Elementary Cryptography

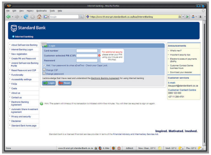
Welcome to the Quantum Era!

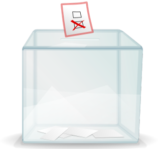
Serge Vaudenay

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

LASEC

Cryptography = Science of Information and Communication Security

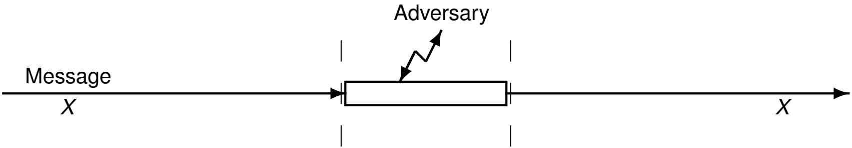




Cryptographic Problems in Ancient Time

- privacy (by encryption)

The Fundamental Trilogy



- **Confidentiality** (C): defeat malicious access to X
- **Authentication** (A): defeat malicious forgery of X
- **Integrity** (I): defeat malicious modification of X

A Few Cryptographic Problems

- privacy (by encryption)
- detection malicious modification of information
- data authentication
- access control
- timestamping
- fair exchange
- digital rights management
- more privacy (anonymity, unlinkability, deniability, ...)

Applications

- bank cards
- E-commerce
- mobile telephony
- e-passport
- mobile communication (Bluetooth, WiFi...)
- traceability, logistic & supply chains (RFID)
- pay-TV, DRM
- access control (car lock systems, metro...)
- payment (e-cash)
- electronic voting

What Can Be Assumed Secret?

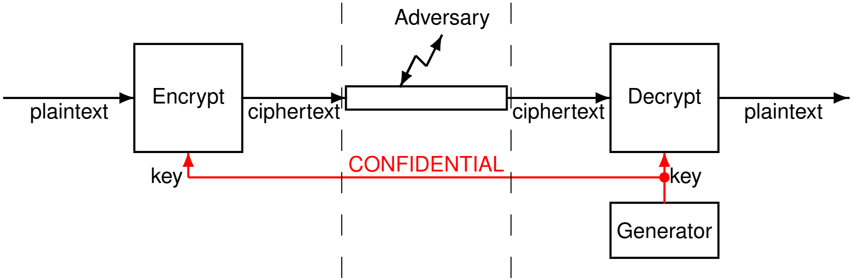
- to design a cryptographic system is a difficult task
- products (implementing cryptography) are massively deployed
- we cannot assume that adversaries ignore which cryptographic system is used
- we can assume the secrecy of a key, though
- **security by obscurity** can only fail!
  (i.e. using secret algorithms and relying on their secrecy)

Kerckhoffs Principles (1883)

1. Le système doit être matériellement, sinon mathématiquement, indéchiffrable;
2. **Il faut qu'il n'exige pas le secret, et qu'il puisse sans inconvénient tomber entre les mains de l'ennemi;**
3. La clef doit pouvoir en être communiquée et retenue sans le secours de notes écrites, et être changée ou modifiée au gré des correspondants;
4. Il faut qu'il soit applicable à la correspondance télégraphique;
5. Il faut qu'il soit portatif et que son maniement ou son fonctionnement n'exige pas le concours de plusieurs personnes;
6. Enfin, il est nécessaire, vu les circonstances qui en commandent l'application, que le système soit d'un usage facile, ne demandant ni tension d'esprit, ni la connaissance d'une longue série de règles à observer.

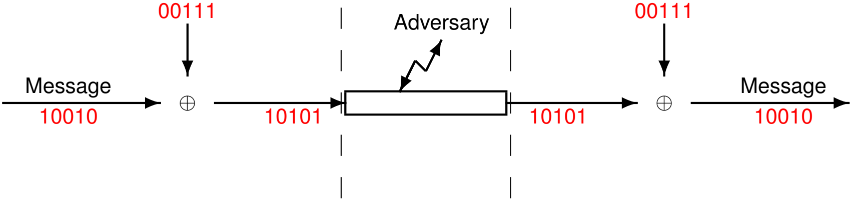
- meaning:
  security analysis must assume that the adversary knows the algorithms
- common misunderstanding:
  ~~algorithms must be public~~

Symmetric Encryption



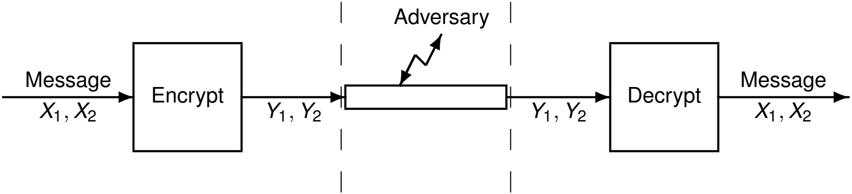
Vernam Cipher

⊕	0	1
0	0	1
1	1	0



Using the Same Key Twice is Bad

$$\begin{aligned} Y_1 &= X_1 \oplus K \\ Y_2 &= X_2 \oplus K \end{aligned}$$



$$Y_1 \oplus Y_2 = (X_1 \oplus K) \oplus (X_2 \oplus K) = (X_1 \oplus X_2) \oplus (K \oplus K) = X_1 \oplus X_2$$

