복호화 결과가 채팅창에 출력된다.

(서버와 클라이언트 모두 메시지 송수신 가능)

5) 파일 전송 기능 테스트

클라이언트 하단의 클립버을 클릭하고 files/test_files/폴더 내 파일을 선택한다.

( 클릭)



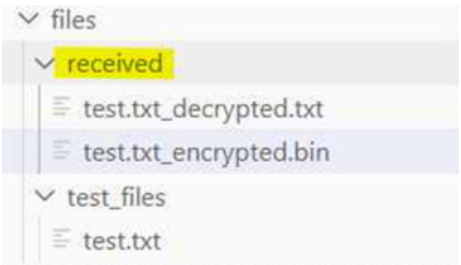
선택 후 자동으로 다음 순서가 수행된다.

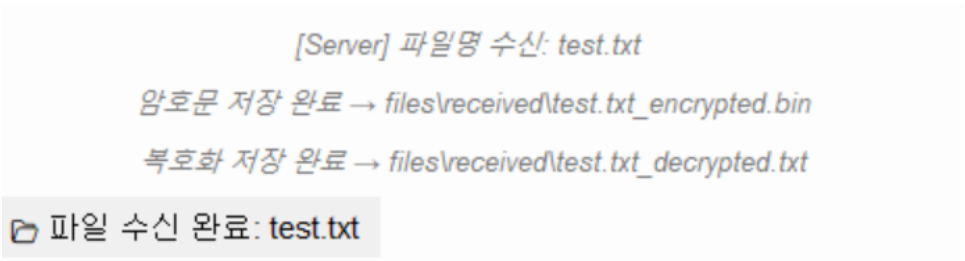
- 클라이언트: 파일을 AES로 암호화

- 암호화된 데이터를 서버로 전송

- 채팅창에 다음 메시지 표시 → 파일 전송 완료: test.txt

서버 측에서는 동일 AES 키로 복호화가 수행되어 files/received/폴더 내에 암호화된 파일과 복호화된 파일이 자동 저장된다.





서버 채팅창에서 → 파일 수신 완료:test.txt 메시지를 확인할 수 있다.