

# Introduction to Cryptology

## 1.1 - Cryptography Today

Federico Pintore

Mathematical Institute, University of Oxford (UK)



UNIVERSITY OF  
OXFORD

# What is Cryptology?

Cryptology is the study of

- ❖ cryptography, and
- ❖ cryptanalysis.

# What is Cryptology?

Cryptology is the study of

- ❖ cryptography, and
- ❖ cryptanalysis.

Cryptography: the study of mathematical and algorithmic techniques which ensure information security.

# What is Cryptology?

Cryptology is the study of

- ❖ cryptography, and
- ❖ cryptanalysis.

Cryptography: the study of mathematical and algorithmic techniques which ensure information security.

Cryptanalysis: the study of mathematical and algorithmic techniques to vanquish cryptographic techniques.

# Information security objectives

They vary depending on context and requirements.

Examples:

- ❖ secrecy
- ❖ data integrity
- ❖ authentication
- ❖ non-repudiation
- ❖ ...

# Secrecy

Context: Two parties, **Alice** and **Bob**, communicate over an *unsecured channel* (i.e. a means of convey information, in which unauthorised parties can read, modify, delete, ...).

Objective: Protect the information from unauthorised parties.

Technique: Encryption scheme.

# Encryption scheme

- ❖ a *message space*  $\mathcal{M}$  (elements are called *plaintexts*)
- ❖ a *ciphertext space*  $\mathcal{C}$  (elements are called *ciphertexts*)
- ❖ a *key space*  $\mathcal{K}$
- ❖ a set  $\{E_e : \mathcal{M} \rightarrow \mathcal{C} | e \in \mathcal{K}\}$  of *encryption* bijections
- ❖ a set  $\{D_d : \mathcal{C} \rightarrow \mathcal{M} | d \in \mathcal{K}\}$  of *decryption* bijections s.t.

$$\forall e \in \mathcal{K} \ \exists! \ d \in \mathcal{K} : D_d = E_e^{-1}$$

Given  $e \in \mathcal{K}$ ,  $(e, d)$  is called *key pair*.

# Encryption scheme

- ❖ a *message space*  $\mathcal{M}$  (elements are called *plaintexts*)
- ❖ a *ciphertext space*  $\mathcal{C}$  (elements are called *ciphertexts*)
- ❖ a *key space*  $\mathcal{K}$
- ❖ a set  $\{E_e : \mathcal{M} \rightarrow \mathcal{C} | e \in \mathcal{K}\}$  of *encryption* bijections
- ❖ a set  $\{D_d : \mathcal{C} \rightarrow \mathcal{M} | d \in \mathcal{K}\}$  of *decryption* bijections s.t.

$$\forall e \in \mathcal{K} \ \exists! \ d \in \mathcal{K} : D_d = E_e^{-1}$$

Given  $e \in \mathcal{K}$ ,  $(e, d)$  is called *key pair*.

In a **symmetric-key encryption**,  $e = d$  for each key pair.



# Data integrity, authentication, non-repudiation

Context: two parties, **Alice** and **Bob**, communicate over an unsecured channel.

Data integrity: guarantee that data has not undergone unauthorised alteration.

Authentication: identification of parties entering into a communication.

Non-repudiation: avoid denial of previous commitments/actions by communicating parties.

Technique: Digital Signature scheme.

# Digital signatures

- ❖ a *message space*  $\mathcal{M}$
- ❖ a *signature space*  $\mathcal{S}$
- ❖ to each party  $A$  it corresponds a secret map  $\mathcal{S}_A : \mathcal{M} \rightarrow \mathcal{S}$
- ❖ to each party  $A$  it corresponds a public map  $\mathcal{V}_A : \mathcal{M} \times \mathcal{S} \rightarrow \{\text{true}, \text{false}\}$

# Digital signatures

- ❖ a *message space*  $\mathcal{M}$
- ❖ a *signature space*  $\mathcal{S}$
- ❖ to each party  $A$  it corresponds a secret map  $\mathcal{S}_A : \mathcal{M} \rightarrow \mathcal{S}$
- ❖ to each party  $A$  it corresponds a public map  $\mathcal{V}_A : \mathcal{M} \times \mathcal{S} \rightarrow \{\text{true}, \text{false}\}$

Digital signatures are **public-key cryptosystems**.

Also symmetric-key techniques exist for these objectives.

# What Cryptography was

Cryptography has a long history:

- ❖ largely an art,
- ❖ exploited to enable secret communications,
- ❖ until the 1970s, mainly used for military purposes.

The proliferation of communication systems since the 1960s has propelled its **transition from an art to a science.**

# What Cryptography is

Today, Cryptography is a science:

- ❖ formal definitions of security
- ❖ rigorous proofs of security
- ❖ precise and *simple* assumptions
- ❖ ubiquitous in our everyday life

# Definitions of security

Example: when is an encryption secure?

# Definitions of security

Example: when is an encryption secure?

- ❑ impossible to recover the key

# Definitions of security

Example: when is an encryption secure?

- ❑ impossible to recover the key
- ❑ impossible to recover the entire plaintext



# Definitions of security

Example: when is an encryption secure?

- ❑ impossible to recover the key
- ❑ impossible to recover the entire plaintext
- ❑ impossible to recover any information on the plaintext

# Definitions of security

Example: when is an encryption secure?