leakage of the $X_1 \oplus X_2$ value

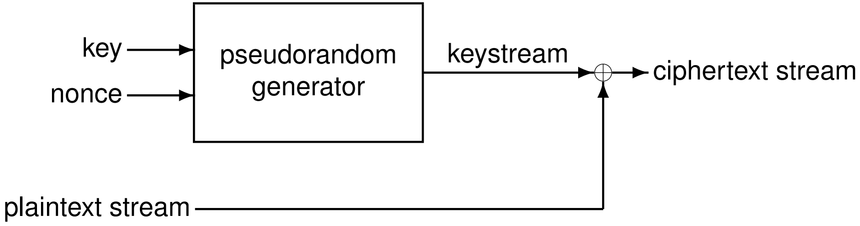
Information Theory

Claude Shannon

[Claude Shannon]

- formalized the notion of perfect secrecy
- the Vernam cipher (when correctly used) is perfectly secure
- perfect security implies that the key space is at least as large as the message space

Using a Pseudorandom Key: Stream Cipher



nonce = **n**umber which can be used **once**
(necessary to avoid re-using a keystream)

Kinds of Symmetric Encryption Schemes

- **stream cipher** (length-preserving, needs a nonce)
- **block cipher** (encrypts only 128-bit blocks)
- block cipher in a **mode of operation** (some length-preserving, some with nonces)

Inventory of Symmetric Encryption Schemes

wildlife: ARMADILLO BEAR BLOWFISH DRAGON FOX FROG LION MOSQUITO RABBIT SERPENT SHACAL SHARK TWOFISH

flora: CAMELLIA LILY SEED

pantheon: ANUBIS MARS KHAFRE KHUFU LUCIFER MICKEY SHANNON TURING

gastronomic: COCONUT GRANDCRU KFC MILENAGE PEANUT WALNUT

elements: CRYPTON ICE ICEBERG RAINBOW SNOW

eccentric: ABC ACHTERBAHN AKELARRE CAST DEAL DECIM EDON FEAL FUBUKI GOST HELIX HIEROCRYPT IDEA KASUMI KATAN KHAZAD KTANTAN LEX LEVIATHAN LOKI MACGUFFIN MADRYGA MAGENTA MIR MISTY NIMBUS NOEKEON NUSH PHELIX PRESENT PY QUAD REDOC RIJNDAEL SAFER SALSA SCREAM SFINKS SKIPJACK SMS4 SQUARE SOBER SOSEMANUK XTEA 3-WAY YAMB

uninspired: A5 **AES** BMGL C2 CJCSG CMEA CS-CIPHER DES DFC E0 E2 FCSR HPC MMB Q RC2 RC4 RC5 RC6 SC TSC WG

A 128-Bit Key

11000000	10010011	00000011	01001001
11010011	11110010	01111011	10100101
10101001	00110001	00110000	11011110
00101110	01001110	00011111	00100001

`c0930349 d3f27ba5 a93130de 2e4e1f21`

number of combinations:

$$\begin{aligned} & \overbrace{2 \times 2 \times 2 \times \cdots \times 2}^{\text{128 times}} \\ = \quad & 2^{128} \\ = \quad & \underbrace{340\,282\,366\,920\,938\,463\,463\,374\,607\,431\,768\,211\,456}_{\text{39 digits}} \end{aligned}$$

Exhaustive Search on 128 Bits?

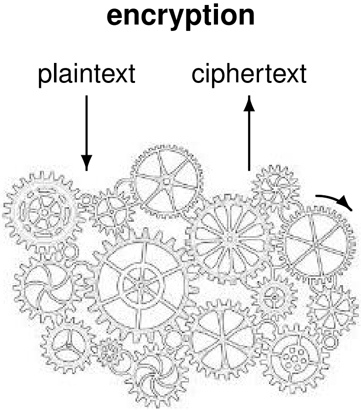
Order of Magnitude of 2^{128}

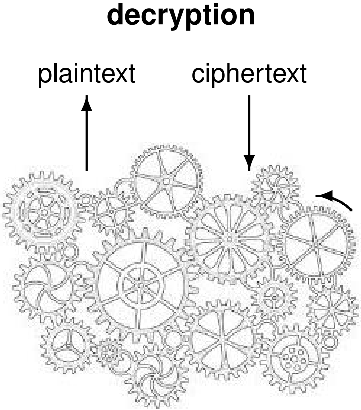
some big numbers:

- human population: 2^{33}
- number of cells in a human body: 2^{47}
- age of the universe: 2^{59}s
- number of atoms in 12g of carbon: 2^{79}
- diameter of the universe: 2^{90}m (2^{123}Å)
- mass of Earth: 2^{93}g ($\approx 2^{114}$ amoebas)
- number of atoms in the universe: 2^{266}

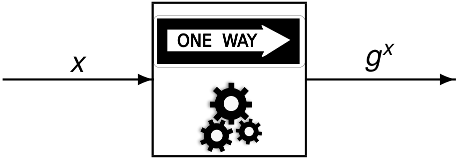
in 2007, a standard PC could test 1 000 000 keys per second
to test 2^{128} within 15 Billion years, we need 720 000 Billion of 2007-PCs!

