Digital Signatures

Nicolas T. Courtois
- University College of London

Roadmap

- Legal aspects
- What are Digital Signatures ?
- How Secure they are ?
- Main realizations known
- Applications

1.
What is a [Digital] Signature ?

Legal Aspects

Vocabulary

frequently confused

crypto only

•crypto* - a D.S.

•secure device

•qualified certif.

Digital Signatures

$\subset$

Advanced Electronic Signatures.

$\subset$

Electronic Signatures.

just some electronic tag/evidence…

Electronic Signatures

Idea: some electronic data associated to an electronic document that proves (?) sth. (not much)…

Goal: Electronic records and signatures should be admissible in court. Can even be just a PIN code (!). How strong are solutions and in what context secure enough – different problem. Usually admitted, have to challenge them in court.

Electronic Signature: Def:

Definition [US]: an electronic sound, symbol, or process, attached to or logically associated with a record and executed or adopted by a person with the intent to sign the record. [Uniform Electronic Transactions Act, US].

Definition [EU]: data in electronic form which are attached to, or logically associated with, other electronic data and which serve as a method of authentication.

=> (apparently no "intent")

Digital Signature.

Idea: cryptographic technique.

Definition: 3 algorithms…

Security Goals/Properties: Message Authenticity, Unforgeability, Non-repudiation, Third-party Verifiability…

The European Directive on Electronic Signatures

The European Directive of December 13, 1999

Main goals:

- free movement of signatures between the EU countries to accompany free movement of goods and services.
- Recognition as evidence in court.

Effect: Member states are required to implement the Directive => translate into national law.

Electronic and Advanced Signatures (in The European Directive)

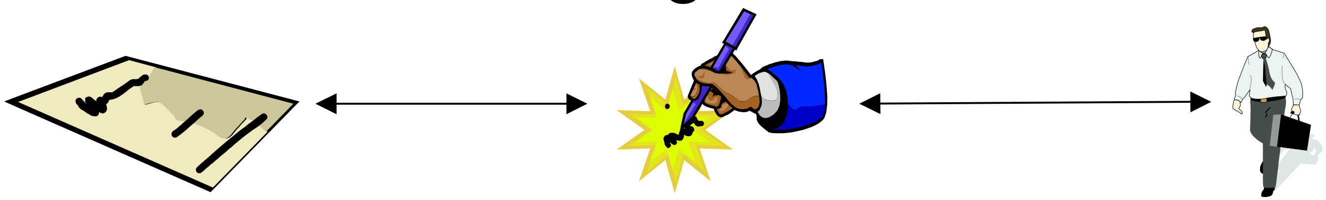
1. Electronic Signature.

Definition [EU]: data in electronic form which are attached to, or logically associated with, other electronic data and which serve as a method of authentication.

=> (apparently no "intent" like in the US)

2. Advanced Electronic Signature.

2x link.



An electronic signature that:

- is uniquely linked to a signatory and capable of identifying the signatory, and created by means the signatory can maintain under his sole control,
- and linked to the data being signed such that any change of the data is detectable.

Electronic == Handwritten ?

Equivalence (as strong in terms of law) under two conditions:

1. Produced by a secure signing device.

   [hardware device !]

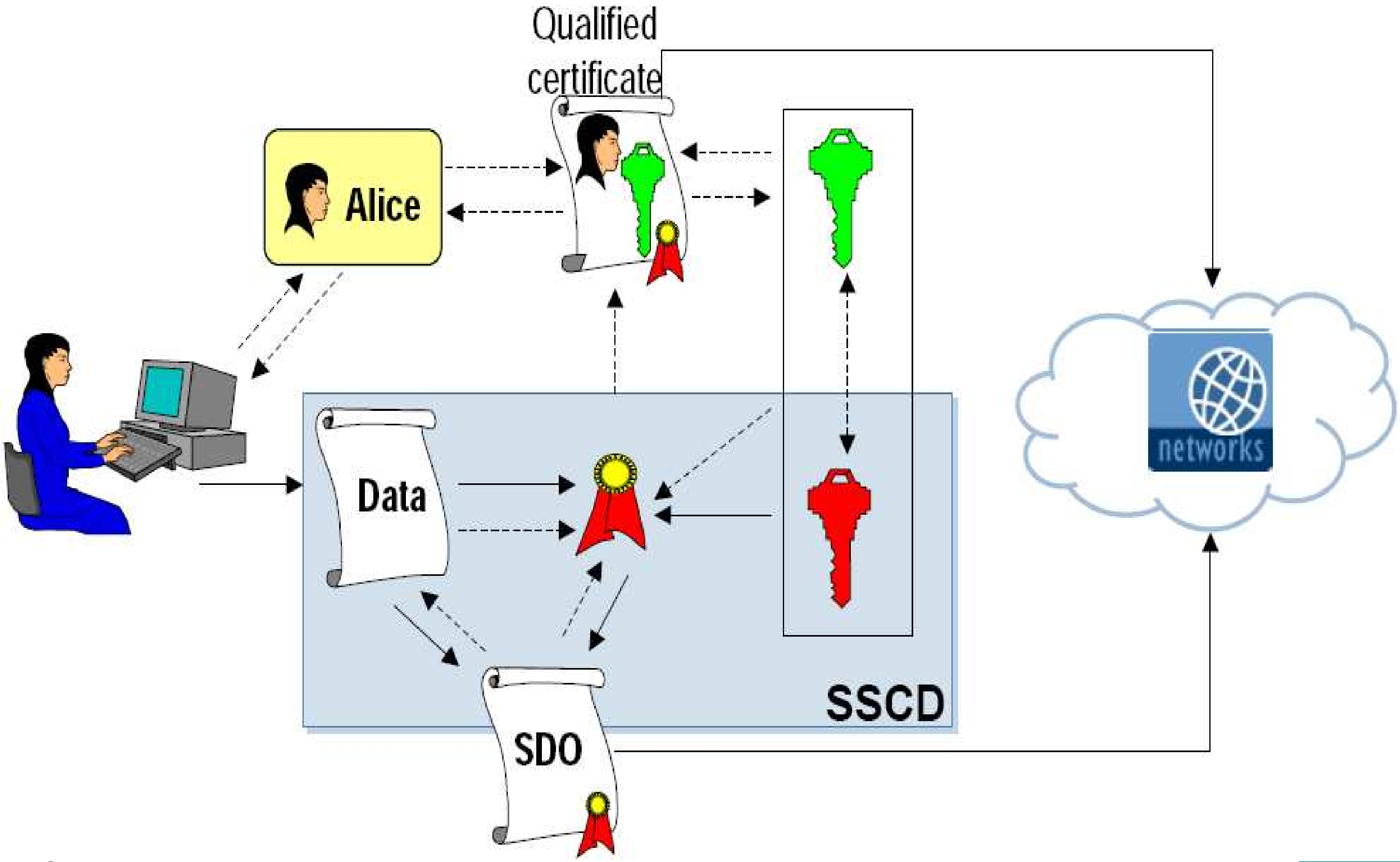
2. Based on a qualified certificate.

