

# Security Notions and Definitions



Nicolas T. Courtois



- University College of London



## Part 0

# Some Vocabulary Revision



## What is Cryptography ?

- Classical notions (cf. dictionaries).
  - Cryptography: “the art keeping messages secure”. Classically - mostly about secure communication...
  - Cryptanalysis: the art/science of breaking codes and ciphers.
  - Cryptology: Cryptography+Cryptanalysis.
- These definitions are very much outdated....

## My Own Definitions (1):

### Cryptography:

The art and practical techniques for achieving “data security goals” ...

### Cryptology:

The science of achieving “data security goals” ...

# Cryptanalysis

from the Greek

- **kryptós**, "hidden"
- **analýein**, "to loosen" or "to untie"  
= Breaking (secret) codes

Term coined in 1920  
by William F. Friedman.



## More Modern Approach:

Cryptanalysis = evaluation of existing weak points: Challenging the security goals by attacks and showing that the security is lower than expected (does not have to recover the key in practice, just show that some claims are not justified).

## My Own Definitions (3):

### Cryptology:

The science that allows to justify / undermine cryptographic security.

Science: establishing facts and proving theorems.

Modern Cryptology: Prove the “data security goals” are achieved, while minimising the number of, and maximizing the plausibility of “basic assumptions”, to be tested by the attackers.



## Vocabulary

### Messages:

- confidentiality==secrecy  $\neq$
- steganography (*stéganos – gràfein*): conceal the very existence of the “message”
  - Message (‘stego-work’) is embedded in ‘cover-work’
    - can be encrypted independently
- covert channels: use one cryptosystem and add an extra hidden channel
  - (e.g. embed some hidden message inside a standard digital signature)
- information hiding / embedding / watermarking: may be known/readable, make hard to remove...

### People:

- privacy:
  - ability to control how data about you propagate, conceals properties of people, not the messages
- anonymity, pseudonymity



## Codes and Ciphers

confidentiality==secrecy

- code, coding theory, encode / decode
  - no secret here
- cipher = a secret code
- cryptosystem = cryptographic system
- clear\_text=plain\_text
- ciphertext=cryptogram

- encryption = encipherment
- decryption  $\geq$  decipherment
- cryptanalysis  $\leq$  breaking the cryptosystem (in practice)

## Redundancy

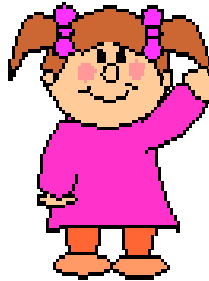
How error correcting codes work ?

They add redundancy to the message.

## Alice and Bob

confidentiality

- Alice



- Bob

- Eve



multiparty setting

- Alice



- Bob

- Charlie



## Part 1

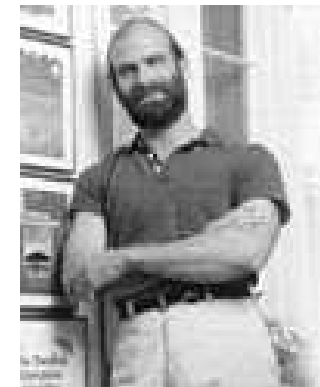
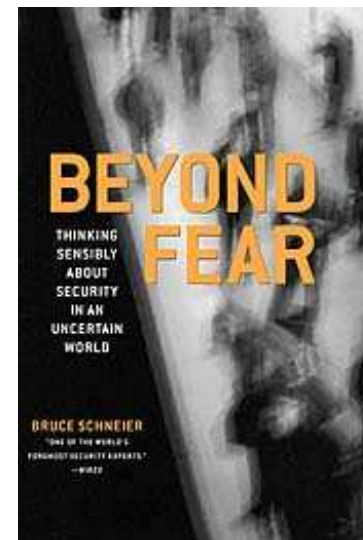
# [In]Security Boom ?!



## Reinvent Our Ideas About Security ?

“We don’t need to learn something completely new; we need to learn to be **smarter, more skeptical, and more skilled** about what we already know. “

Bruce Schneier “Beyond Fear” book [2003].  
“professional thinker about security”



## What is Security About? [Courtois]

Security:  
protect the value(s).

What value(s) ?  
ALL !

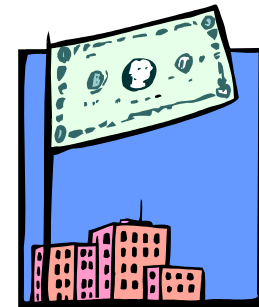
# Security: protect the value(s).

What value(s) ?

- **Money** [economical security]

But NOT ONLY MONEY.

- **Life** and the quality of life. (CB, car panic button)
- Privacy is a value in itself.
- Good technology vs. bad one
- **Freedom, Justice**, etc...

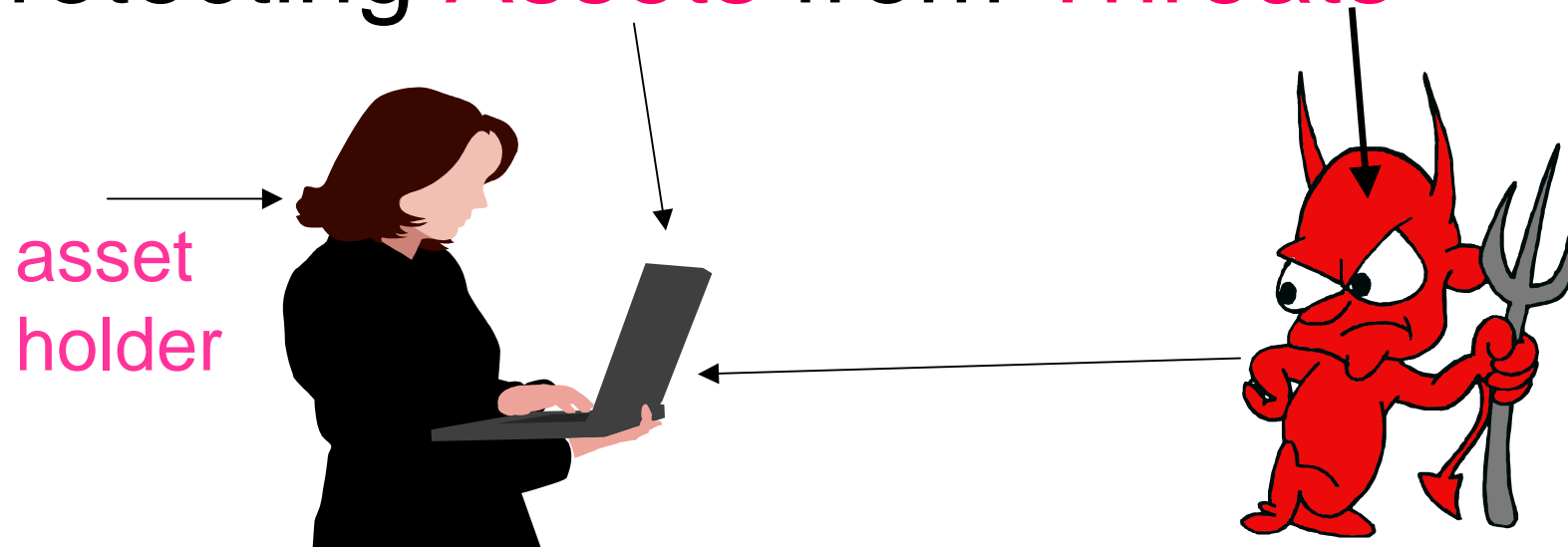


# Security:



Common Criteria [ISO15408]  
terminology:

## Protecting **Assets** from **Threats**





## Main Goals:

- Confidentiality
  - Integrity
  - Authenticity
- Accountability
- Availability

# In There a Need for More Security?

or will I get a job tomorrow?

Hegel [1770-1831], German philosophy

The history is the process of  
broadening freedom.

## Why Security, why Cryptography ?

Freedom, new technologies

⇒ More possibilities

⇒ More security problems...

Freedom  $\rightarrow \infty$

⇒

Security  $\rightarrow 0$

## Freedom $\Rightarrow$ Security issues...

### Examples:

- invention of train/metro  
 $\Rightarrow$  threat of pickpockets
- car traffic:  
 $\Rightarrow$  major cause of death (2M/year),  
trains are much safer
- invention of the internet  
 $\Rightarrow$  pirates and hackers.

## Why Security, why Cryptography ?

- Freedom of travel by car, plane, train etc..  
⇒ major security issues of our society  
(e.g. 1.2 M people die every year in car accidents !)  
⇒ emphasis on security.
- Unlimited exchange of data over the internet  
⇒ unlimited security problems.

Prof. Shigeo Tsujii,

[invited talk at ICISC 2005 conference]

“The society should be designed to help people to enjoy the freedoms”... “broadened by IT”.

Enjoy benefits, limit/remove disadvantages.

## The Need for Security

The goal of cryptology [Courtois]:

Add security to the information technologies (IT)

that are

enablers/disablers

**by nature** insecure.



More prof. Shigeo Tsujii:

IT Security  
Technologies and Solutions



Information Ethics

# Information Security:

## Multi-disciplinary Science.

- Ethical foundations - what we do (not) want...
- Cryptology: Maths, computer science, technology...
- Economics of fraud vs. the case for 'security business', new technology marketing, adoption barriers, usability barriers
- Legislation and Regulation:
  - EU Directives and state laws and regulations, obligations of disclosure for public companies,
  - law deterrents define attack patterns and shape the threat landscape
  - laws and public demand shape products and markets ...
  - industry best practices and rules, security policies.
- Psychology and sociology, human behaviour, human influence, human interaction...
- Understanding risk in complex eco-systems, strategic and organizational responses to insecurity/threats, risk perception and risk management,
- Security awareness and education/training, towards a security culture...

## Is it Easy to Achieve ?

- The security is **difficult**. BUT:
- **The security is for everyone**. Every day we have to take informed security decisions:
  - when we drive,
  - we do shopping on the internet
  - walk in a street at dark/take a taxi
  - etc..

## Is it Easy to Achieve ?

- The security is difficult.
- Cannot be seen, only sometimes the opposite can be seen: when a product/technology is not secure.

Even then the security breach is usually hard to find and point out.

(it may take 20 years to find an obvious attack !)

