TTP SmartCard-based ElGamal Cryptosystem using Threshold Scheme for Electronic Elections

Authors: Jordi Pujol-Ahulló, **Roger Jardí-Cedó**, Jordi Castellà-Roca

UNESCO Chair in Data Privacy

Universitat Rovira i Virgili (Spain)

Department of Computer Science Eng. & Maths

Summary

- What? Goals
- Why? Motivation
- Which? Protocol
- How? Implementation
- Results
- Conclusions and Future Work

What We Want? Goals

- Implementation (secure and efficient):
  - Multi-authority election scheme [Cramer, Genero and Schoenmakers, 97]
    - ElGamal cryptosystem
    - Shamir's Secret Sharing Scheme
- Using:
  - JavaCards (SmartCard)

Motivation

- Because…
  - Electronic elections employ typically asymmetric cryptosystems to encrypt votes.
  - The corresponding secret key is a sensible piece of information.
- We should…
  - Enforce the security (secrecy and anonymity) of the eVoting systems.
- By…
  - Distributing the election key (private) into several shares.
  - The secret sharing schemes are widely used in electronic elections.

Motivation

- In practice…
  - …ElGamal cryptosystem is widely used to **encrypt** the votes
  - …JavaCard (smartcard) is used to **enhance security** and **usability**
    - Smart-cards are Tamper-proof devices
    - They make easier the shares portability
  - …The JavaCard API gives no support for ElGamal although smartcard HW may give support!
- So, there is no implementation combining
  - JavaCard, ElGamal and Threshold Scheme

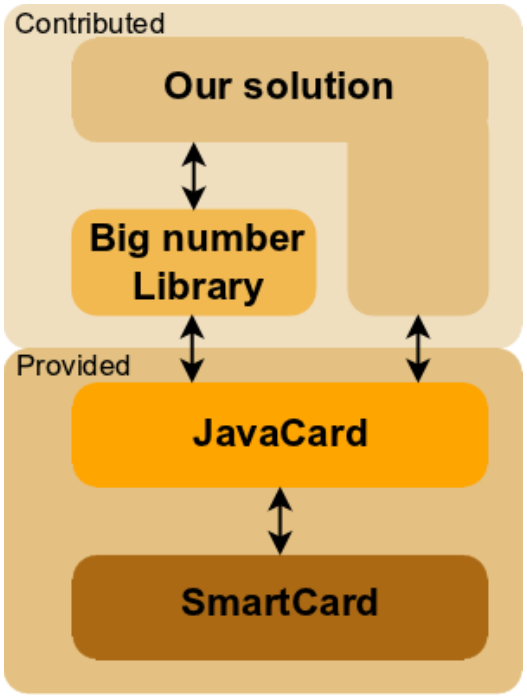
Protocol

- Electoral board generation:
  1. Election of the electoral board (according to current law).
  2. ElGamal public key and shares generation according to Shamir proposal (from one out of the smartcards – president of electoral board).
  3. **Shares distribution**
     1. **Create a secure channel between the smart-cards**
     2. **Send each share to the corresponding smart-card**
  4. **Verification and storage of the shares**
     1. **Every smartcard has only its share.**
- Voting:
  5. Votes encryption according to the corresponding protocol using the ElGamal public key.
- Tally:
  6. **Distributed decryption of the votes, without reconstructing the private key.**

Implementation

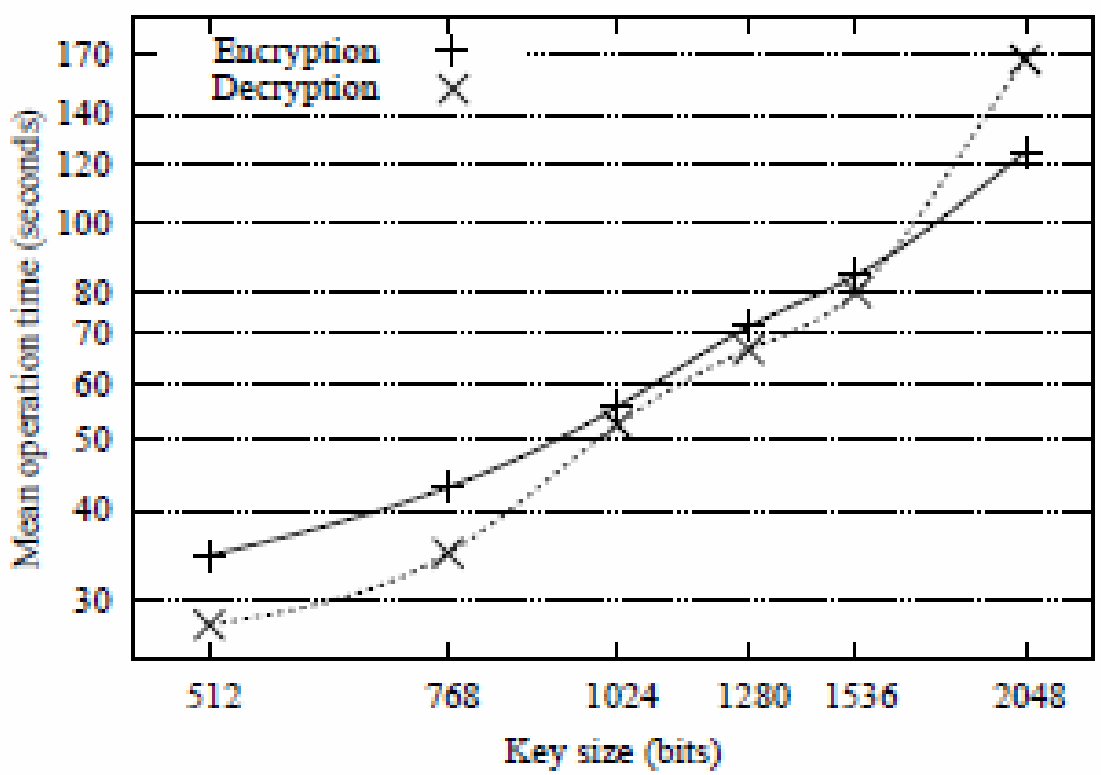
We have implemented this protocol defining…

- Big number library
  - Offers:
    - Big Number Storage
    - Modular arithmetic
      - ModPow via HW
      - ModMul via Binomial theorem (using ModPow)
- ElGamal API
  - Uses the Big number library
  - Offers encryption/decryption functionalities
- Multi-authority election scheme API
  - Uses Big number librtary and ElGamal API functionalities
  - Offers shares generation and private key reconstruction functionalities



Results

- Using JCOP v.2.2 72Kb of EEPROM memory



- Bear in mind that the results belong to the encryption and decryption operation

Conclusions and Future Work

- Efficiency
  - The SmardCard HW is used as much as possible to accelerate the modular operations the costs are acceptable
    - Some operations of the protocol are executed in the pc (i.e. Lagrange coefficients)
- Security
  - JavaCard is a tamper-proof device
    - Secure operations and communication channel

- As a future work, we are working in a Non-TTP solution with a distributed generations of the shares.

TTP SmartCard-based ElGamal Cryptosystem using Threshold Scheme for Electronic Elections

Thank you for your attention!

Any question?