CRYE 6138 Security Models

Jorge L. Villar

UBa Cyber Crypto Center, Fall 2025

Introduction

Outline

Contents

Approximate timing (12 hours in total):

- Introduction (1 hour)
- Computational Security (1 hour)
- Security Models for Encryption (4 hours)
- Security Models for Signatures (2 hours)
- Security Assumptions and Reductions (2 hours)
- Security Proof Techniques (2 hours)

Organization

Lectures

On-line two-hours sessions with a middle short break

- **Deliverables:** Some problems to be solved (50% of the grade)
- **Final Exam** (50% of the grade)

Contact information for further assessment

By e-mail to `jorge.villar@upc.edu`

Study Material

Professor's webpage: `https://web.mat.upc.edu/jorge.villar/course_secmodels.html`

More public resources in other sections of the same webpage

Jorge L. Villar's webpage

Home | Publications | CV | Teaching

Teaching

In the current semester (Fall 2025), I'm teaching **Probability and Statistics** for Physics Engineers and **MATLAB and Its Applications in Engineering** at ETSETB. I also teach **Data Protection** in the Master's degree in Cybersecurity (MCYBERS), at the same school .

I'm also collaborating with UBa Cyber Crypto Center and giving some on-line lectures there at the MSc level.

I have taught other subjects at ETSETB: Linear Algebra, Calculus, Advanced Calculus, Differential Equations, Mathematics for Telecommunication, Probability and Stochastic Processes and Simulation and Numerical Methods Applied to Telecommunication. In addition, I have also taught Cryptology as an optional subject at FME. I was also interested in the design of self-learning tools.

Classnotes and slides

Find below some class notes, slides and other documents related to the different subjects:

- Data Protection (Master MCYBERS)
- Cryptography (Optional subject FME)
- Introduction to Cryptology (Master Program at UBa Cyber Crypto Center)
- Security Models in Cryptography (Master Program at UBa Cyber Crypto Center)
- Codes and Cryptography (Master MAMME)
- Blockchain (PAE ETSETB)

Teaching publications

Home | Publications | CV | Teaching

Outline

The Setting (I)

Perfect environment: No storage or communication errors or excessive message delivery delays.

The Setting (I)

Perfect environment: No storage or communication errors or excessive message delivery delays.

Users: divided into

- good guys (honest)
- bad guys (corrupted by an adversary)

The Setting (I)

Perfect environment: No storage or communication errors or excessive message delivery delays.

Users: divided into

- good guys (honest)
- bad guys (corrupted by an adversary)

Alternative model: Rational Cryptography (from Game Theory). Only selfish guys (not necessarily honest, can collude).

The Setting (I)

Perfect environment: No storage or communication errors or excessive message delivery delays.

Users: divided into

- good guys (honest)
- bad guys (corrupted by an adversary)

Alternative model: Rational Cryptography (from Game Theory). Only selfish guys (not necessarily honest, can collude).

Simplest case: One honest user, one bad user.
E.g.: Secure binary data storage.

The Setting (II)

Adversarial behavior:

The Setting (II)

Adversarial behavior:

- **static:** corrupted users are fixed before starting the actual attack
- **dynamic:** corrupted users are decided on-the-fly during the attack

The Setting (II)

Adversarial behavior:

- **static:** corrupted users are fixed before starting the actual attack
- **dynamic:** corrupted users are decided on-the-fly during the attack

- **passive:** corrupted users follow the protocol and try to learn more than they are allowed to
- **active:** corrupted users deviate from the protocol in any arbitrary way

The Setting (II)

Adversarial behavior:

- **static:** corrupted users are fixed before starting the actual attack
- **dynamic:** corrupted users are decided on-the-fly during the attack

- **passive:** corrupted users follow the protocol and try to learn more than they are allowed to
- **active:** corrupted users deviate from the protocol in any arbitrary way

- **bounded:** the adversary has limited resources (computational power, memory)
- **unbounded:** the adversary has unlimited resources

Perfect Symmetric Encryption

Definition (Perfect Privacy)

For any probability distribution (source) of $M \in \mathcal{M}$ and for a uniformly distributed $K \in \mathcal{K}$, **the random variables M and $\mathrm{Enc}(K, M)$ are independent.**

Theorem

For any correct and perfectly private symmetric encryption scheme $\ell_{\mathcal{C}} \geq \ell_{\mathcal{M}}$ and $\ell_{\mathcal{K}} \geq \ell_{\mathcal{M}}$.

The key cannot be reused for further encryptions!

There is no perfect solution for binary private storage!
In practice, we need $\ell_{\mathcal{K}} \ll \ell_{\mathcal{M}}$.

Computational Symmetric Encryption

Definition (Informal Computational Privacy)

For any probability distribution (source) of $M \in \mathcal{M}$ and for a uniformly distributed $K \in \mathcal{K}$, **the random variables M and** $\text{Enc}(K, M)$ **behave for a bounded adversary as if they were independent.**

Based on efficient statistical tests a computationally bounded adversary can run.

Needs some extra assumptions from Complexity Theory.

CRYE 6138 Security Models

Jorge L. Villar

UBa Cyber Crypto Center, Fall 2025

Introduction

—END—