### \*\*\*Karl Popper's Philosophy of Science

- Most security claims should be hold as 'provisionally' true.
  - Cannot be proven.
  - They can only be disproved.

## The Security is Difficult.

- The really good security is **difficult** to find/achieve/buy.

 “Just buy our latest “OZ XP” solution and you’ve all-you-need-in-one security for the next 500 years”....

 Buy an anti-virus, anti-spam, anti-spyware and 57 rootkit and Trojan removal tools...

## Who needs Security / Cryptography

Security is usually a matter of public interest...

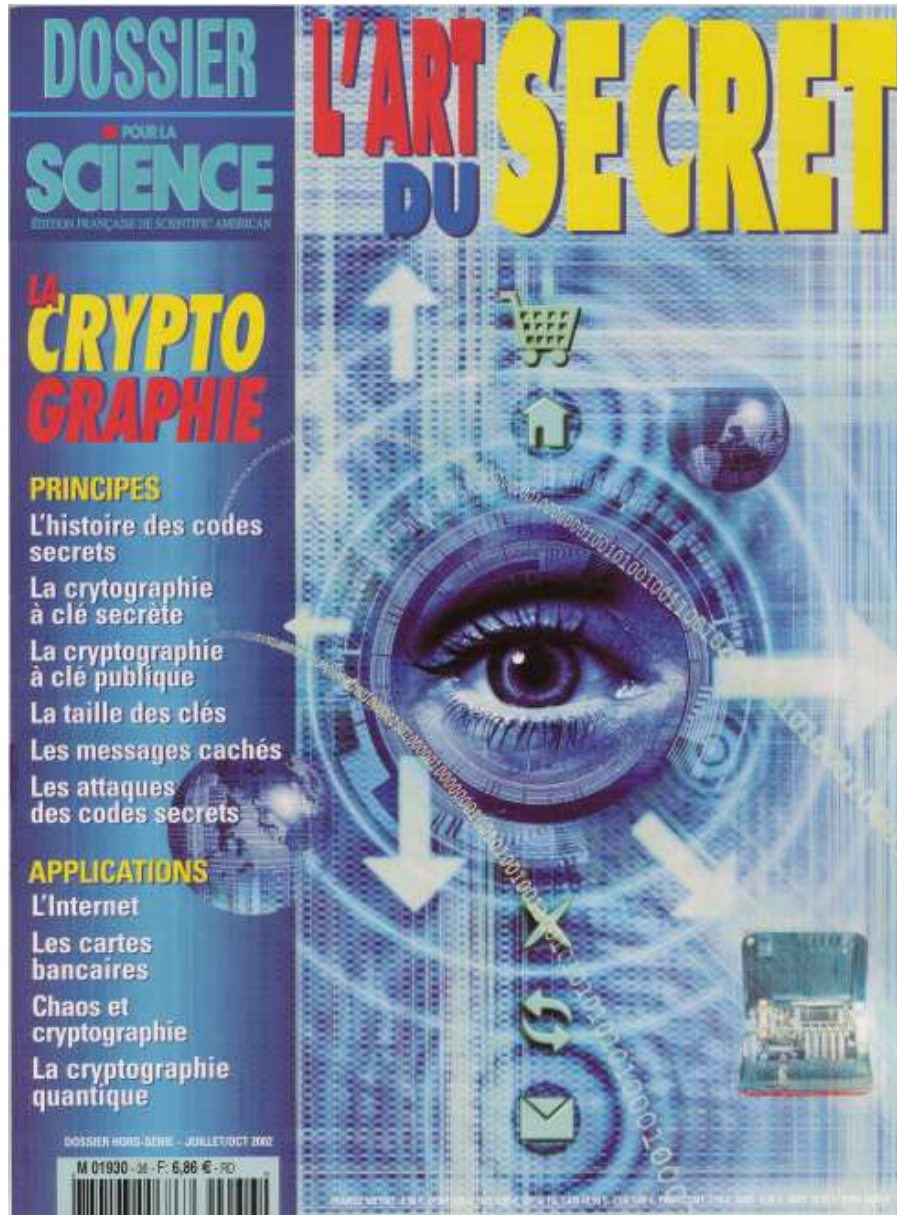
Frequently neglected,  
who cares about the public interest...

Currently private companies spent more  
on coffee than on real security...



For example smart cards for banking  
have not been adopted in many countries  
⇒ Ten times as much fraud.

See:



« Pour La Science »

(last summer, second edition soon ?)


- Jacques Patarin  
La Sécurité des  
Cartes Bancaires  
page 66
- Nicolas Courtois,  
Authentification  
Page 54



## Social aspects of security.

- Difficult, cannot be seen.
  - False security solutions and false security experts are abundant.
  - Some secret government agencies are not an exception either.
- Obscurity is an enemy of security.

## Who to Trust ? Crypto Expertise.

- Very few countries have a real expertise in cryptology. USA, UK, France, Israel, ...  
France: long tradition (more than 4 centuries) of well founded and important government / diplomatic / military cryptographic secret service: “le service du chiffre”.
- Most companies in the security market don't have sufficient expertise !  
⇒ pejorative: “Commercial security”;  
 works if you believe it.

Luckily...

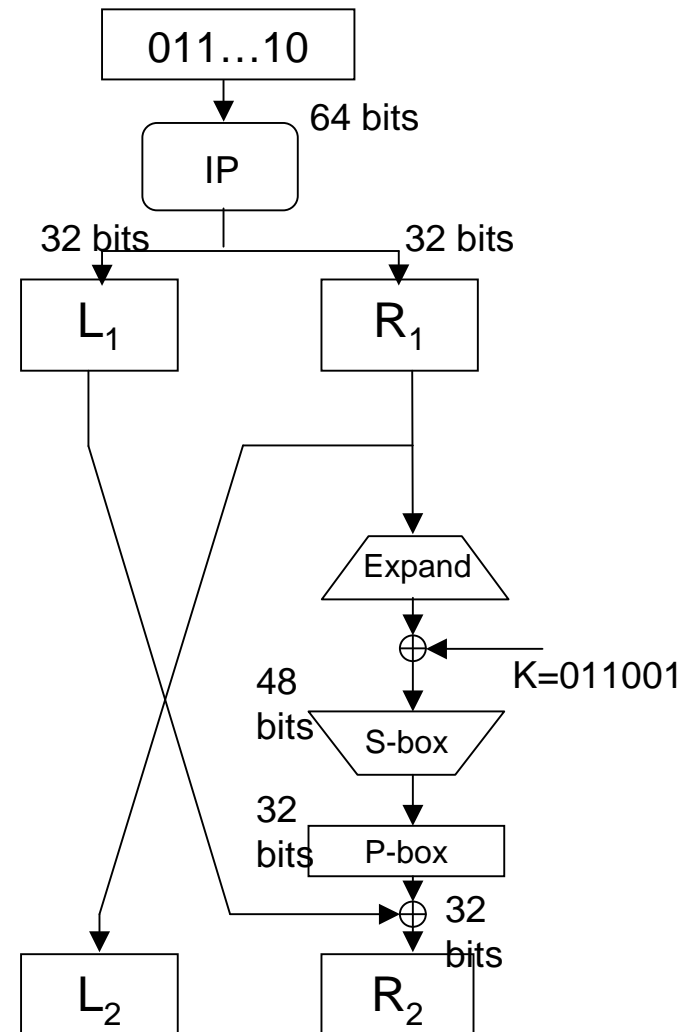
Good security solutions are  
known to the public.

In particular, cryptography  
makes **unbreakable** “locks”  
(unbreakable even with  
combined resources of the whole planet)

## Example: Encryption

- Public domain algorithm.

DES  
Data  
Encryption  
Standard



## \*\*\*The End of DES...

Designed by IBM and the NSA in the 70s.

Widely used, not secure anymore.

$2^{56}$  possible keys – few minutes to break.

Triple-DES still widely used...  
Progressive replacement by AES.

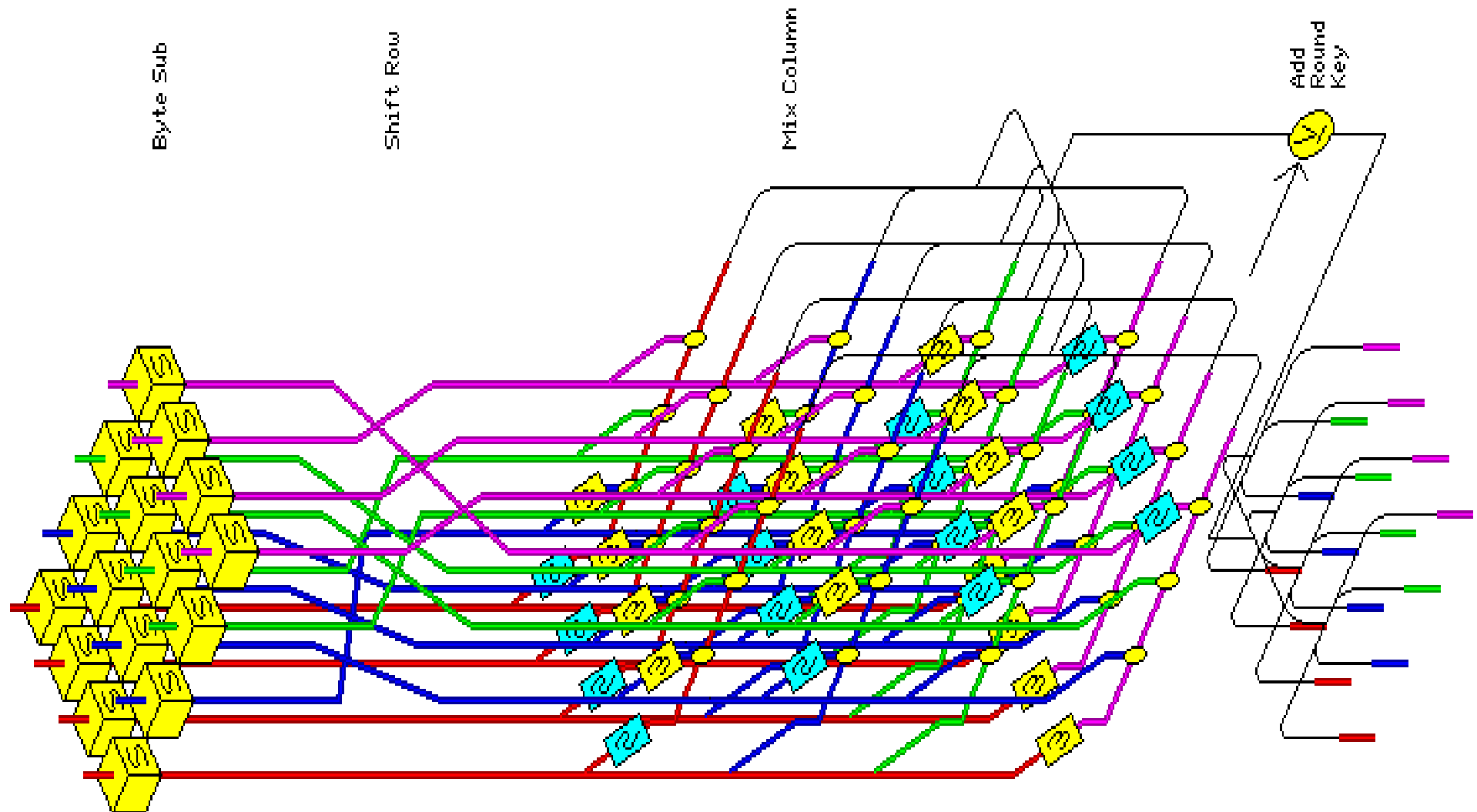
Encrypt with AES...