"Advanced Signature" a.k.a. "Qualified Signature"

Is it normal, good or bad ?

Handwritten signatures can be "perfectly" imitated as well. In some aspects electronic signatures are much more secure…

SSCD = Secure Signature Creation Device



The European Directive on Electronic Signatures

CSPs = Certification Service Providers
more than just CA (Certification Authorities).

- They have the right to issue QC (Qualified Certificates) on some territory.
  - QC can contain arbitrary limitations provided standardized/recognized [e.g. <= 1000 €].
- CSPs are LIABLE for damage (for negligence e.g. to revoke) - potentially huge liability !.
  - ⇒ have to implement tough [physical,IT,…] security.
  - ⇒ Explains why one has to pay for signatures… (e.g. 50 £ per year for a string of bits…).

(<u>Technical solution:</u> (not done) rely on several CAs, check all the certificates. Impossible to corrupt everyone…)

Electronic Signatures in the UK

EU Directive => Translation into national law.

1. The Electronic Communications Act 2000.
   - Section 7(1). Electronic signatures are admissible in evidence about the authenticity or integrity of a communication or data.
2. The Electronic Signatures Regulations 2002 (SI 2002 No. 318).
   - Regulation 3: QC and CSPs.

[Manual and Digital] Signatures

Two main functions:

1. Identify the signer
2. Approbation of the document.

Manual ≠ Digital Signatures

Two main functions

1. Identify the signer
2. Approbation

...in electronic word:

1. Easy to copy !
2. Easy to alter the document !

Consequence => A digital signature does depend on the document.

(need to protect document integrity, did not exist before !)

Digital Signatures

Three main functions?

1. Identify the signer (solved)
2. Approbation (*not easy*…)
3. Integrity of the message (solved)

Requirements so far:

Three main functions:

1. Identify the signer (solved)
2. Approbation (*not easy…*)
3. Integrity of the message (solved)

Digital Signatures - Bonus

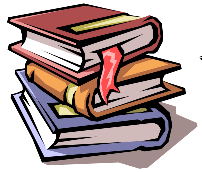
Another main function !

1. Identify the signer (certify origin, solved)
2. Approbation (hard to get !)
3. Integrity of the message (solved)
4. Automatic verification,
   and better:
   Public Verifiability
   (easy => became mandatory)

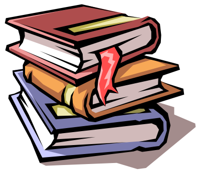
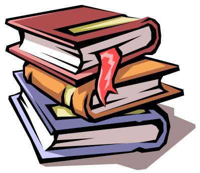
2.

Towards Technical Solutions

How These Problems are Solved ?



1. Identify the signer – doable
 => solved by crypto + trusted key infrastructure /PKI/ + secure hardware)
2. Approbation - hard

=> by crypto + law + policy + trusted hardware/software
3. Integrity of the message
  => solved by crypto only

4. Public Verifiability
  => solved by crypto only


How These Problems are Solved ?

1. Identify the signer

Non-repudiation:

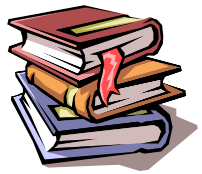
(French: Non-répudiation, Imputabilité).

The signer is the ONLY and UNIQUE person that can create the (signed) document.

Non-Repudiation (== “Imputability”)

The signer is the ONLY UNIQUE person that can create the document.

⇒ Existed already for manual signatures.



⇒ CAN ONLY BE DONE with PUBLIC KEY CRYPTOGRAPHY !

- ⇒ Impossible with DES or AES.

⇒ Secure hardware is ALSO NECESSARY

- ⇒ Impossible without a smart card (or other kind of trusted and closed hardware).

⇒ Source of trust necessary

- ⇒ One authentic public key: ROM, CD-ROM - sth. that cannot be altered.

3.

Cryptographic Signatures

***Message Authenticity – Goals

Different security levels:

1. Correct transmission – no (random) transmission error. A malicious attacker can always modify it.
   - Achieved with CRC and/or error [correction]/detection codes.
2. Integrity – no modification possible if the “tag/digest” is authentic. If we cannot guarantee the authenticity of the tag, a malicious attacker can still modify and re-compute the hash.
   - Achieved with cryptographic hash functions (= MDC). (e.g. SHA-1).
3. Authenticity – specific source. Authentified with some secret information (key).
   - Achieved with a MAC (= a hash function with a key = a secret-key signature).

4a. Non-repudiation – very strong requirement. Only one person/entity/device can produce this document.
   - Achieved with Digital Signatures. The strongest method of message authentication.

4b. Public verify-ability. Everybody can be convinced of the authenticity (trust the bank ?).
   - Achieved with Digital Signatures. The strongest method of message authentication.

Digital Signatures vs. Authentication

- Strongest known form of Message Authentication.
- Allows also authentication of a token/device/person (e.g. EMV DDA, US Passport):
  - challenge –response (just sign the challenge)
- The reverse does not hold:
  - Not always possible to transform authentication into signature. More costly in general !

Sym. encryption << P.K. authentication < signature



**Signatures

Can be:

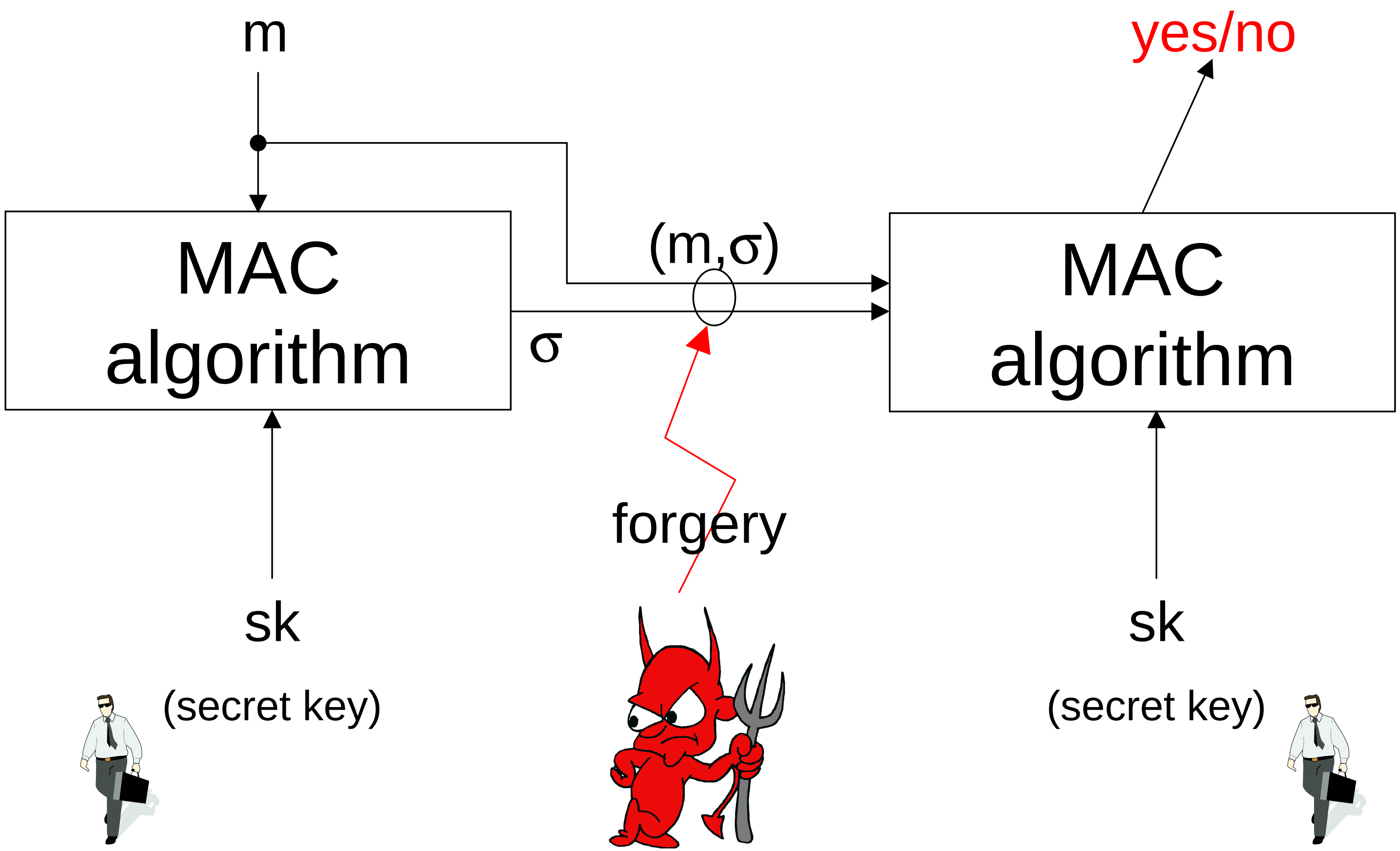
Public key:

- Real full-fledged digital signatures.

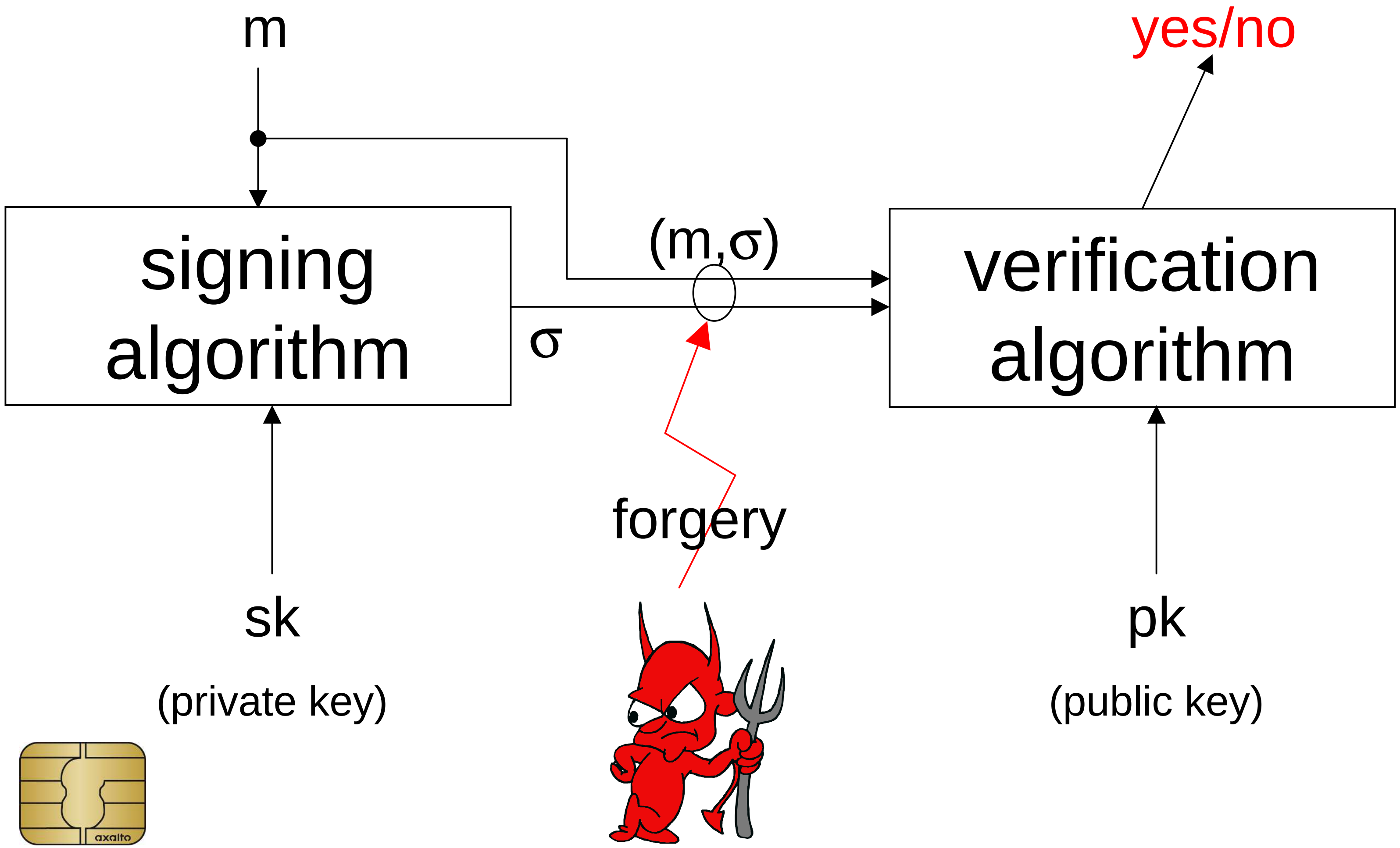
Secret key:

- Not « real signatures » but MACs.
- Widely used in practice, OK if you trust the verifier…

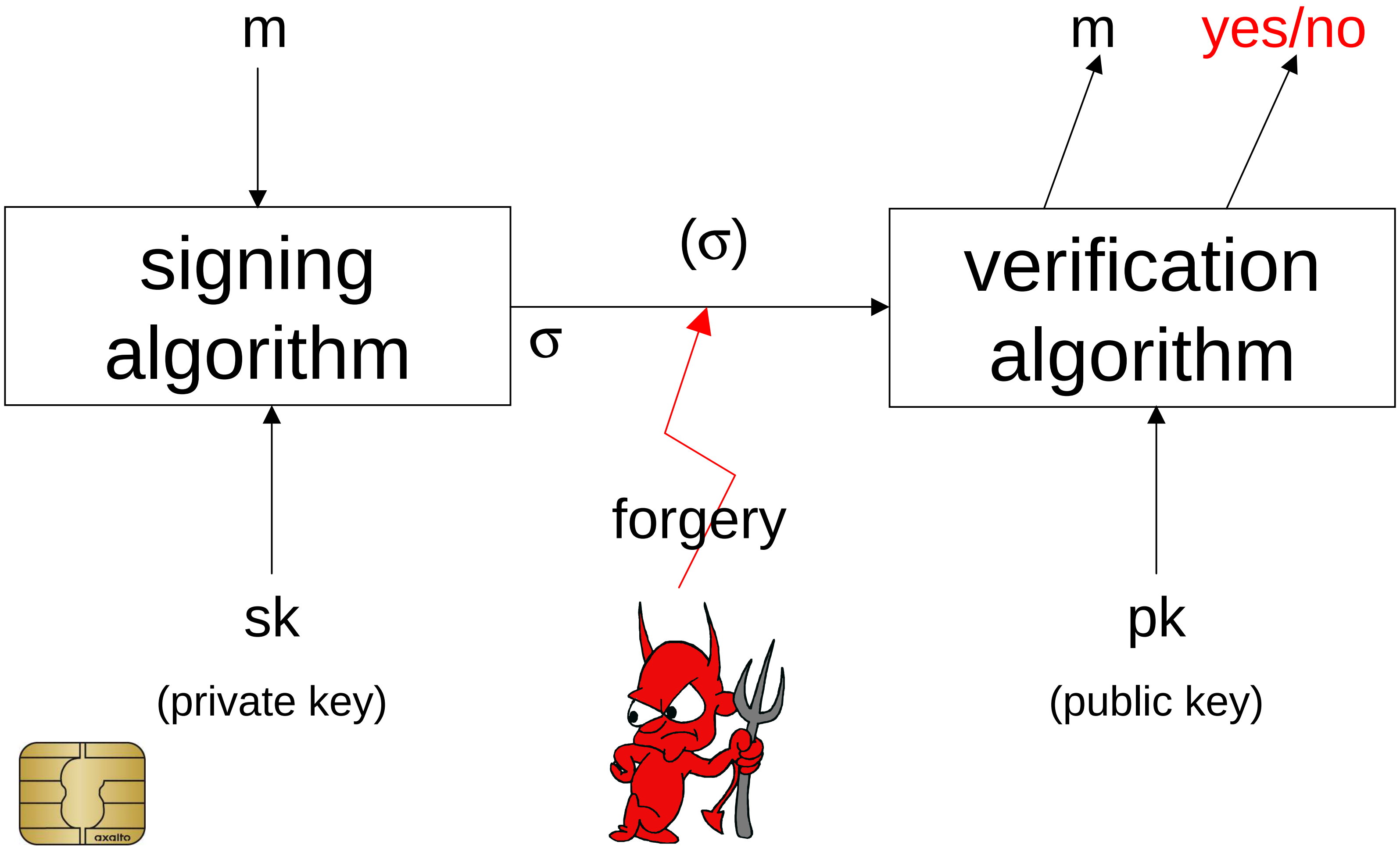
MACs = "Secret-Key Signatures"



Digital Signatures



Digital Signatures with Message Recovery



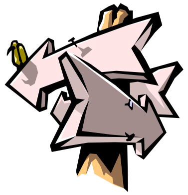
****Signatures - Requirements

1. Authenticity – guarantees the document signed by…
2. Non-repudiation – normally only possible with public-key signatures.
3. Public verify-ability - normally only possible with public-key signatures.

4. How to Do It Right ?

Until around 2001, nobody knew exactly !

Some international standards were broken.



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.

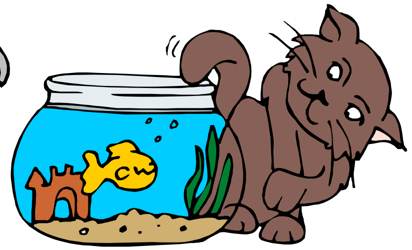
There is no other way known.

Many security notions, but…

Take the STRONGEST POSSIBLE version:

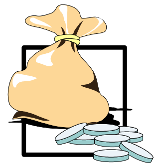
1. Adversarial Goal.

the weakest possible !



2. Resources of the Adversary:

The strongest possible: 10 G$.



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

Secure Public Key Signature

The "good" definition [Golwasser-Micali-Rivest 1988]:

[Strong] EUF - CMA (Existential Unforgeability under CMA)

1. Adversarial Goal.

   Find any new pair (m,σ) (new m)!

   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.

$m_i \longrightarrow$

$\longleftarrow \sigma_i$

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

A scheme is $(\overset{2.}{T}, \overset{1.}{\varepsilon})$ $\overset{3.}{\text{-UEF-CMA}}$ if…

Version 1: P vs. NP asymptotic security.

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

Version 2: Concrete security.

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

4.1.
First Try

Access (3.) - Basic Attacks on Signatures

Again assume that the public key is indeed known…

- Public Key Only === a.k.a. Key Only Attack.
- Known Message Attack. Access to several pairs (m,σ).
- Directed [==Non-Adaptive] Chosen Message Attack. (DCMA).
  - Single Occurrence Chosen Message Attack. (SOCMA).
- Fully Adaptive Chosen Message Attack. (CMA).

Textbook RSA Signature

- Signature: $\sigma = m^d$.
- Verification: $m \stackrel{?}{=} \sigma^e$.

Never use it.

What do We Sign ? The Problem:

Public key crypto is <u>very slow</u>.

Sign a long message with RSA, impossible, even on a 4 GHz CPU !

