Codes and Cryptography

Jorge L. Villar

MAMME, Fall 2015

PART I

Outline

Structure of the Course

PART I: **Codes** (24 hours)
Simeon Ball
First Exam: 29th October

PART II: **Cryptography** (28 hours)
Jorge Villar / Javier Herranz
Second Exam: 23rd December

Final Exam (only if necessary): 8th January

More information at:

- `http://mamme.masters.upc.edu/`
- `https://fme-intranet.upc.edu/tmp/consgd/cursactual/34954-e-3.pdf`
- `http://www-ma4.upc.edu/~jvillar/crypto_course/`
- `http://atenea.upc.edu/`

Outline

Introduction

The main topic:
Efficient and Reliable Data Transmission and Storage

Introduction

The main topic:
Efficient and Reliable Data Transmission and Storage

Data: digital data source = (infinite) sequence of binary or q-ary symbols

Introduction

The main topic:
Efficient and Reliable Data Transmission and Storage

Data: digital data source = (infinite) sequence of binary or q-ary symbols

Reliable:

Introduction

The main topic:
Efficient and Reliable Data Transmission and Storage

Data: digital data source = (infinite) sequence of binary or q-ary symbols

Reliable: against

- imperfect environment $\longrightarrow$ **Information Coding**
- malicious users $\longrightarrow$ **Cryptography**

Introduction

The main topic:
Efficient and Reliable Data Transmission and Storage

Data: digital data source = (infinite) sequence of binary or q-ary symbols

Reliable: against

- imperfect environment $\longrightarrow$ **Information Coding**
- malicious users $\longrightarrow$ **Cryptography**

Introduction

The main topic:
Efficient and Reliable Data Transmission and Storage

Data: digital data source = (infinite) sequence of binary or q-ary symbols

Reliable: against

- imperfect environment $\longrightarrow$ **Information Coding**
- malicious users $\longrightarrow$ **Cryptography**

Efficient:

Introduction

The main topic:
Efficient and Reliable Data Transmission and Storage
e.g.: Binary Data Storage

Data: digital data source = (infinite) sequence of binary or q-ary symbols

Reliable: against

- imperfect environment $\longrightarrow$ **Information Coding**
- malicious users $\longrightarrow$ **Cryptography**

Efficient: w.r.t.

- read / write / modify / delete time (per bit / per access)
- physical space
- life of stored data

Example: Binary Data Storage (I)

Reliable against imperfect environment $\longrightarrow$ **Information Coding**

Add redundancy (w/o degrading efficiency too much)

Example: Binary Data Storage (I)

Reliable against imperfect environment $\longrightarrow$ **Information Coding**

Add redundancy (w/o degrading efficiency too much)

- Use error correction codes: detect/correct a few errors
- Use data shuffling: protects against correlated error locations (Compact Disk)

Example: Binary Data Storage (I)

Reliable against imperfect environment $\longrightarrow$ **Information Coding**

Add redundancy (w/o degrading efficiency too much)

- Use error correction codes: detect/correct a few errors
- Use data shuffling: protects against correlated error locations (Compact Disk)

Efficient w.r.t. physical space $\longrightarrow$ **Information Coding**

Example: Binary Data Storage (I)

Reliable against imperfect environment $\longrightarrow$ **Information Coding**

Add redundancy (w/o degrading efficiency too much)

- Use error correction codes: detect/correct a few errors
- Use data shuffling: protects against correlated error locations (Compact Disk)

Efficient w.r.t. physical space $\longrightarrow$ **Information Coding**

Use a compression code (w/o degrading speed too much)

Example: Binary Data Storage (II)

Reliable against malicious users $\longrightarrow$ **Cryptography**

Example: Binary Data Storage (II)

Reliable against malicious users $\longrightarrow$ **Cryptography**

Confidentiality: Use symmetric encryption

Example: Binary Data Storage (II)

Reliable against malicious users $\longrightarrow$ **Cryptography**

Confidentiality: Use symmetric encryption

Integrity: Use message authentication codes

Example: Binary Data Storage (II)

Reliable against malicious users $\longrightarrow$ **Cryptography**

Confidentiality: Use symmetric encryption

Integrity: Use message authentication codes

Availability: Use secret sharing schemes

Codes and Cryptography

Jorge L. Villar

MAMME, Fall 2015

END OF PART I