Advanced  
Encryption  
Standard

FIPS standard  
for US government and the industry.

Block Cipher,  
1 block = 128 bits = 16 octets.

# One Round of AES (out of 10):



## AES Keys

- **128** bits keys. Also 192 and 256 bits but these are not permitted in France... No need at all.
- **$2^{128}$**  possible keys. HUGE NUMBER.
- Should remain secure until **2070**.
- Academic attacks at sight, later,  
[Courtois et al]: **faster than  $2^{128}$**  ?
- Even if broken faster than  **$2^{128}$** , still much more secure than triple DES,  
not sure that we know  
a better algorithm.



OK.

- Let's buy it (AES) !

(in few years from now, there will probably be no **Axalto** product without AES...)

## Encrypt Email - Commercial IT Security.



- Buy it from [www.searchsecurity.com](http://www.searchsecurity.com) ?
- One of the most prominent promoters of commercial information security.
- "The web's best security-specific information resource for enterprise IT-professionals."

Buy from [www.searchsecurity.com](http://www.searchsecurity.com) ?

They do NOT employ a single person with an expertise in cryptography.

- According to them, AES has 9 rounds.
- AES is the main tool in applied cryptographic security,
- It's like if your math teacher told you that  $2+2=5$  and nobody noticed it was false for 2 years !

## Buy Security ?

- Cannot be bought. 
- We need:
  - Protections [Cryptology]. 
  - Backup when they fail.

## Buy Security ?

- Cannot be bought.
- But ALWAYS HAS A COST (and rather several costs)



## The Problem we Have...

- Cannot buy security so easily.
- Does require appropriate security technology/maths/security proofs/software/hardware, but also:  
paying attention to security.

Security requires to be smart.

## Trap Number 1736:

- Maybe the program uses only 8 bits ?  
How do we know ? (I once checked the Norton Utilities 40-bit Encryption...)

## Trap Number 1737:

- Maybe there is a trapdoor: the key is leaked to the US headquarters of the company / directly to the NSA (Swiss Crypto AG affair).

## The Very Nature of Security:

Bruce Schneier “Beyond Fear” book [2003], p.1:



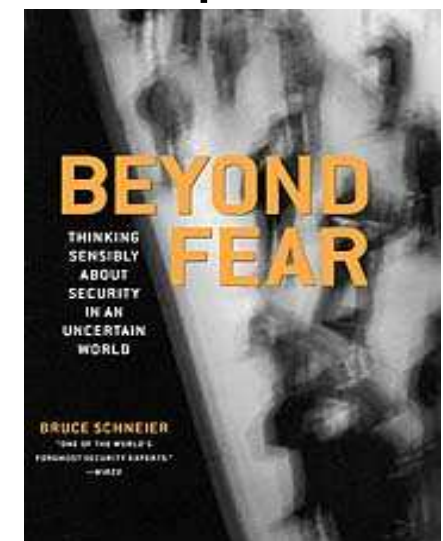
Critical to any security decision is the notion of

**[security] trade-offs,**

meaning the **costs** – terms of money, convenience, comfort, freedoms, and so on - that inevitably attach themselves to any security system. People make security trade-offs naturally.

Many **costs** are intangible and hard to trade for money:

Paying attention, loss of freedom&privacy,  
being subject to unforeseen risks and consequences etc.





## Commercial Security: It gets worse...

- You cannot buy it only because it is barely present on the market.
- The customers do not see the difference, why bother having it in the first place ?
- And also a hidden agenda:

## When Bill Gates Says:

- “Security is all we care about”,

He thinks:

- We want us (our market) to be secure...

If somebody talks about security..

Ask the question:

security for whom ?

## Is Security Available Today ?

My claim:

1. On some fronts problems are solvable and **nearly solved**: E.g.
  - UMTS, Bank cards.
  - Asymmetric cryptography: provable security...
2. On other fronts the **scarcity** of security **remains**... and is getting worse and worse so far... E.g.
  - Security of a PC.
  - Symmetric cryptography: new attacks and new attacks every year...

The demand grows faster than the supply. The demand is not organised, has no financial power and is not developed commercially... The market is still driven by scarcity and false security solutions.

## Why don't we fix it ?

- Nobody wants it to be fixed...

(and many people do not realise that it **can** be fixed).

Bruce Schneier:

"**Cybercrime** "...is not a technological problem. It's an economic problem: the **incentives aren't there** for smart people to solve the problem ..."

(The Economist, 29 November 2003, top of the page 77).

## Part 2

# Modern Cryptography



## What is Cryptography ?

- Much more than encrypting things.
- Can achieve all kinds of security goals, not only privacy.

## Goals of Cryptography

1. **Confidentiality**: privacy, anonymity or pseudonymity.
2. **Authenticity**, Integrity, Non-repudiation...
3. Fair play and resistance to malicious behaviours in multiparty protocols...
4. Meta: Trust (or Accountability), Openness, Governance, Compliance, Auditing, Alerting, Risk Assessment...

## Means to Achieve These Goals

### **Cryptographic Schemes / Cryptographic**

**Protocols:** Necessary ingredients:

1. The best mathematics and
2. computer science on earth.
3. Review and constant scrutiny of hundreds of independent experts.

### **How to use these correctly:**

4. people/programmers understanding “how to use it”
5. + appropriate software/hardware environment (e.g. smart cards)
6. + “trusted infrastructure” (trusted companies).



## Means and Tools to Achieve Security

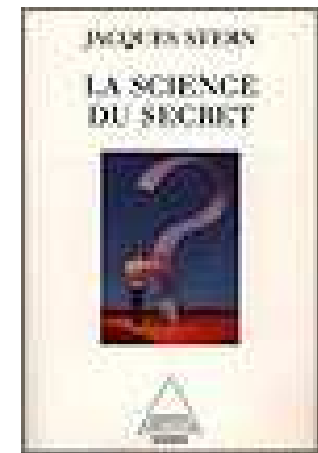
# MAIN TOOL in Cryptography / Security:

The Secret  
(or Secrecy)

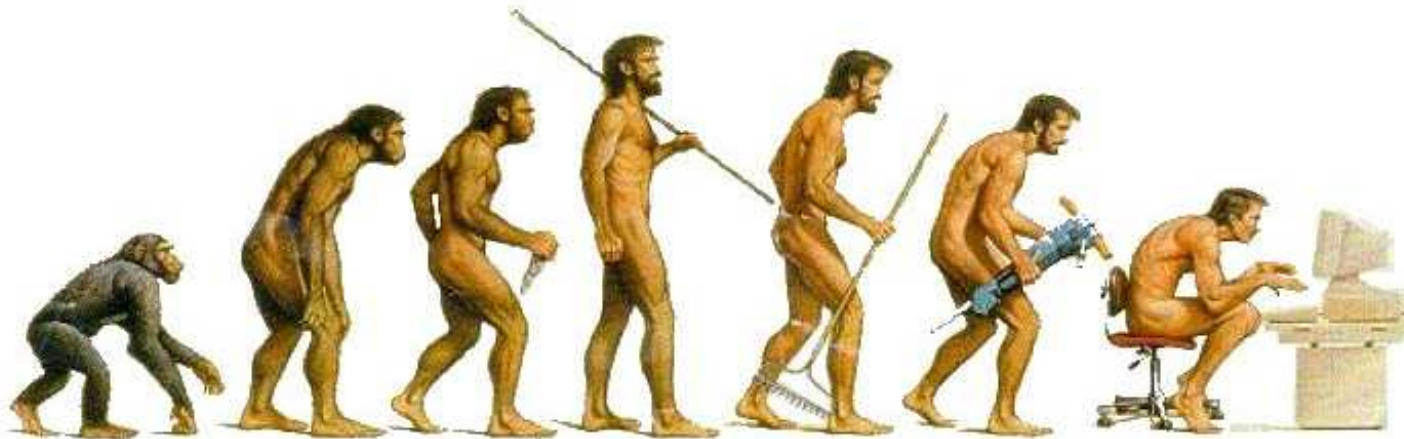
Jacques Stern book:

# La Science du Secret

(éditions Odile Jacob)



# 3 Stages



## Evolution of Information Security

3 stages [Courtois] :

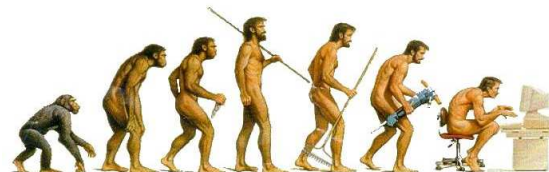
1. Protections that are secret



2. Based on a secret key

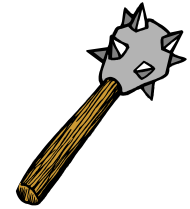


3. Public key solutions.



## First Stage – Security By Obscurity

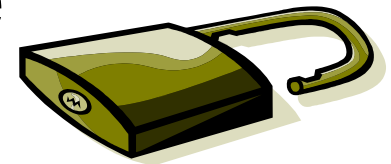
The stone age of cryptography...