- ❖ impossible to recover the key
- ❖ impossible to recover the entire plaintext
- ❖ impossible to recover any information on the plaintext

A security definition is composed by:

- ❖ **security guarantees** (what the technique should prevent the adversary from doing)
- ❖ **threat model** (what is the *power* of the adversary)

# Proofs of security

**Provable security**: rigorous proof that a scheme satisfies a given definition of security

Most cryptosystems cannot be proven secure unconditionally.  
Proofs of security usually rely on assumptions.

Example: proof by reduction

- ❖ it is a proof by contradiction.
- ❖ a **reduction** turns an attacker against the security guarantees into an algorithm to solve a hard problem.

# Assumptions

Mathematical assumption: a given problem is **hard to solve**

- ❖ integer factorisation problem (given a composite number  $n$ , compute its factorisation)
- ❖ discrete logarithm problem (given a cyclic group  $\mathbb{G} = \langle g \rangle$  and  $h \in G$ , compute  $k \in \mathbb{Z}_{|G|}$  s.t.  $g^k = h$ )
- ❖ shortest vector problem (given a subgroup  $\mathcal{L} \subset \mathbb{Z}^t$ , find  $w \in \mathcal{L}$  s.t.  $\|w\| = \min\{\|v = (v_1, \dots, v_t)\| = \sum_{i=1}^t v_i^2 \mid v \in \mathcal{K}\}$ )

# Use in real life

- ❏ ATM machines
- ❏ On-line banking
- ❏ e-commerce
- ❏ emails
- ❏ cloud computing
- ❏ Streaming media providers
- ❏ ...


# Web Browsers

Sport news, comment and res: X +

theguardian.com/uk/sport

## News


Hide



▶

### /// Kipchoge shatters two-hour barrier to crash into mainstream

Sean Ingle



Pierre van Hooijdonk / /// I told Ron Atkinson he was a pub manager at Forest

22

Elements Console Sources Security >>

### Security overview

Overview

Main origin (secure)

- https://www.theguardian.com

Secure origins

- https://assets.guim.co.uk
- https://i.guim.co.uk
- https://phar.gu-web.net
- https://ophan.theguardian.com
- https://uploads.guim.co.uk
- https://tags.crowdctrl.net
- https://www.google-analytics.com
- https://sb.scorecardresearch.com
- https://secure-au.lmrworldwide.com
- https://www.facebook.com
- https://interactive.guim.co.uk
- https://api.nextgen.guardianapi.com
- https://support.theguardian.com
- https://cdn-gl.lmrworldwide.com
- https://www.googletagservices.com

This page is secure (valid HTTPS).

- Certificate - valid and trusted  
The connection to this site is using a valid, trusted server certificate issued by GlobalSign CloudSSL CA - SHA256 - G3.  
[View certificate](#)
- Connection - secure connection settings  
The connection to this site is encrypted and authenticated using TLS 1.2, ECDHE\_RSA with X25519, and AES\_128\_GCM.
- Resources - all served securely  
All resources on this page are served securely.

# Messaging Systems



# Mobile Applications

## **An Empirical Study of Cryptographic Misuse in Android Applications**

Manuel Egele, David Brumley  
Carnegie Mellon University  
{megele,dbrumley}@cmu.edu

Yanick Fratantonio, Christopher Kruegel  
University of California, Santa Barbara  
{yanick,chris}@cs.ucsb.edu

## **ABSTRACT**

Developers use cryptographic APIs in Android with the intent of securing data such as passwords and personal information on mobile devices. In this paper, we ask whether developers use the cryptographic APIs in a fashion that provides typical cryptographic notions of security, e.g., IND-CPA security. We develop program analysis techniques to automatically check programs on the Google Play marketplace, and find that 10,327 out of 11,748 applications that use cryptographic APIs – 88% overall – make at least one mistake. These numbers show that applications do not use cryptographic APIs in a fashion that maximizes overall security. We then suggest specific remediations based on our analysis towards improving overall cryptographic security in Android applications.



# Crypto Makes the Headlines

**Support The Guardian**  
Available for everyone, funded by readers

Sign in

**The Guardian**

Contribute →

Subscribe →

News

Opinion

Sport

Culture

Lifestyle

UK World Business Football Environment UK politics Education Society Science More

Apple

● This article is more than 3 years old

## Inside the FBI's encryption battle with Apple

**For months, the FBI searched for a compelling case that would force Apple to weaken iPhone security - and then the San Bernardino shooting happened**



<https://www.theguardian.com/uk/commentisfree>

# Cryptanalysis

It aims at showing that a cryptosystem is not secure.

- ❖ exploit gaps/flaws in the security proof
- ❖ make a problem *less* hard to solve than assumed

# Further Reading I



Jean-Jacques Quisquater, Myriam Quisquater, Muriel Quisquater, Michaël Quisquater, Louis Guillou, Marie Guillou, Gaïd Guillou, Anna Guillou, Gwenolé Guillou, and Soazig Guillou.

How to explain zero-knowledge protocols to your children.  
In *Advances in Cryptology—CRYPTO'89 Proceedings*,  
pages 628–631. Springer, 1990.



Alice Silverberg.

Mathematics and cryptography: A marriage of  
convenience?

In *Annual International Conference on the Theory and  
Applications of Cryptographic Techniques - EUROCRYPT  
2020*, pages 3–9. Springer, Cham., 2020.

# Further Reading II



Marc Stevens, Elie Bursztein, Pierre Karpman, Ange Albertini, and Yarik Markov.

The first collision for full sha-1.

In Annual International Cryptology Conference - CRYPTO 2017, pages 570–596. Springer, Cham., 2017.