Reversibility in Symmetric Encryption



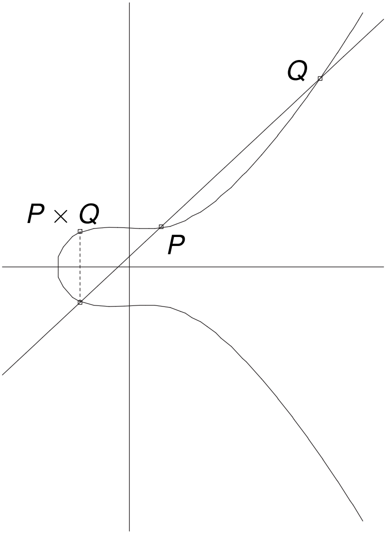


Hard-To-Invert Computation



in some algebraic structures, log is an intractable operation

Multiplication in an Elliptic Curve



$$y^2 = x^3 + ax + b$$

Multiplication Easy $\Longrightarrow$ Exponentiation Easy

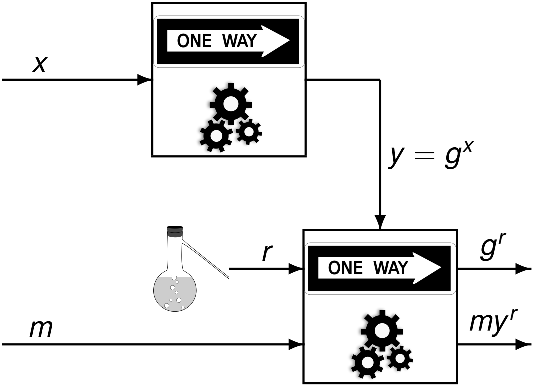
Theorem

We can compute g^x with $2n$ multiplications or less, for $x < 2^{n+1}$.

example: g^{54} in 8 multiplications

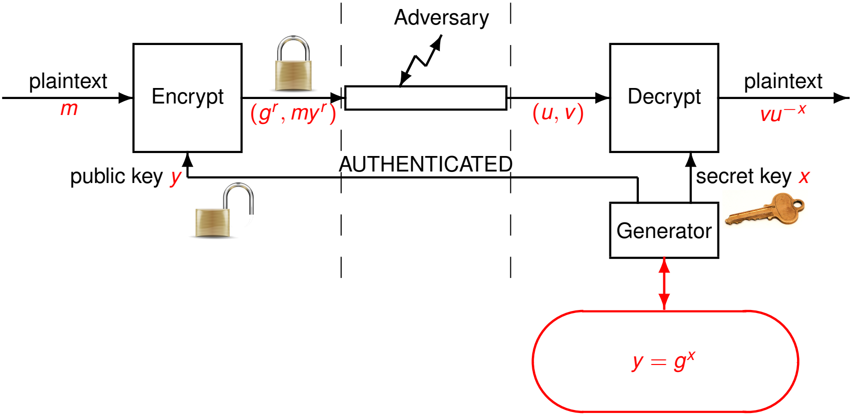
	54	squaring	multiplying
1		g	
2	$= 2$	$g \times g = g^2$	g^2
4	$+4$	$(g^2) \times (g^2) = g^4$	$\times g^4$
8		$(g^4) \times (g^4) = g^8$	
16	$+16$	$(g^8) \times (g^8) = g^{16}$	$\times g^{16}$
32	$+32$	$(g^{16}) \times (g^{16}) = g^{32}$	$\times g^{32}$
			$= g^{54}$

Encryption with a Public Key

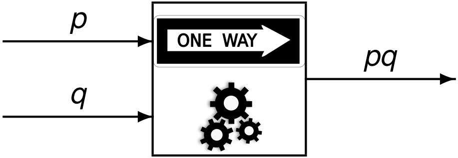


trick: x is a trapdoor allowing to decrypt!
$m = (my^r)/(g^r)^x$ because $(g^r)^x = (g^x)^r$

A Public-Key Cryptosystem (ElGamal)



A Familiar Hard-To-Invert Computation



Factoring Record

complexity: $e^{\mathcal{O}\left((\ln n)^{\frac{1}{3}}(\ln \ln n)^{\frac{2}{3}}\right)}$

RSA200

= 27997833911221327870829467638722601621070446786955
428537560009929326128400107609345671052955360856061822351910951365788637105954482006576775098580557613579098734950144178863178946295187237869221823983
= 35324619344027701212726049781984643686711974001976
25023649303468776121253679423200058547956528088349
$\times$
79258699544783330333470858414800596877379758573642
19960734330341455767872818152135381409304740185467

factored in 2005 with the equivalent of 55 years on a PC 2.2GHz

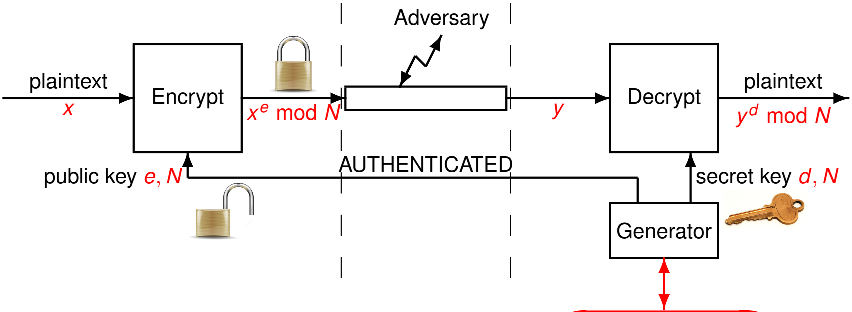
Rivest-Shamir-Adleman (RSA)