Like hiding the key under the doormat.

Usually broken if you try long enough.

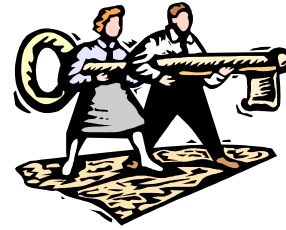
Hackers paradise: just give me enough coffee...



Unpredictable and catastrophic when some information leaks out...

## Second Stage – Secret Key Cryptography

Shared Key.



The key remains secret.

Algorithm can be published !



## Kerckhoffs principle: [1883]

“The system must remain secure should it fall in enemy hands ...”



# Short History of Cryptology

Until 1977 all cryptographic algorithms were secret...

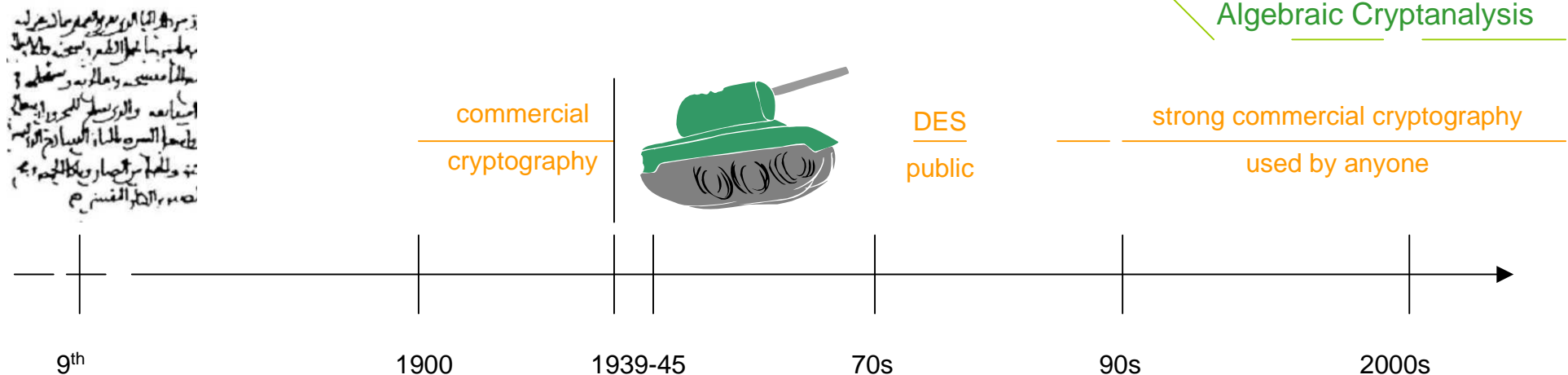
Until about 1945-60s all ciphers were broken...

Statistical cryptanalysis: frequency and language -> DC, LC, GLC, BLC,

Cryptanalysis with machines -> computers – increasing computing power

from art to higher mathematics... [Cocks, RSA, Public Key Cryptology]

## Algebraic Cryptanalysis

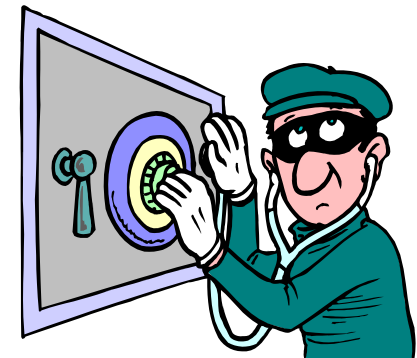




## Second Stage – Secret Key Cryptography

Appeared with perfecting  
Enigma... **More and more  
computation**, necessity to  
build machines called  
“bombs”.

Computational Security:  
time+money.

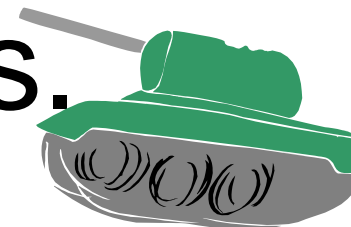


## Second Stage – Secret Key Cryptography

Good Crypto:  
**can publish** the algorithm.



- In 1977 the American government publishes DES.
- Before: good encryption algorithms were highly classified weapons.



## Proprietary Algorithms

- Maybe I can break it ?
- No time, no motivation: many “lousy” algorithms, few people able to break them...

## Partial Solution...

- If one can break RSA-2048 bits, RSA Security offers 200 000 US\$.
- For ECC: Certicom offers 725 000 US\$.
- For AES: 0 \$ is offered.

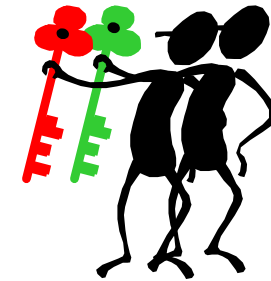
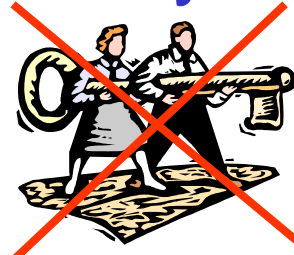
The US government wants cryptologists to work for free...

## Contemporary Cryptology

- Re-Birth of Cryptology: Invention of Public Key Cryptosystems [1970s].
- ONE CAN DO MUCH BETTER than encryption with a [shared] secret key !!!!  
(which is not obvious)

## Third Stage – Public Key Cryptography

No shared key,  
One **private** and  
one **public** key.



Private key:  
generated and stored  
securely...



## Third Stage – Public Key Cryptography

Public key:

can be distributed to many parties.

Does **not** have to be public  
(but the system remains  
secure when it is).



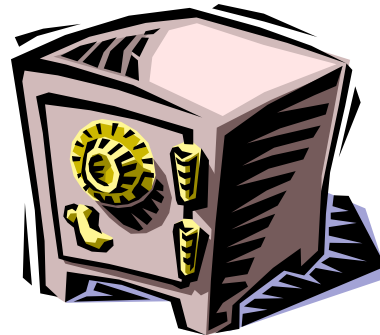
# Modern Cryptography



## How do you Achieve Security

First: Understand what we want.

Then: Try to achieve it.



## Modern Cryptography:

1. First: Understand what we want:

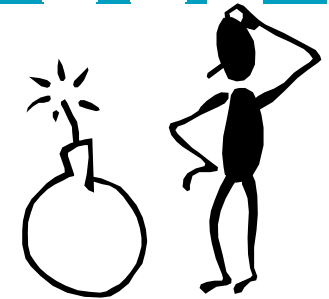
Formal security definitions.

2. Then: Try to achieve it:

Prove the Security w.r.t. a hard problem.

There is no other way known.

## Modern Cryptography:



### Serious Issues:

- Only in about 1998 people understood what is a secure public key encryption system.
- Only in about 2001 people did understand how to use RSA properly.
- Most standards (e.g. ISO) are not secure.

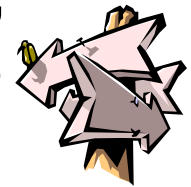
### Minefield:

Try-and-error  
just did not work...



## The Security ? Formal Approach

It took more than twenty years to understand this, many mistakes have been made, even by the most prominent cryptographers...



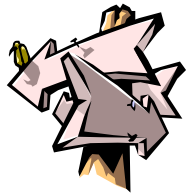
How to prevent mistakes ?

Provable security !

MUST READ:

- Bruce Schneier: “Secrets and Lies”; “Beyond Fear”.
- Invited talk by Jacques Stern, Eurocrypt 2003, Warsaw.

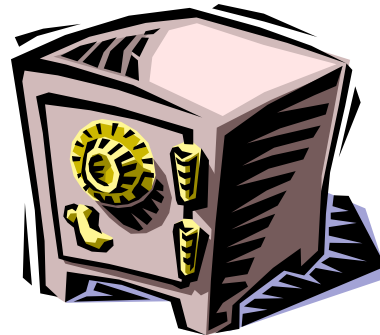




## How do you Achieve Security

First: Understand what we want.

Then: Try to achieve it.





Security  $\geq$  Safety

Difference:

protect against intentional damages...

Notion of an

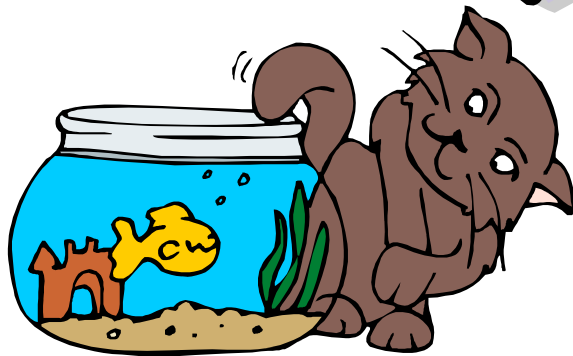
Attacker / Adversary.

Notion of an Attack.



Claim [Courtois, Schneier]:

Computer Security and real-world security are governed by the same laws !!!

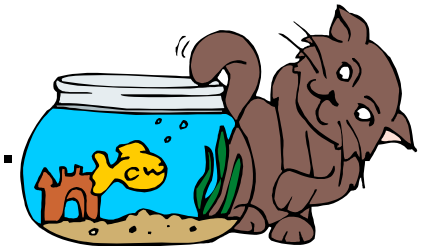




## The Security ? 3-point Formal Approach

What is Security ? Inability to achieve:

1. Security against what: Adversarial Goal.



2. Against whom: resources of the Adversary: money, human resources, computing power, memory, risk, expertise, etc..



3. Access to the system.







## The Security ? 3-point Formal Approach

**Security Notion / Definition** = a triple:

1. Adversarial Goal.

2. Resources of the Adversary.

3. Access / Attack.



One can ONLY talk about security w.r.t. a given triple. May not hold for other triple.



## Example 1: The security of your car.

The 1 may be:

1. Adversarial Goal.
  - puncture tyres
  - break the window
  - tag your car
  - steal it
  - prevent you from using it
  - hide it from you (joke)
  - put a bomb in it, etc...