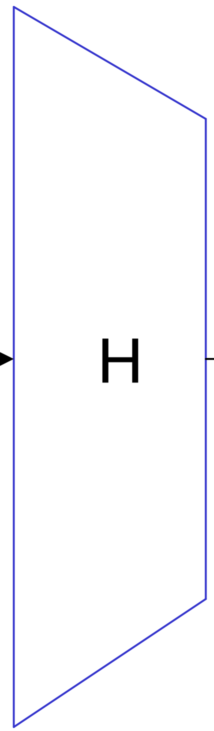
⇒Use hash function.

⇒Sign a short « digest » of the message.

[Cryptographic] Hash Function:

m

A hash function (or hash algorithm) is a reproducible method of turning data (usually a message or a file) into a number suitable to be handled by a computer. These functions provide a way of creating a small digital "fingerprint" from any kind of data. The function chops and mixes (i.e., substitutes or transposes) the data to create the fingerprint, often called a hash value. The hash value is commonly represented as a short string of random-looking letters and numbers (Binary data written in hexadecimal notation).



H(m)

A94A8FE5
CCB19BA6
1C4C0873
D391E987
982FBBD3

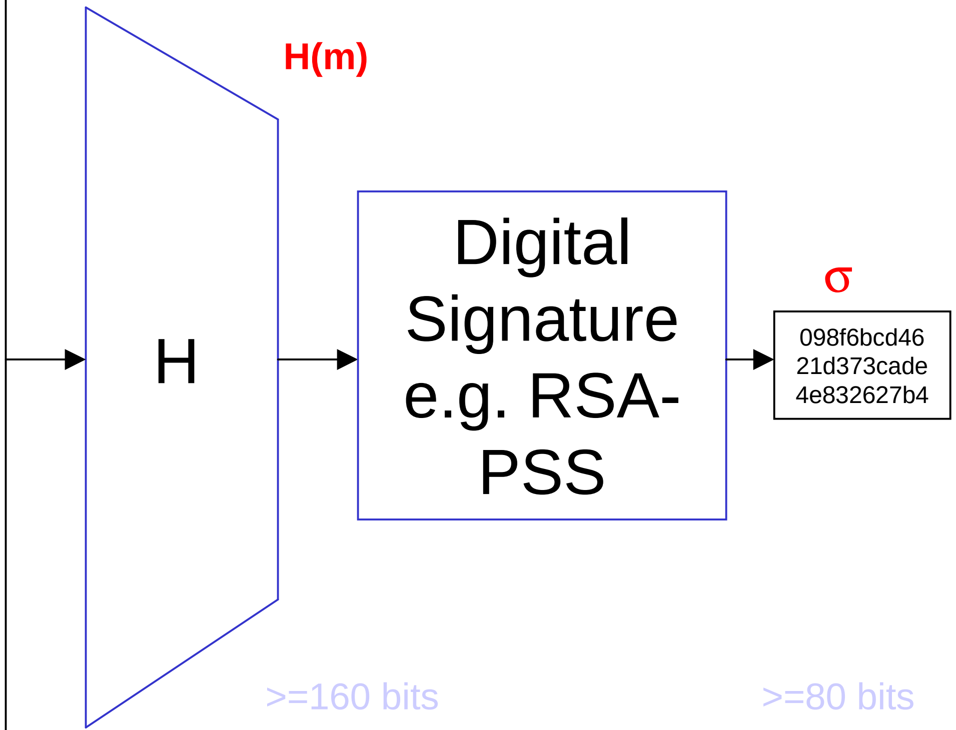
>=160 bits

0-∞ bits

Hash-then-Sign

m

A hash function (or hash algorithm) is a reproducible method of turning data (usually a message or a file) into a number suitable to be handled by a computer. These functions provide a way of creating a small digital "fingerprint" from any kind of data. The function chops and mixes (i.e., substitutes or transposes) the data to create the fingerprint, often called a hash value. The hash value is commonly represented as a short string of random-looking letters and numbers (Binary data written in hexadecimal notation).



0-∞ bits

Full Domain Hash RSA Signature

- Signature: $\sigma = H(m)^d$.
- Verification: $H(m) \mathrel{?=} \sigma^e$.

Please use it.

Provably secure ("tight" security).

Slight problem:

- There is no standardised hash function that produces a hash on 1024 or 2048 bits.
- So RSA-FDH is not very widely used.

5.

Best Known Techniques

How Secure Are Secure Signatures ?

All these are necessary ingredients:

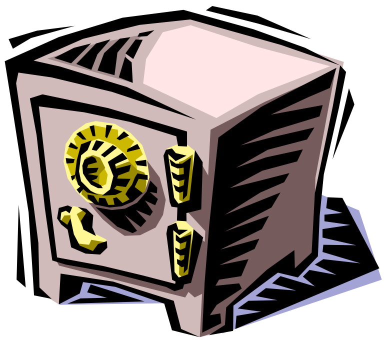
- Secure signing environment (know what you sign).
- Secure hash function.
- Secure PK cryptographic system
  (e.g. RSA) - key size !
- Secure padding! Many were broken
  => provable security.
- All this protected against
  side-channel attacks.
- A complete certification chain:
  all data have to be certified
  (e.g. the elliptic curve a, b, p,G, etc…).
- Source of trust: have one trusted key
  (e.g. in ROM).

c
r
y
p
t
o

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 security to a hard problem.

Possible ?:

Became possible PRECISELY BECAUSE we understood what is a secure digital signature.

[GMR88 definition]

Textbook RSA Signature

- Signature: $\sigma = m^d$.
- Verification: $m \stackrel{?}{=} \sigma^e$.

Never use it.

Provable Security – Recommended Solutions

Signature (easier):

- RSA-PKCS #1 v1.5. insecure (no proof yet, not broken, variants broken)
  - (exists also in PKCS #1 v2.0 and 2.1 cf. www.rsasecurity.com)
- RSA-FDH: perfectly OK. Except how to find hash function on 2048 bits ?
- RSA-PSS: current recommended standard, part of PKCS #1 V.2.x.
  - The best method to sign with RSA >=1024 bits

Hash functions broken =>:

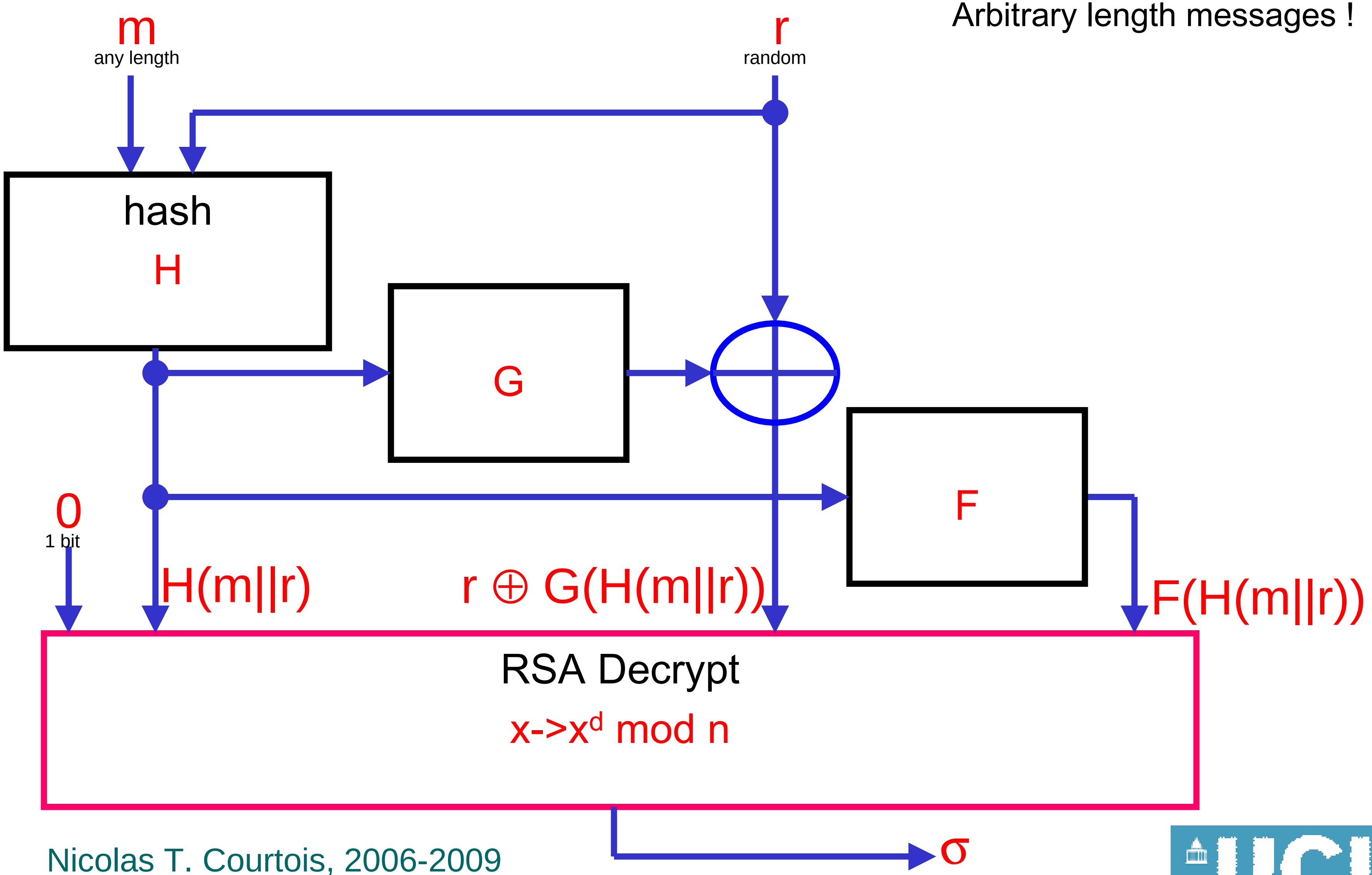
- Very serious for signing exe, doc ,pdf, ps, and other complex formats.
- Not serious AT ALL for signing messages in simple text.

BTW. Recall that CR is not necessary for digital signatures [UOWHF, Boneh result]. Nobody uses this unhappily…

Probabilistic Signature Scheme [Bellare-Rogaway'96]

Uses a hash function H and two one-way functions F and G.

Arbitrary length messages !



Provable Security - Example:

Any attack on RSA-PSS

=> Extract e-th roots mod N.

Secure Signatures – Time Scale

Time to break

Authentication: 1 hour. After it is too late !

Signature: 20 years and more…

Must think about future attacks !

E.g. EMV cards: almost certainly broken
due to the key sizes, 1024 bits @ year 2010.

Further Security

Use timestamping,
or forward-secure D.S.
or destroy the private key.

**But is it hard ?

Any attack on RSA-PSS => Extract e-th roots mod N.

Does not imply factoring !

(nobody knows if there is a difference..)

Guarantees Solution…

- If one can factor RSA-2048 bits, RSA Security offers 200 000 US$.
- Breaking Elliptic Curves: 725 000 $.

=>nobody can claim these are broken…

- BTW. Not even 1 dollar for AES…

6.

Signature Schemes in Practice

Some Signature Schemes

✓RSA-PSS

- ✓ RSA-OAEP – only with long keys [>4096 bits]

✓DSA.

- ✓ Main DSA standard out of date, 80-bit security.
- ✓ Switch to ECDSA – Elliptic Curve, recommended.

✓~~Sflash~~, Quartz [Patarin, Goubin, Courtois]

broken in 2007

Some Signature Schemes on a Smart Card

Cryptosystem	~~SFLASH~~	NTRU	RSA-1024	RSA-1024	ECC-191
Platform	~~SLE-66~~	Philips 8051	SLE-66	ST-19X	SLE-66
ROM [Kbytes]	~~3.1~~	5	NA	NA	NA
Frequency [MHz]	~~10~~ **broken in 2007**	16	10	13	10
Co-processor	~~no~~	no	no	yes	yes
Length of S [bits]	~~259~~	1757	1024	1024	382
Timing [ms]	~~59~~	160	many s	111	180
Timing × Frequency	~~590~~	2560	big	1443	1800

Which One Should Use ?

NSA suite B [2005]:

http://www.nsa.gov/ia/industry/crypto_suite_b.cfm

- ECDSA + SHA-256.

⇒The NSA has acquired a licence for 23 Certicom patents. Can sub-licence.

⇒RSA is no longer recommended !

⇒DSA is dead too.

Cheap Alternative:

RSA-PSS 2048 bits.

- No patents.
- OK if you have enough computing power and RAM…

Signatures

1. MACs are widely used, 100s of times faster. Yet symmetric => fundamentally not very secure…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).