(1978)

[Shamir, Rivest, Adleman]

- concrete **trapdoor permutation**
  (invertible transformation which is easy to compute in one direction but hard in the other, but with the knowledge of trapdoor information)
- $\longrightarrow$ **public-key cryptosystem**
- $\longrightarrow$ **signature scheme**

(Textbook) RSA



p, q, prime numbers

$N = pq$

$\varphi(N) = (p-1)(q-1)$

$1 = \gcd(e, \varphi(N))$

$d = e^{-1} \bmod \varphi(N)$

Theorem (Euler)

$$\forall x \in \mathbf{Z}_N^* \quad x^{\varphi(N)} \bmod N = 1$$

Public-Key Cryptosystems

- **RSA**
- Rabin
- Paillier

} based on factoring

- **ElGamal**
- ECC
- HECC

} based on discrete logarithm

- NTRU
- lattice-based
- McEliece
- TCHo

} "post-quantum"

Symmetric vs Public-Key Cryptography

symmetric

- link-based
- fast
- cheap
- robust

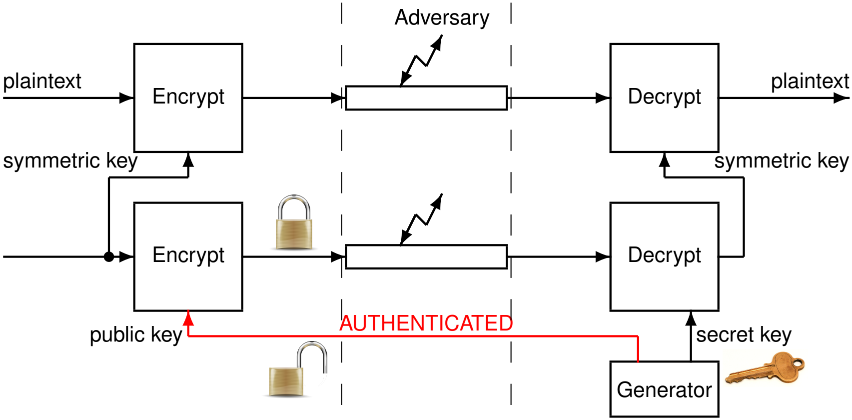
public-key

- user-based
- for short messages
- expensive
- sensitive

hybrid

- public-key crypto is used to establish a short-term symmetric key
- symmetric crypto is used to process the data

Hybrid Encryption



Diffie-Hellman

“New Directions in Cryptography” (1976)

[Merkle, Hellman, Diffie]

- invention of **public-key cryptography**
- notion of “**trapdoor permutation**”
- building a **public-key cryptosystem** from it
- building a **digital signature scheme** from it
- **key agreement protocol**

Diffie-Hellman Protocol

notion from Algebra (could be an elliptic curve)

with a group generated by some g

Alice		**Bob**
pick x at random		
$X \leftarrow g^x$	$\xrightarrow{X}$	pick y at random
	$\xleftarrow{Y}$	$Y \leftarrow g^y$
$K \leftarrow Y^x$		$K \leftarrow X^y$

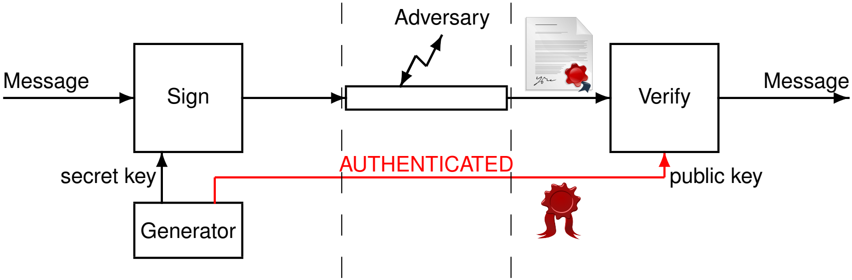
$$(K = g^{xy})$$

security requirement: given (g, g^x, g^y), it must be hard to compute g^{xy} (**Computational Diffie-Hellman Problem**)

Key Agreement

- **key agreement**
  resist passive attacks
  vulnerable against man-in-the-middle attacks
- **authenticated key agreement**
  resist active attacks
  needs some prior authenticated information
  (e.g. public key, secret, password)

Digital Signature: Encryption Upside Down!