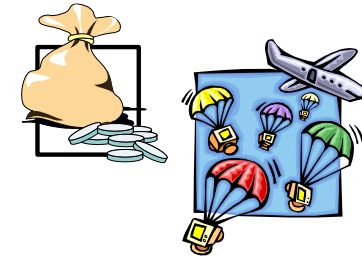


## Example 1: The security of your car.

The point 2 may be:

### 2. Resources of the Adversary:

- Bare hands
- A hammer
- Complete toolbox
- A Boeing 747
- Atomic bomb...





## Example 1: The security of your car.

The point 3 may be:

3. Access to the system:



- When you drive on the road
- When parked on the street.
- Enters your garage
- You borrowed him the key yesterday and he made a copy
- ...



## Example 2: The security of air travel.

September 11<sup>th</sup>:

The security was good, but with respect to a bad assumption.

1. **New goal:** before was rather to obtain/negotiate sth.
2. **Resources of the Adversary:**  
human life / risk capacity.
3. **Standard Access:** passengers.



## September 11<sup>th</sup> lessons:

Security of nuclear plants: completely re-defined.

- Accident risks, rather secure.
- Voluntary risks w.r.t. adversaries that have nothing to lose and unlimited resources (Saddam's fortune: 8 G\$ ?)



## Example 3: One-Time Pad

1. Decrypt / learn anything.
2. Infinite computing power !
3. Passive eavesdropping.

PERFECT !

1. Modify the end of the message.
2. Capacity to do a XOR.
3. Active: in-the-middle.

INSECURE !

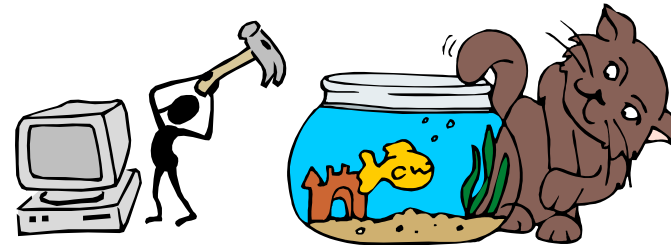


## Many security notions, but...

Take the **STRONGEST POSSIBLE** version:

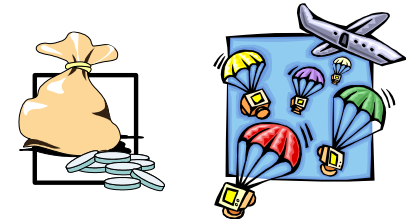
1. Adversarial Goal.

the weakest possible !



2. Resources of the Adversary:

The strongest possible: 10 G\$.



3. Access / Attack: The strongest possible,  
total adaptive “oracle” access.







Take the strongest possible version:

So far it works...

But: Partial order/hierarchy, there may be no “strongest possible” definition that encompasses all possible attacks. Then one have to achieve security w.r.t. several definitions.

- Problem: may interfere...
- Recommendation: if possible define a combined security notion, example: SINGCRYPTION.



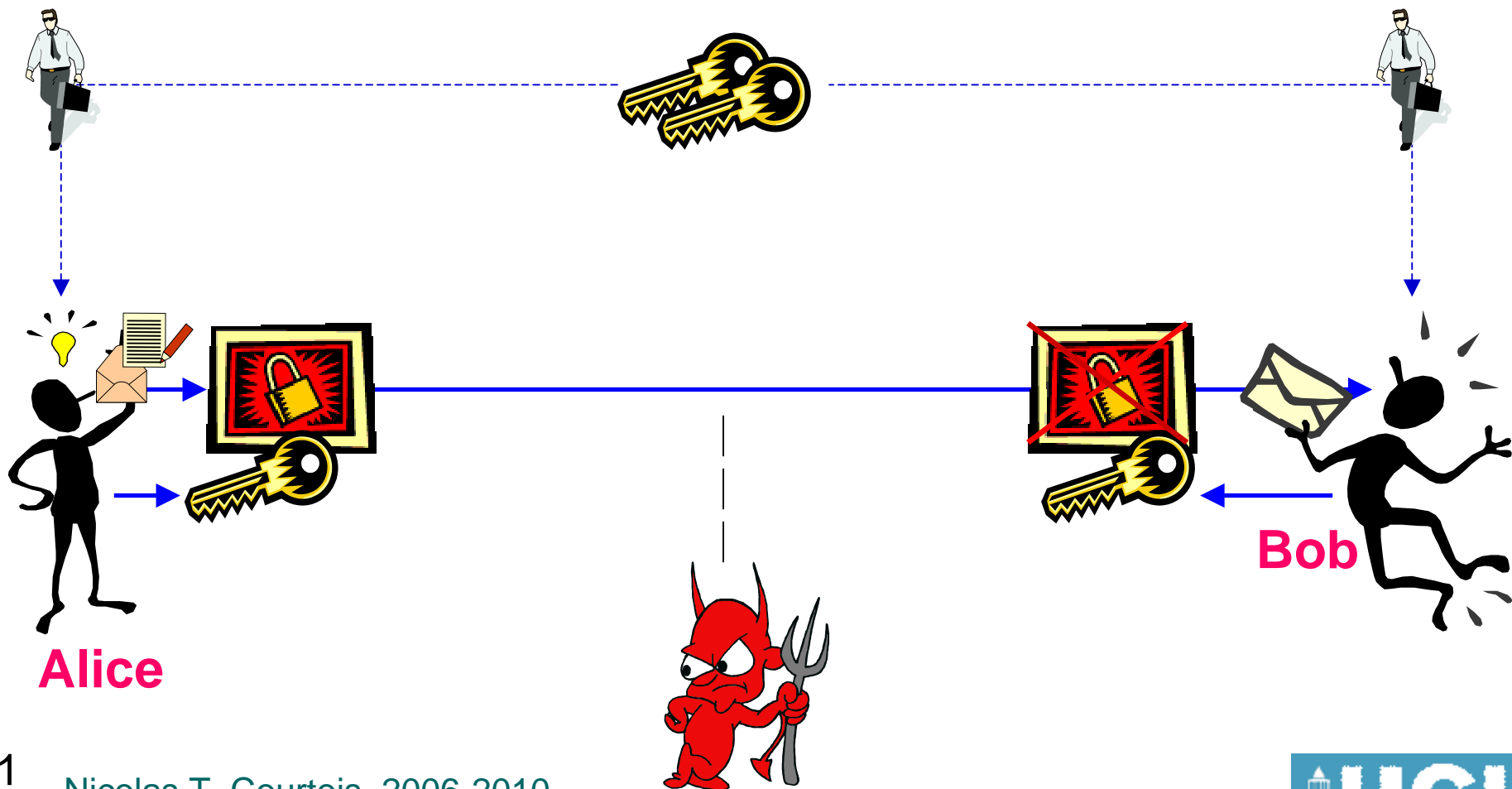
## Take the strongest possible ?

What for ?

1. **Cryptologists are paranoids.** It is a part of their job. (Door closed ? Use the window).
2. **The philosophy:** the door should be just closed, (not almost closed with 1mm space).
3. **It prevents against many future attacks,** not only the attacks that are known today.

(another way to prevent against future attacks: crypto “bio-diversity”: sign/encrypt several with different schemes: RSA + Sflash etc...).

# Secret-Key Encryption





## Example 4: Block Cipher

Block ciphers: What it isn't:

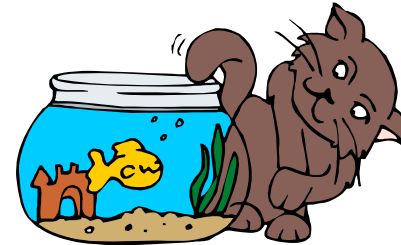
1. Adversarial Goal.

Get the right key and decrypt every message.

2. Resources of the Adversary:  
Blackboard + chalk.

3. Access / Attack:

The scheme is not even public.



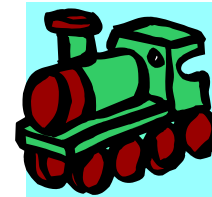
“Snake Oil Ciphers” (Remède de charlatan)...



## 1. The Adversarial Goal.



to know the good key with  
100.00000 % accuracy.



to compute K.



obtain any “significant”  
information on the key K.

Requires to be formalized further...



Protect against known and future attacks.

### 3. Access / Attack.



I see only the ciphertext: Q2HPDOE5@SZ2ML%



I see some plaintext/ciphertext pairs...



I can encrypt one one message of my choice...



I'm able to encrypt/decrypt anything, get the answer and adaptively choose new questions....

1 + 3:

- I'm able to encrypt/decrypt anything.  
and do not even know if this is a cipher  
or random box...

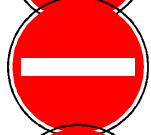
Strictly stronger: if someone can find the  
key or decrypt even one message: he  
can distinguish the cipher from a  
random cipher.

Protects against all the other threats.

## 2: Adversarial Resources



A blackboard+chalk ?



Pocket calculator ?



...



Any Probabilistic Turing Machine doing  $2^{80}$  computations (and any algorithm !!!).

Means: large agency with 1 Billion dollar budget (NSA, ex-KGB etc..).

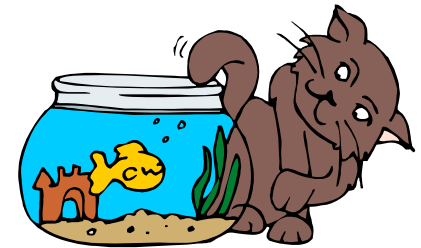




## Example 4: Block Cipher

Block ciphers: the strongest definition known:  
**Strong Pseudo-Random Permutation**

1. Adversarial Goal:  
Distinguish it from a truly random permutation with non-negligible probability.
2. Resources of the Adversary:  
Any Probabilistic Turing Machine doing  $2^{80}$  computations.
3. Access / Attack: May encrypt / decrypt any plaintext / ciphertext. (Adaptively Chosen Plaintext and Chosen Ciphertext Attack).





## Secret Key Encryption

### Strong Pseudo-random Permutation (SPRP)

Only then it becomes a serious cipher...

(there are stronger notions, e.g. super-strong and homogenous permutation generators [Patarin]... not of interest to us...)