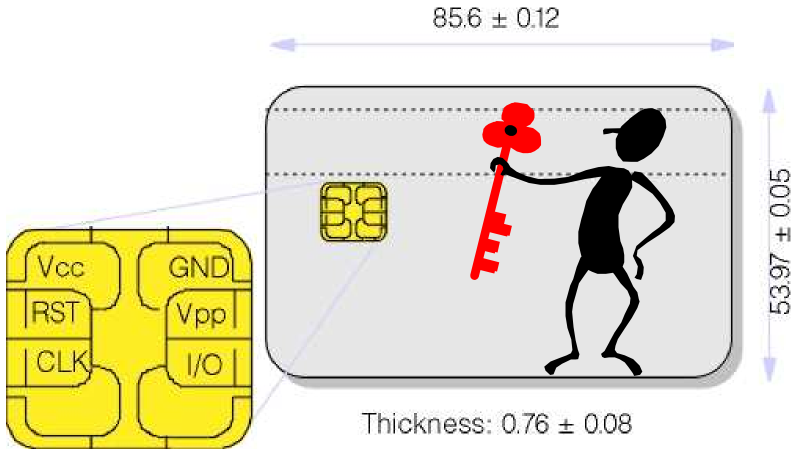
All these developments are ahead. Very little of this is in fact used today…

Secure Hardware Devices

KEEP private keys private all the time !

Must be securely

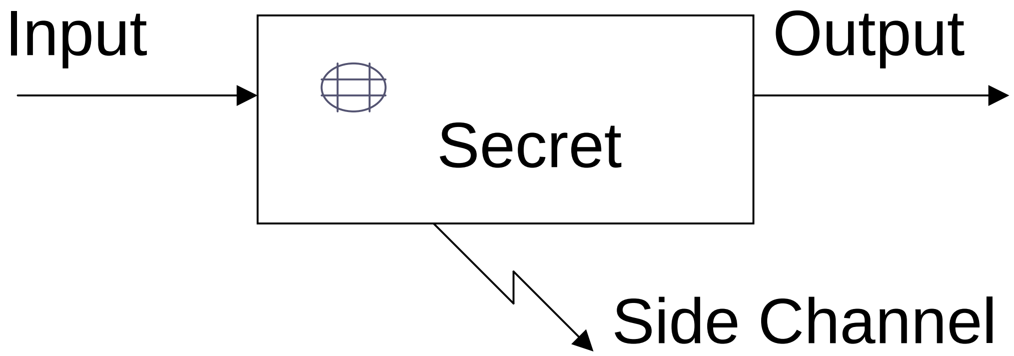
- Generated
- Stored
- Used
- Backup
- Destroyed



- No real security with a PC.

- Example: Smart Cards.

Note: the cards must still be protected against channel attacks !



cost: +30 % ?

7.
Applications of Digital Signatures

Nicolas T. Courtois, 2006-2009

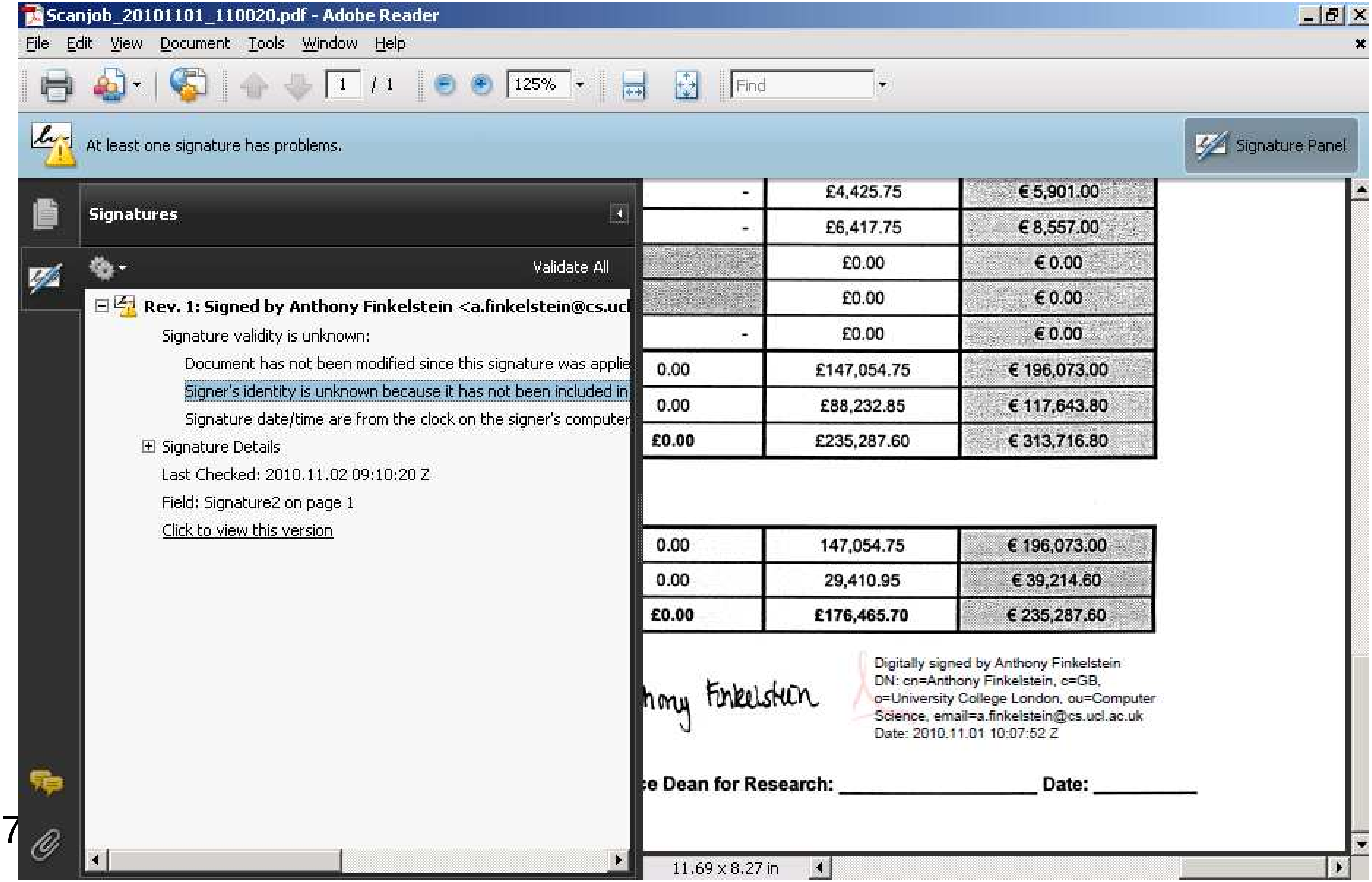
Main Applications of Digital Signatures

- Bank cards
- Web – SSL
- Software authentication (Microsoft, Java Card0 Apple, Google Aps, Nokia)

