Final Exam

Instructions:

- You are allowed to consult anyone and anything other than other students in this course. If you use a paid, service, however (which I do not recommend), you must mention this clearly in your reply.
- Feel free to use LLMs or any other online aid.
- I don't expect perfect solutions: I want to see your thought processes clearly, as well as your creativity. If a particular problem is not solvable using cryptography, say so - isolate that bit and do the rest!
- You can score a maximum of 300 points. 150 from this exam, and the remaining 150 from the viva that will follow. The viva will be arranged at a mutually agreeable time after you submit.
- Submit your solutions on the classroom by the due date. Don't write too much; bullet points are fine!

Advice:

- **Do not spend too much time on any single problem.** Read them all first, and attack them in the order that allows you to make the most progress.
- Make sure you define your functionalities and assumptions clearly. We have discussed this in detail in class, but make sure you walk the talk!

Question	Points
Abuse Reporting	50
File Hosting	50
Payment Systems	50
Total:	150

Name: ______________________ E-mail: ______________________

Problem 1. [50 points] **Abuse Reporting** (1 part)
Consider any end-to-end encrypted messaging system (including E2EE email etc.). We want to create a mechanism for reporting abuse to a law enforcement (LE) server, but want to allow the reporter to remain anonymous. The reporter must get a receipt which they can later use in court to follow up on the case.

Simultaneously, LE has limited resources and wants to prioritise cases which have many reports, so they set a threshold - they will only investigate cases with at last k reports. This raises two problems: one, how can we prevent LE from reading a message until they get k reports, and two, how to prevent the same (anonymous) person from repeatedly reporting the same message?

Problem 2. [50 points] **File Hosting** (1 part)
We want to have an encrypted, anonymous file hosting service. Any user can sign up, pay some money, and the upload a file, which will be made available for download by anyone for a period of time (based on the payment).

Describe the minimal assumptions you think are needed to have enable a malicious server. You may assume a somewhat secure setup, i.e., that the server is semi-honest up to (and including) the payment (as is the user performing the upload). After that time, everyone can be malicious.

Problem 3. [50 points] **Payment Systems** (1 part)
Consider UPI: we want to add a receipt mechanism, so that both sides get ‘proof’ that the transaction was successful.

When you make a payment to a merchant, consider the problem of receiving an acknowledgement - and the merchant then knowing that you have received that ack - and so on. How can we mitigate this? Can we do this without involving the server (on UPI lite) as well?