Signature Schemes

- **RSA**
- Rabin

} based on factoring

- ElGamal
- Schnorr
- **DSA**
- **ECDSA**

} based on discrete logarithm

- NTRU
- lattice-based

} "post-quantum"

Signature: From Paper to Bits

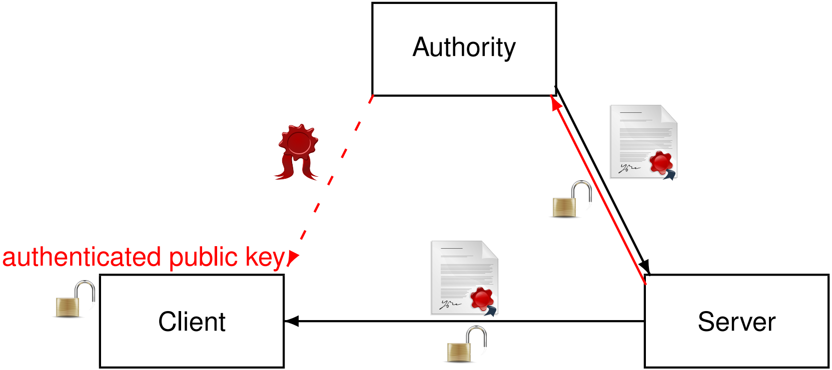
paper signature

- hard to copy
- same signature
- verified with a model
- needs human effort
- photocopies are non-binding

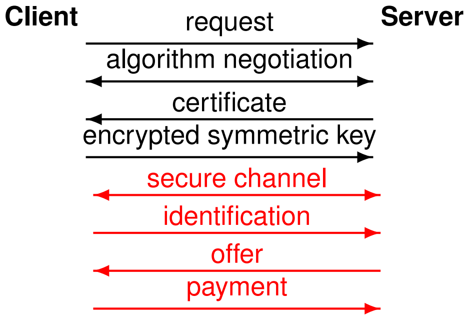
digital signature

- easy to copy
- message-dependent
- verified with a public key
- machine computable
- copies are digital evidence

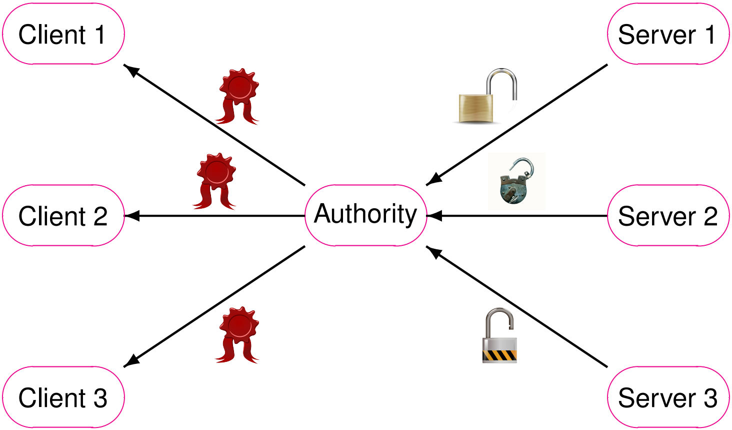
Public-Key Infrastructure (PKI)



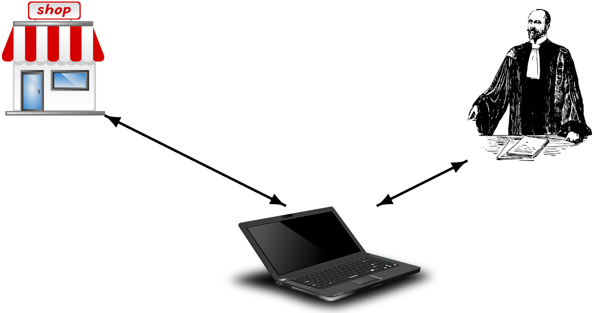
Transaction with “https”