## Secret Key Encryption

**SPRP. Informally:** mandatory requirement of the AES contest.

**Formally:** doesn't apply to a single object such as AES...

(cf. also Rijmen and Daemen define “very strong” notions of K-secure and Hermetic ciphers, can be applied to single cipher).  
(Bellare, Rogaway et al. define notions of “Finite PRF” and “Finite PRP”, see def 5 page 17 of full ver. Of Bellare, Desai, Jokipii, Rogaway, S.F.C.S. IEEE 1997.)

DES is too weak. Today it would never be accepted as a standard.



## Example 5: Public Key Encryption

PKE: the strongest definition known:

**IND-CCA2 == NM-CCA2**

1. Adversarial Goal.

Distinguish between encryptions of only two messages chosen by to the adversary.

2. Resources of the Adversary: Any Probabilistic Turing Machine doing  $2^{80}$  computations.



3. Access / Attack:

May decrypt any ciphertext except one (target).  
(Adaptively Chosen Ciphertext Attacks).





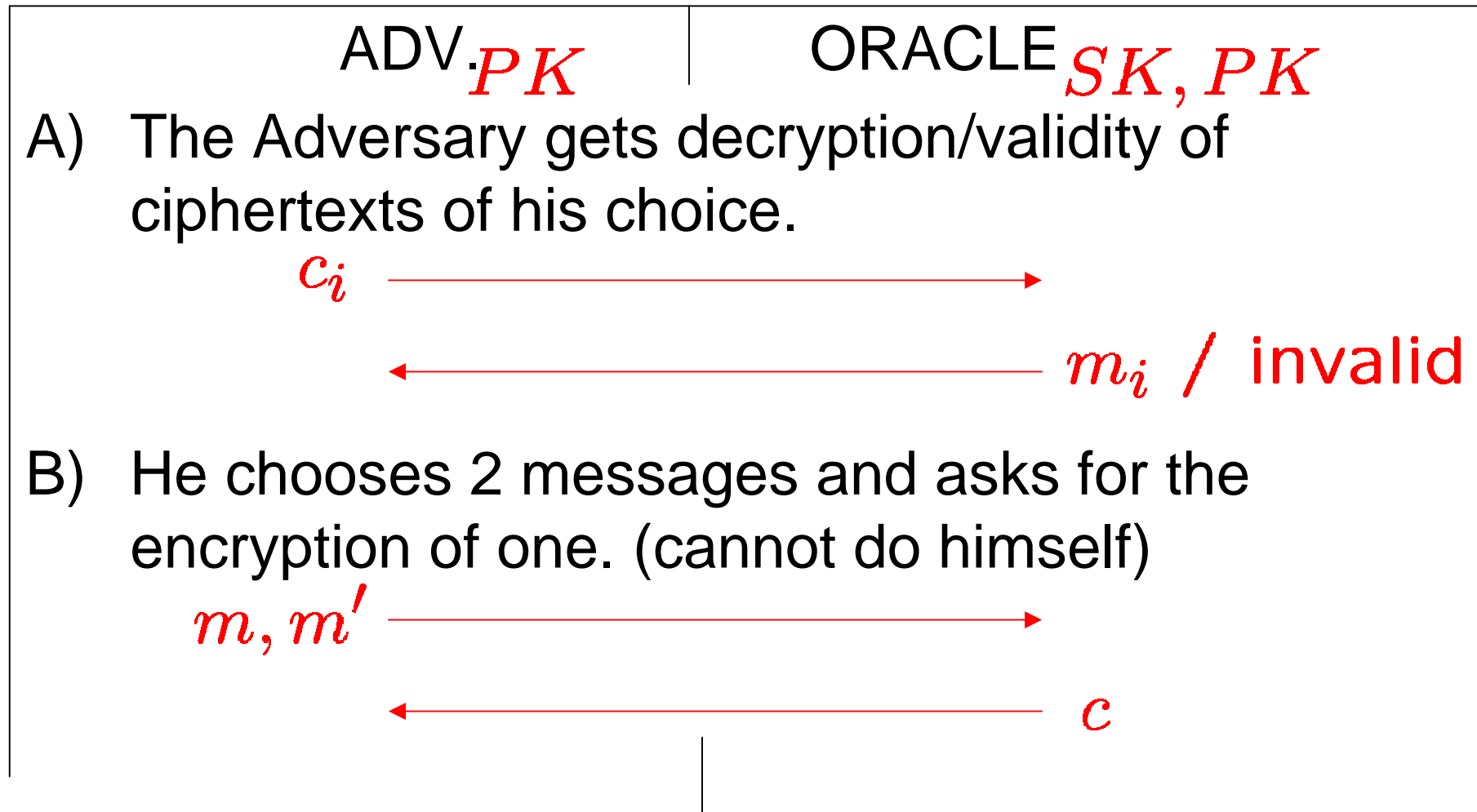
## \* Attacks on PK Encryption



1. Adversarial Goal.
  - Recover the private key,
    - e.g. factor  $N = pq$ .
  - Decrypt messages (much easier !)
    - e.g. e-th roots:  $x \mapsto x^{1/e} \bmod N$
  - Distinguish between  $E(\text{WAR})$ ,  $E(\text{PEACE})$
  - Distinguish between encrypting two chosen messages.

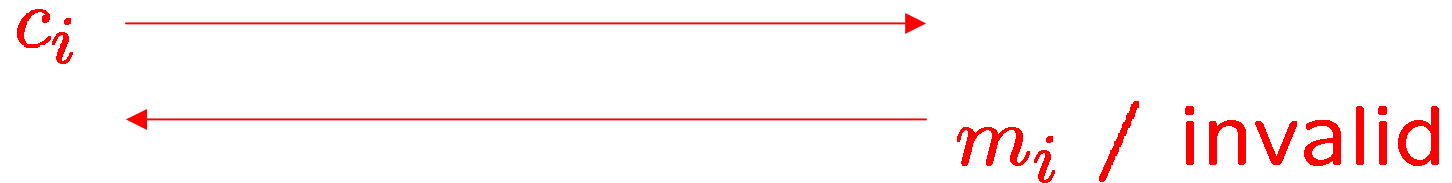
## \*“Good” Public Key Encryption – IND-CCA2

Semantic security w.r.t. CCA. Formulated as a game:



## \*IND-CCA2 Game contd.

C) The same as A, but all  $c_i \neq c$ .



D) Try to guess which one it was ?????



Adversary wins if he guesses correctly.

Let  $\varepsilon = |p - 1/2|$  be the **Advantage**, with  $p$  being the probability that the adversary wins.

## \*IND-CCA2 Game contd.

Let  $\epsilon = |p - 1/2|$  be the advantage of the adversary.

Definition:

A P.K. encryption scheme is  $(T, \epsilon)$ -IND-CCA2 if for **every** Adversary with running time  $\leq T$ , the probability of winning the IND-CCA2 game is  $\leq \epsilon$ .

Version 1: P vs. NP asymptotic security.

if  $T = n^{O(1)}$  then  $\epsilon = o(1/n^{O(1)})$

Version 2: Concrete security.

if  $T \leq 2^{80}$  then  $\epsilon \leq 2^{-40}$





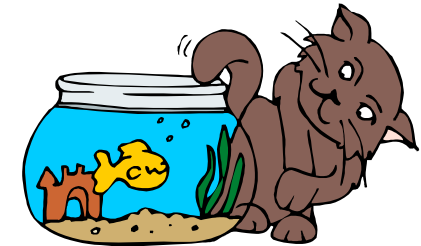
## Example 6: Public Key Signature

The “good” definition [Micali-Rivest 1988]:  
[Strong] **EF(or Unforgeability) / CMA**

### 1. Adversarial Goal.

Find any new pair  **$(M, \sigma)$**  !

Strong version: even if  $M$  is old (signed before).



### 2. Resources of the Adversary:

Any Probabilistic Turing Machine doing  **$2^{80}$**  computations.



### 3. Access / Attack:

May sign any message except one (target).  
(Adaptively Chosen Message Attacks).





## \*Attacks on Signature Schemes



1. Adversarial Goal.
  - BK - Recover the private key,
    - e.g. factor  $N = pq$ .
  - UF - Universal forgery – sign any message, may be easier ! e.g. compute:  $x \mapsto x^{1/e} \bmod N$
  - SF - Selective Forgery – sign some messages
  - EF - Existential Forgery – just sign any message, even if it means nothing useful.
  - Malleability: sign a message that has been already signed by the legitimate user.

# \*Signatures – Unforgeability-CMA2 Game

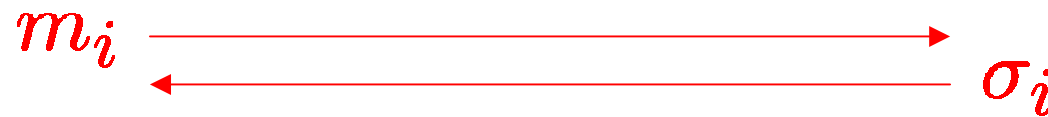
## One-more signature principle.

[Goldwasser, Micali, Rivest 1988].

ADV.  $PK$

ORACLE  $SK, PK$

A) The Adversary gets a signature of any message.



B) He wants to find a **new** valid pair message signature:  $(m, \sigma)$ ,  $m \neq m_i$

A scheme is  $(T, \varepsilon)$ -EF-CMA2 if...

Version 1: P vs. NP asymptotic security.

if  $T = n^{O(1)}$  then  $\varepsilon = o(1/n^{O(1)})$

Version 2: Concrete security.

if  $T \leq 2^{80}$  then  $\varepsilon \leq 2^{-40}$

## \*\*Related Reading:



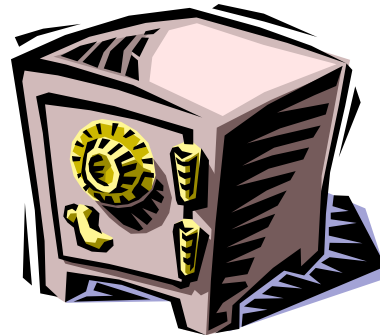
« Pour La Science »

- Jacques Patarin  
**La Sécurité des Cartes Bancaires**  
page 66
- Nicolas Courtois,  
**Authentication**  
Page 54
- Louis Goubin,  
**Attaques sur RSA**  
page 58

## How do you Achieve Security

First: Understand what we want.

