Project 3: Topic 2

PKI

Luca Campa

Department of Computer Science
Universität Innsbruck

May 30, 2025

Topics

- Introduction
- Task Description
- Learning Objective
- Dates
- Evaluation

Introduction

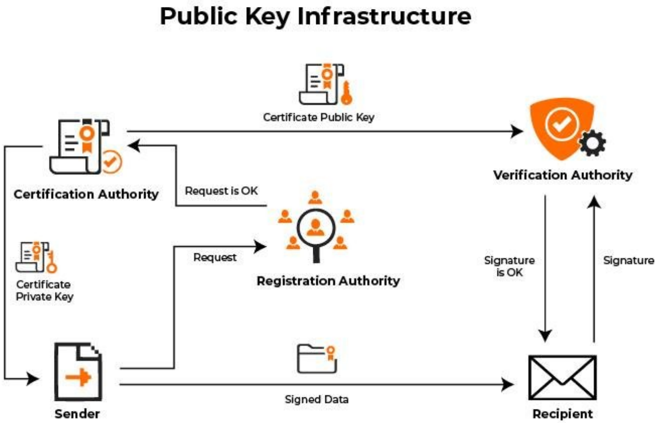
The Public Key Infrastructure (PKI) is a set of entities and procedures needed to create, distribute, revoke, store public information (namely digital certificates and public keys) of communication endpoints.

The idea is that:

- Each identity can request the creation of a public key - private key pair to the Certificate Authority (CA)
- The public key will be stored in a publicly available database
- The digital certificate will contain or the public key or the private key. The one containing the private key is sent back to the applicant, the one containing the public key is stored into the publicly available database. Both certificates are signed by the issuer (the CA).
- Each identity can ask for revoking its data.
- If compromised, the certificates are revoked.
- Each certificate contains an expiration date (usually short, except for the CAs).

Introduction

The following diagram depicts what has been previously described:



Task description

Using Python and the encryption systems we have discussed, develop your PKI. More specifically, you must create a CA and the publicly available database. Then, I should be able to register, get my keys and send a signed message (a couple message - signature) to another party who will be able to recover my public key from the database and verify the signature.

CA

- Accept registration requests
- Generates private-public keys
- Stores public key in the public database
- Sends the private key to the applicant (the private key must be signed by the CA and the user, upon reception, must check that signature before using it).

Task description

You can simulate different entities by using Docker containers or using different tabs of your terminal. Please, describe carefully the procedure to run your project.

Task description

Cryptographic requirements

- Use RSA or ElGamal signatures.
- Certificates must contain the necessary fields. (I'll let you decide. Fields can provide different information, so be careful. **Justify your choices**.) To store the keys use the standard method shown during the lecture.
- Certificates must be in PEM or DER format.
- (OPTIONAL +3 points) Create the CRL (Certificate Revocation List). Why is it needed? What is the potential threat if no revocation list is used?

(Please, email me if you have doubts about those points)

Task description

Your project must be structures in the following way:

- it must contain a `src` folder where the tool will be located
- it must contain a `README.md` file that contains the detailed usage instructions with a detailed justification of the implementation choices.

Learning objective

- Understanding how to develop a simple PKI.
- Understanding how to parse certificates.
- Understanding digital signatures and their usage.

Dates

- **Deadline for submission**: 24 June at 23:59
- **Oral presentation**: 26 June 2025 (After the Quiz)

Evaluation

The project will count as the 20% of the final grade, in particular:

- 10% Implementation:
  - 2% if the code is running
  - 8% if the code returns correct results on all the test inputs (we will provide you with some of them, but not all)
- 10% Oral presentation: every member of the group must explain one of the parts of the project.

Bonus points (OPTIONAL items) will be taken into account for the final mark of the project. If you obtain more than 20 points, the additional points will be taken into account for the final mark of the course.
It is suggested, but not mandatory, to do the project in groups of **maximum 4** students.