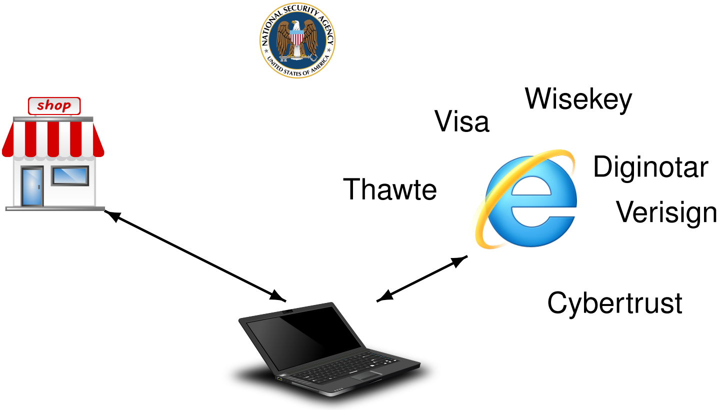
Critical Channels



Idealized Security

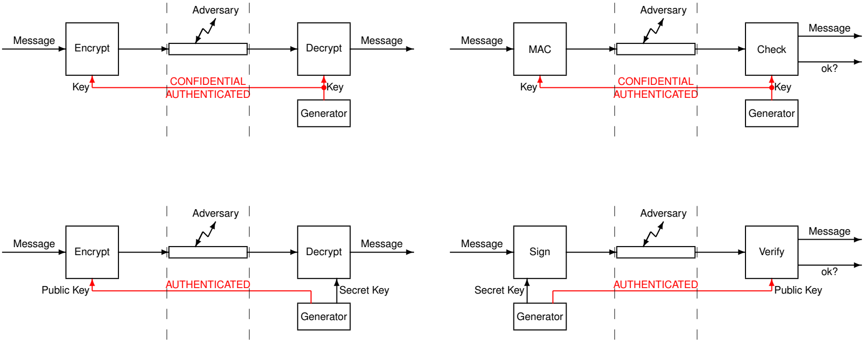


Security in Practice: Spot the Error



4 Main Cryptographic Primitives

confidential transmission **authenticated transmission**

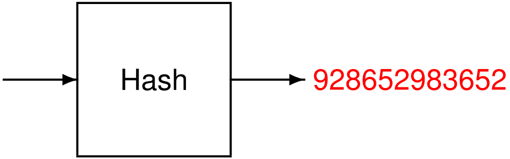


Cryptographic Primitives

- symmetric encryption
- message authentication code
- hash function
- key agreement protocol
- public-key cryptosystem
- digital signature

A 6th Important Cryptographic Primitive

La cigale ayant chanté tout l'été se trouva fort dépourvue quand la bise fut venue pas un seul petit morçeau de mouche ou de vermisseau elle alla trouver famine chez la fourmie sa voisine ...

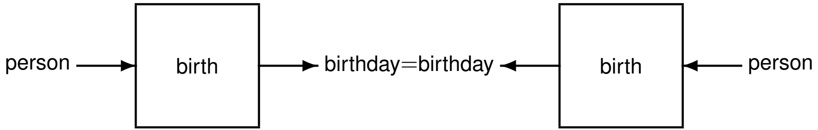


- can hash a string of arbitrary length
- produce digests (hashes) of standard length (e.g. 224 bits)
- sometimes called "fingerprint"
- use: sign a hash instead of a randomly formatted message

Meaning of Breaking

- for **encryption**
  show that we can recover the decryption key
  (more generally): show that we can decrypt a target ciphertext when we have access to a decryption oracle, but without submitting the target to the oracle
- for **signature**
  show that we can recover the signing key
  (more generally): show that we can forge the signature for a target message when we have access to a signing oracle, but without submitting the target to the oracle
- for **hashing**
  show that we can produce two documents with the same hash (same fingerprint)

Collision Search



Birthday Paradox

Theorem

If we pick independent random numbers in $\{1, 2, \ldots, N\}$ *with uniform distribution, n times, we get at least one number twice with probability* $p \approx 1 - e^{-\frac{n^2}{2N}}$ *for* $n \ll N$.

For $N = 365$:

n	10	15	20	25	30	35	40
probability	12%	25%	41%	57%	71%	81%	89%

Common Algorithms

be careful with the key lengths

- symmetric encryption: AES
- hash function: SHA3
- MAC/PRF: HMAC-SHA2
- authentication encryption: AES-CCM, AES-GCM

  be careful with the nonce
- key agreement: DH, ECDH
- cryptosystem: RSA, elliptic-curve cryptography
- signature: RSA, DSA, ECDSA

  be careful with the randomness

Key Length

- symmetric encryption/MAC: bit-security
- RSA: check tables
- hash with collision resistance: digest of **twice** bit-security
- hash without collision resistance: digest of bit-security
- discrete logarithm/DH in a group: **twice** bit-security
  caveat: if subgroup of $\mathbf{Z}_p^*$, p must be of size like for RSA

method	year	sym.	RSA	DL		EC	hash
Lenstra-Verheul	2015	82	1613	145	1613	154	163
Lenstra updated	2015	78	1245	156	1245	156	156
ECRYPT II	2011–15	80	1248	160	1248	160	160
NIST	2011–30	112	2048	224	2048	224	224
FNISA	2010–20	100	2048	200	2048	200	200
BSI	2011–15	–	1976	224	2048	224	224

(`http://www.keylength.com` by Quisquater)

Secure Communication

all together now!

- be careful!
  security does not add up...
  too many ways to make mistakes
- all-in-one primitives exist (**authenticated encryption**)
- various **setup assumptions**
  - secure hardware (badge, SIM card, smart card, TPM)
  - trusted third party (key server, authority)
  - PKI (with one/several certificate authorities)
  - password/preshared secret
  - out-of-band channel (SAS)
- more modern security notions (e.g. for instant messaging)

Case Studies

if we had time...

- TLS
- wifi
- mobile telephony
- signal
- bluetooth
- blockchains
- NFC payment
- MRTD

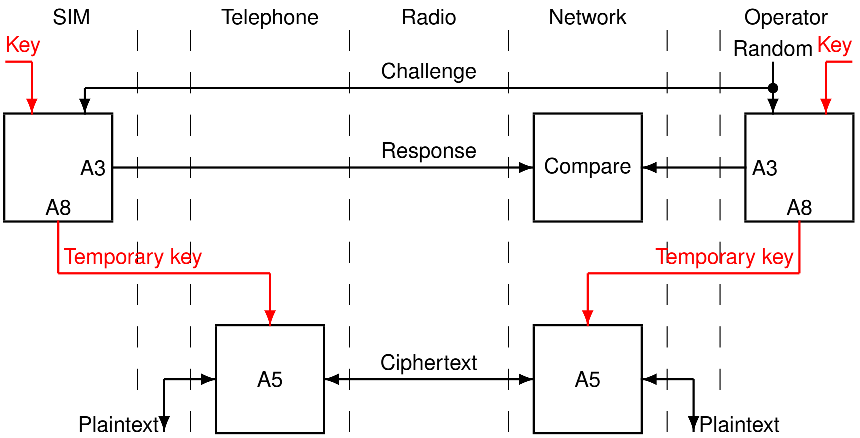
Mobile Telephony

- principle 1: authentication of mobile system
- principle 2: privacy protection in the wireless link