Then: Try to achieve it.



How?

Cryptography: We just try.

Cryptology: Prove it mathematically.

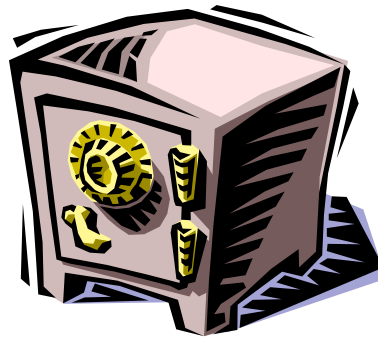
## Provable Security ?

Reduce the difficulty of breaking a system to a difficult mathematical (and computational) problem.

All the security reduces to this single problem.

## How do you Achieve Security

1. First: Understand what we want.
2. Then: Prove the security.  
Now all the security boils down  
to a single hard problem.



3. But is it really hard ? Evaluation.

Evaluation:

Try to break the  
problem(s).



How do you Evaluate Security

By breaking systems,  
or attacking the basic  
problems.

## How do you Evaluate Security

Bruce Schneier:

The security of a cipher  
is not in its design, it is  
in the analysis.

# Algebraic Attacks on Ciphers

Breaking a « good » cipher should require:

“as much work as solving a system of simultaneous equations in a large number of unknowns of a complex type”

**[Shannon, 1949]**



For 50 years people believed that large systems of equations are impossible to solve...

## Summary - Modern Cryptology:

The Security Requires:

1. Strong (Paranoid) and Formal Security Definitions.
2. Security Proofs w.r.t.
  - Sometimes: Additional Assumptions (ROM)
  - Always: **basic hard** problems.
3. Should be constantly **re-evaluated**
  - Need to question the definitions/assumptions
  - Must try **to attack** the basic problems !

## Part 3

# Cryptography Goes Applied

## Why Gemalto makes Smart Cards ?

Because our clients asked us to. **Bad answer.**

**A company that sees only the mature markets  $\Rightarrow$  soon on the decline ?**

## **Technology Company:**

Because people desperately need them and  
we only need to help them to  
realize it !

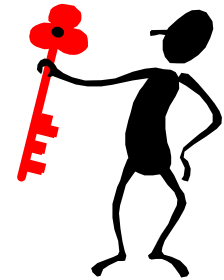
## Why Smart Cards ?

Poor security with secret key solutions.

Best protections:

Private-Public key solutions.

KEEP private keys private all the time !



HOW ?



## Why Smart Cards ?



Best protections:

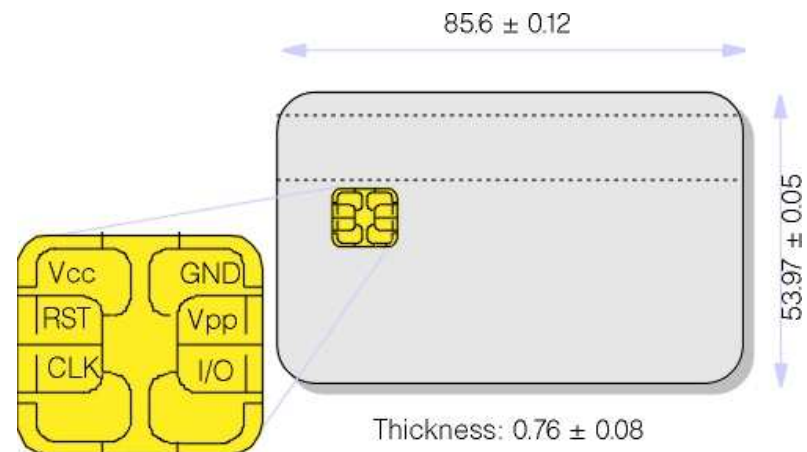
Private-Public key solutions.

KEEP private keys private all the time !

No real security with a PC.

Must be securely

- Generated
- Stored
- Used
- Backup
- Destroyed





# What Do We Need ?

= “a secure hardware device” !

USB interface



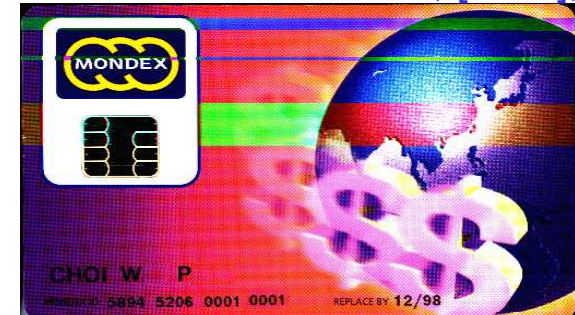
USB Token form factor

ISO, [USB]



SIM card form factor

ISO, [USB], [RF]



credit card form factor

1. **"intelligent"** (Smart): the card
  - handles computations (e.g. crypto)
  - manages data (OS, file system, access rights)
  - takes informed security decisions (...destroy itself !)
2. Hopefully **"unbreakable"**:  
nobody can know/modify what is inside.

## The Smart Cards Odyssey ?

Conclusion: for good security smart cards (or other secure hardware e.g. USB token) cannot be avoided ! **NECESSARY INGREDIENT !**

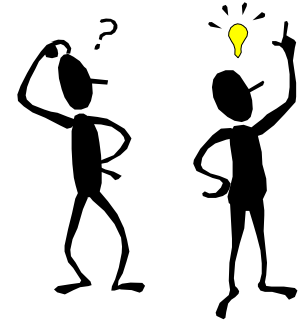


Deployment of PK-based solutions has just started, examples:

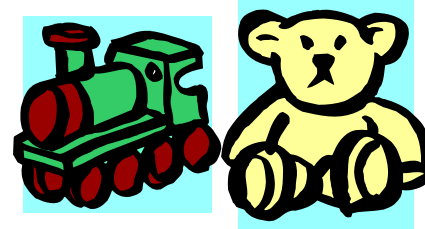
- EMV bank cards...
- Electronic passports and ID cards...
- Corporate badge with PC login and PGP functionality [Pentagon, Microsoft, etc]

## [Formal Approach] Security of Smart Cards

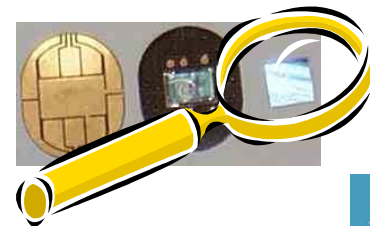
1. Adversarial Goal: the weakest possible:  
Obtain any information  
on the secret key  $K$ .



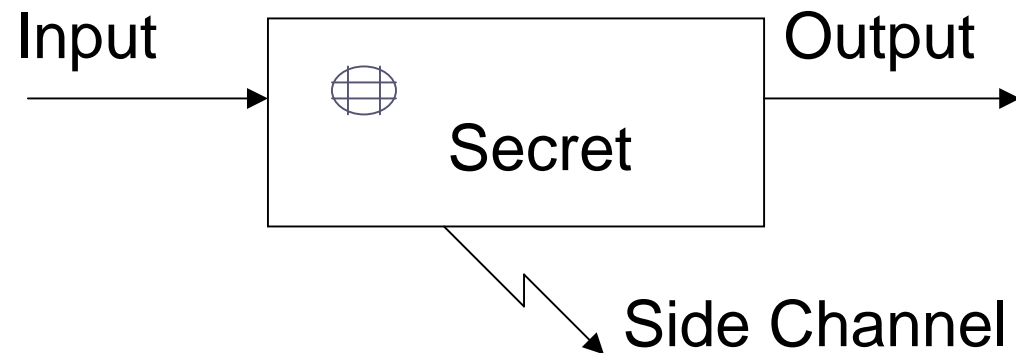
2. Against whom:  
reasonable hacker  
off-the shelf equipment.



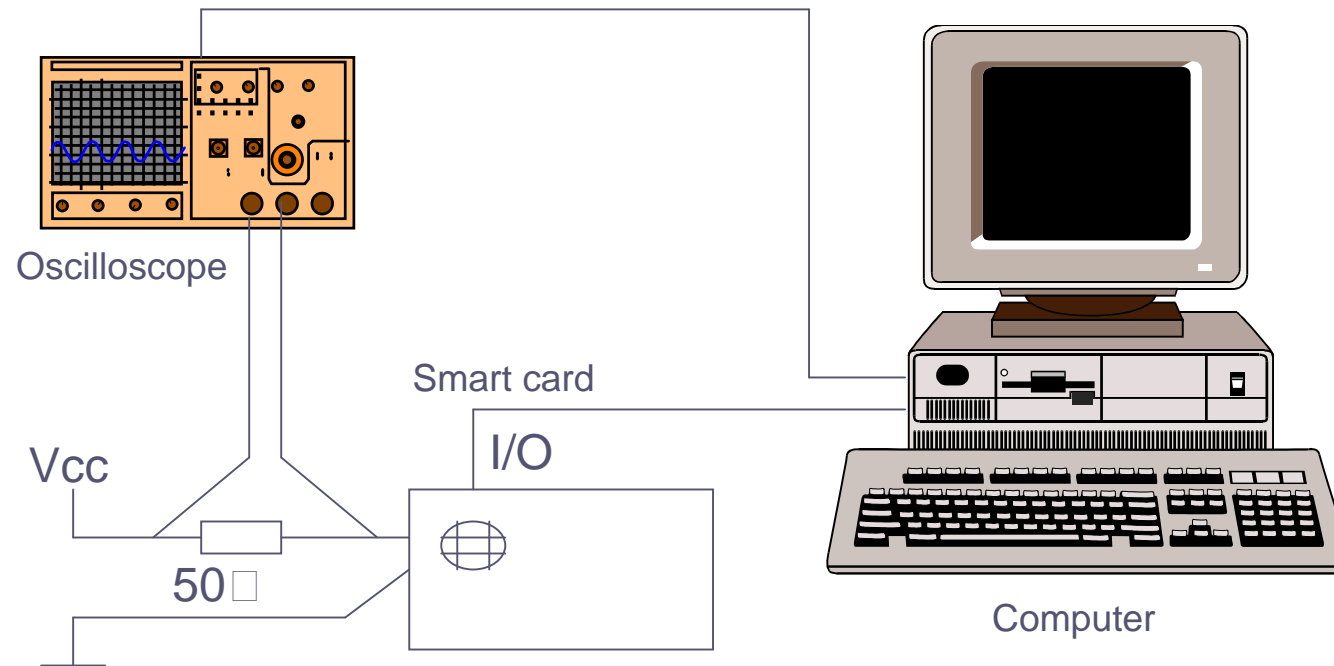
3. Access to the system:  
the device is totally in the  
hands of the Adversary