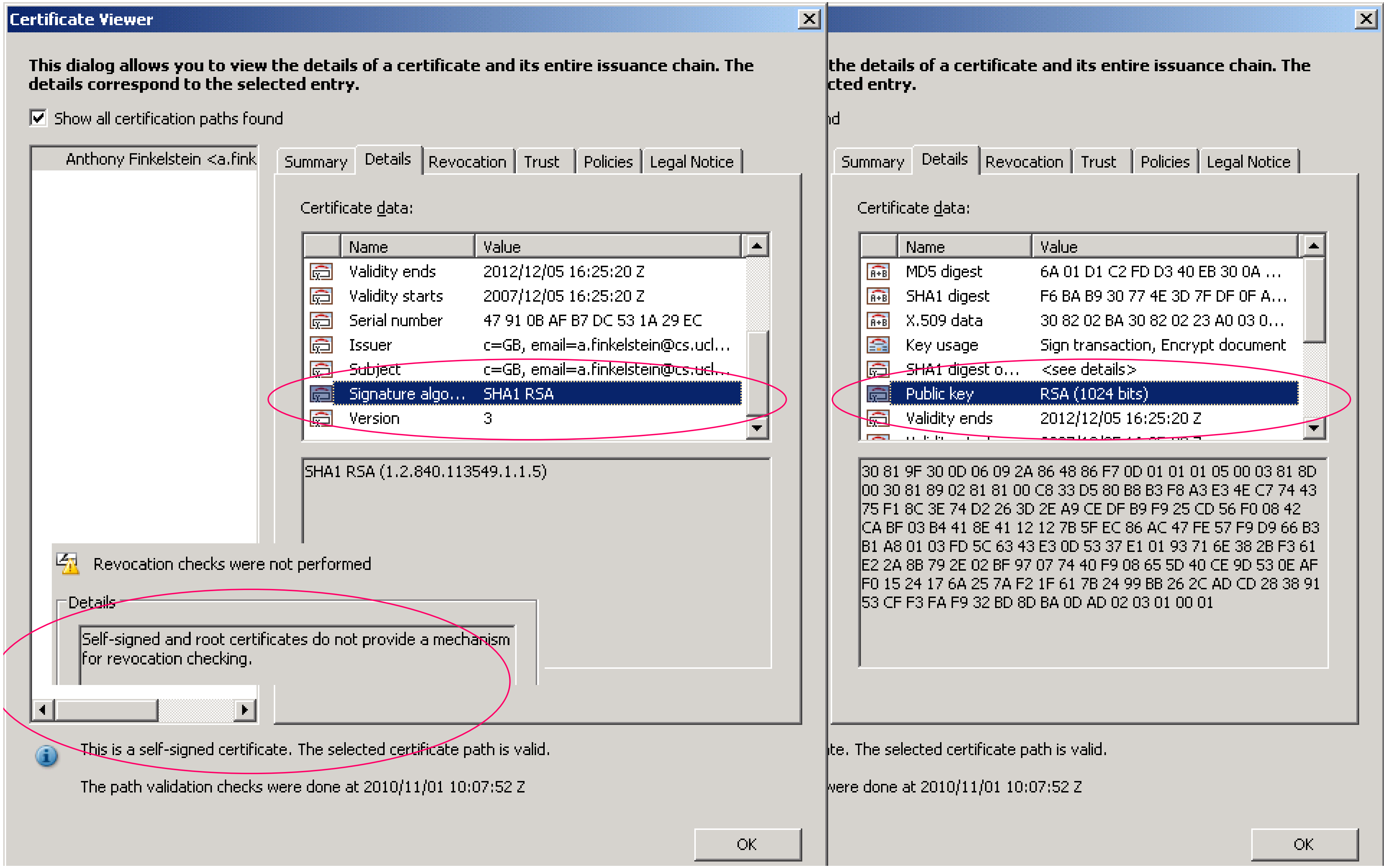
More Applications of Digital Signatures…

- e-ID cards, e-Passports
- All public key solutions (even encryption only !) require PKI, requires signatures !
- Secure email, authenticity and anti-spam
- Data and disk authenticity
- Signing notary acts
- Signing medical prescriptions: CPS signs data before sending to Caisse d'Assurance Maladie.
- Vitale 2 will sign when you buy medicines at a pharmacy shop.

Digitally Signed pdf @UCL



Is It Secure?


Certificate Viewer
This dialog allows you to view the details of a certificate and its entire issuance chain. The details correspond to the selected entry.
Show all certification paths found
Anthony Finkelstein <a.fink
Summary
Details
Revocation
Trust
Policies
Legal Notice
Certificate data:
Name
Value
Validity ends 2012/12/05 16:25:20 Z
Validity starts 2007/12/05 16:25:20 Z
Serial number 47 91 0B AF B7 DC 53 1A 29 EC
Issuer c=GB, email=a.finkelstein@cs.ucl...
Subject c=GB, email=a.finkelstein@cs.ucl...
Signature algo... SHA1 RSA
Version 3
SHA1 RSA (1.2.840.113549.1.1.5)
Revocation checks were not performed
Details
Self-signed and root certificates do not provide a mechanism for revocation checking.
This is a self-signed certificate. The selected certificate path is valid.
The path validation checks were done at 2010/11/01 10:07:52 Z
OK
the details of a certificate and its entire issuance chain. The
cted entry.
nd
Summary
Details
Revocation
Trust
Policies
Legal Notice
Certificate data:
Name
Value
MD5 digest 6A 01 D1 C2 FD D3 40 EB 30 0A ...
SHA1 digest F6 BA B9 30 77 4E 3D 7F DF 0F A...
X.509 data 30 82 02 BA 30 82 02 23 A0 03 0...
Key usage Sign transaction, Encrypt document
SHA1 digest o... <see details>
Public key RSA (1024 bits)
Validity ends 2012/12/05 16:25:20 Z
30 81 9F 30 0D 06 09 2A 86 48 86 F7 0D 01 01 01 05 00 03 81 8D
00 30 81 89 02 81 81 00 C8 33 D5 80 B8 B3 F8 A3 E3 4E C7 74 43
75 F1 8C 3E 74 D2 26 3D 2E A9 CE DF B9 F9 25 CD 56 F0 08 42
CA BF 03 B4 41 8E 41 12 12 7B 5F EC 86 AC 47 FE 57 F9 D9 66 B3
B1 A8 01 03 FD 5C 63 43 E3 0D 53 37 E1 01 93 71 6E 38 2B F3 61
E2 2A 8B 79 2E 02 BF 97 07 74 40 F9 08 65 5D 40 CE 9D 53 0E AF
F0 15 24 17 6A 25 7A F2 1F 61 7B 24 99 BB 26 2C AD CD 28 38 91
53 CF F3 FA F9 32 BD 8D BA 0D AD 02 03 01 00 01
te. The selected certificate path is valid.
were done at 2010/11/01 10:07:52 Z
OK