GSM architecture:

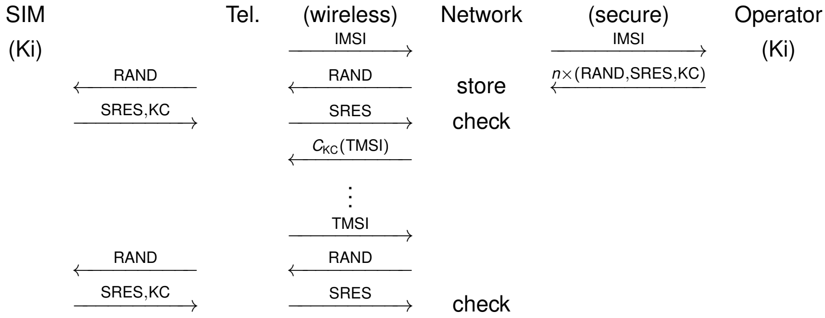
- challenge-response protocol based on Ki
- short-term encryption key (derived from Ki)
- identity IMSI replaced by a pseudonym TMSI as soon as possible
- Ki never leaves the security module (SIM card) or home security database (HLR)

GSM Protocol



GSM Authentication

$$\text{A3/8}(\text{Ki}, \text{RAND}) = (\text{SRES}, \text{KC})$$



GSM Encryption

- several standard algorithms: A5/0, **A5/1**, A5/2, A5/3
- cipher imposed by network
- new KC for each session
- synchronized frame counter (used as a nonce)

Security of Privacy protections

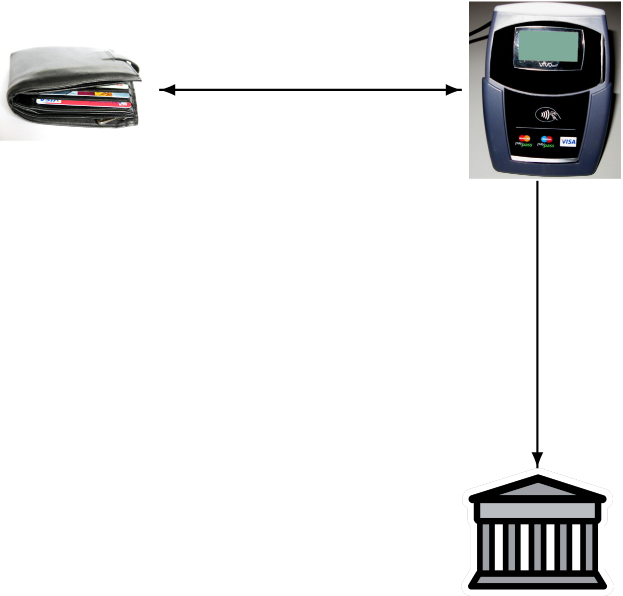
- blinding the identity is not effective at all:
  - challenges can be replayed to trace mobile telephones
  - fake network can force identification in clear (re-synchronization protocol)
- security of A5/0 (no encryption) void
- security of A5/2 weak
- security of A5/1 not high
- security of A5/3 high
- fake network can force to weak encryption (they all use the same key)
- replaying a challenge will force reusing a one-time key
- message integrity protection is ineffective

security: ☹

Improvements in 3G Mobile Telephony

- challenges are authenticated
  (fake network cannot forge them)
- integrity protection (MAC)
- protection against challenge-replay attacks
- uses block cipher KASUMI instead of stream cipher A5/1

NFC Payment



(Simplified) EMV PayPass Protocol

Card: PrivC, K_M

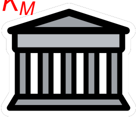
Card → Terminal: Cert(PubC, SSAD), PAN, CDOL — Terminal: verify

Terminal → Card: UN, amount, info — Terminal: pick UN; Card: inc. ATC

Card: compute — Card → Terminal: ATC, SDAD — Terminal: verify

Terminal → Bank: AC, amount, ATC, info

Bank: K_M

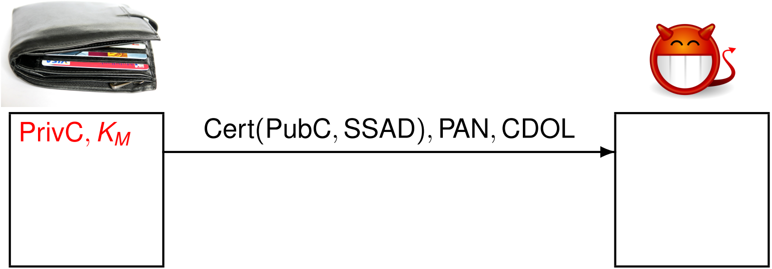


- PAN: serial number of the card
- SSAD: info about the card including PAN
- CDOL: description of what is needed in info
- ATC: number of the transaction
- $AC = MAC_{Enc_{K_M}(ATC)}(amount, ATC, info)$
- $SDAD = Sign_{PrivC}(AC, UN, amount, ATC, info)$

From Paper to Bits...