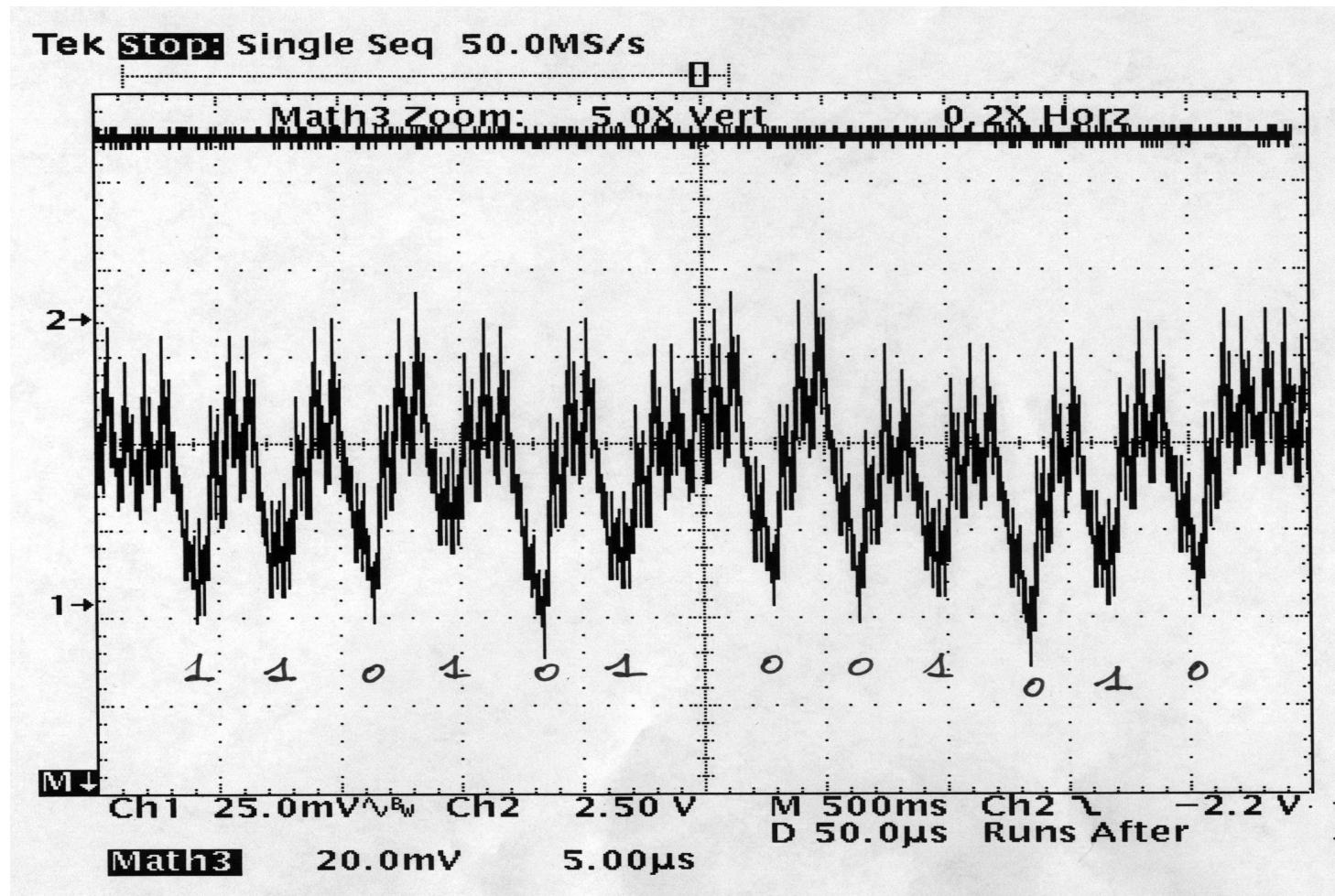
### 3. The secret is in the hands of the Adversary



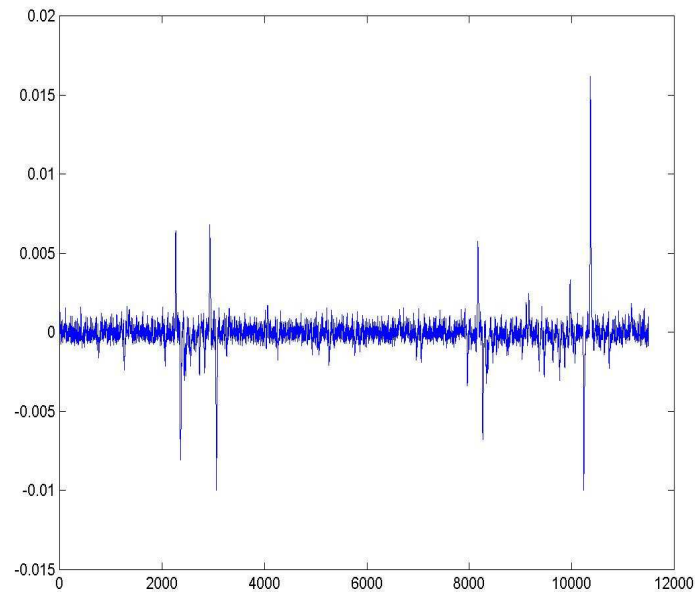
### 3. Example: Power Attacks



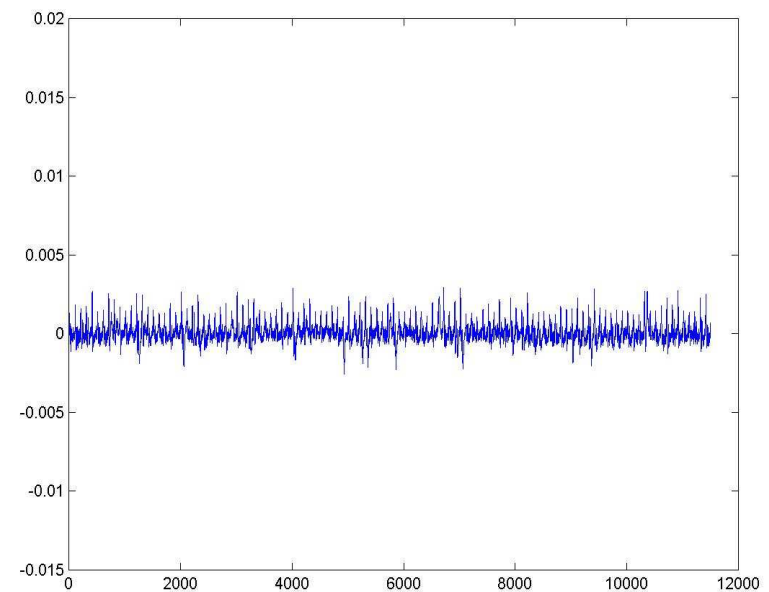
## SPA Attacks



## DPA Attacks



Right 6 bits



Wrong 6 bits

## DFA Attacks...

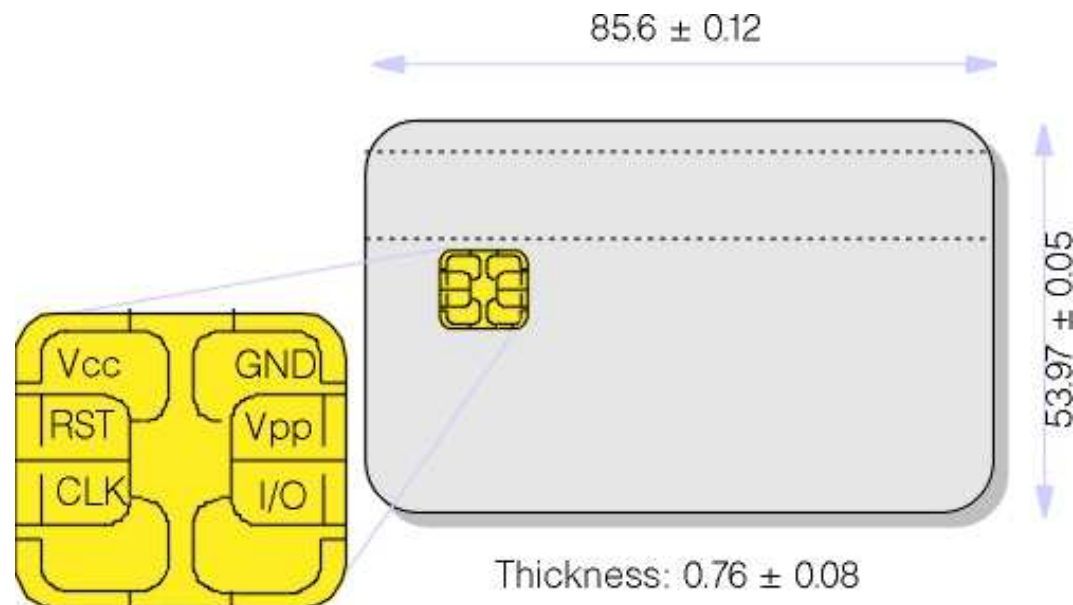
Provoke **Faults** in the device...

And then deduce the key by detailed  
mathematical analysis.



## How Secure are Smart Cards ?

Here also, the security is not granted.



## How Secure are Smart Cards ?

Many hardware and software counter-measures are implemented to prevent the electronic attacks.

- Additional cost.
- Many years of research...
- Hundreds of published papers.
- Many patents.

## Future: Really Good Data/IT Security:

1. Public key solutions are a MUST.
  - Will slowly become ubiquitous.  
PK crypto everywhere !
2. Consequence: Secure Hardware Devices are a MUST (keep private thing private).

All these developments are ahead.

PK crypto everywhere?

- this potential remains 98 % unused
  - cost...