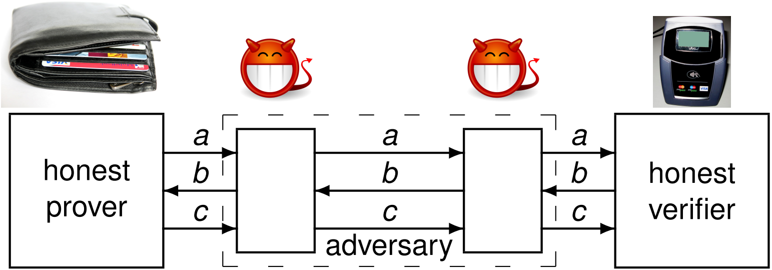
- holder is not aware a payment is happening
- holder is not aware of the payment amount
- no access control of the payment terminal (no PIN)
- payee is not authenticated (info could be anyone)
- privacy issue (SSAD leaks)

Skimming

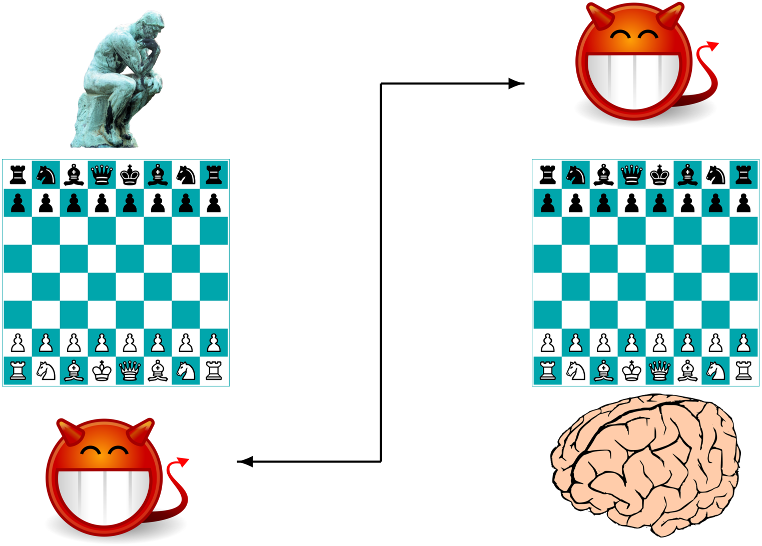


get name on card, credit card number, expiration date, etc

Relay Attacks



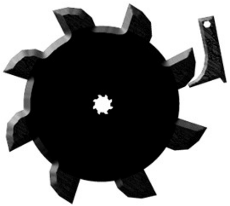
Playing against two Chess Grandmasters



Relay Attacks in Real

- opening cars and ignition (key with no button)
- RFID access to buildings or hotel room
- toll payment system
- NFC credit card (for payment with no PIN)
- access to public transport
- ...

Signal



Signal

used in WhatsApp

- **secure messaging** (confidentiality, authenticity, integrity of messages)
- **forward and future secrecy** (confidentiality preserved even though secrets leak)
- **deniability** (no transferable proof of message authorship leaks)
- **asynchronous** (can be done offline)
- detect replay/reorder/deletion attacks
- allow decryption of out-of-order messages
- don't leak metadata

Initial Key Agreement

Alice		**Server**		**Bob**
	Alice, G^a →	register	← Bob, G^b, $G^{x_{b,i}}$	$i = 1, \ldots, 100$
	Bob? →			
$x_{b,\text{eph}} \leftarrow G^{x_{b,i}}$	← G^b, $G^{x_{b,i}}$	erase $G^{x_{b,i}}$		

pick $x_{a,\text{eph}}$

state: $(G^b, G^{x_{b,\text{eph}}}, G^{x_{a,\text{eph}}})$

compute secret —— $G^{x_{a,\text{eph}}}, G^{x_{b,\text{eph}}}, \text{Enc}_{\text{secret}}(\text{msg})$ ——→

$$[\text{secret} = G^{a x_{b,\text{eph}}} \,||\, G^{b x_{a,\text{eph}}} \,||\, G^{x_{a,\text{eph}} x_{b,\text{eph}}}]$$

← Alice? (Bob to Server)

$(G \in \text{Curve25519})$

G^a → (Server to Bob) — compute secret

state: $(G^a, G^{x_{a,\text{eph}}}, G^{x_{b,\text{eph}}})$

decrypt

erase $x_{b,\text{eph}}$

pick $x_{b,\text{eph}}$

compute secret ←—— $G^{x_{b,\text{eph}}}, G^{x_{a,\text{eph}}}, \text{Enc}_{\text{secret}}(\text{msg})$ —— compute secret

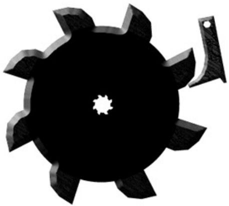
decrypt

erase $x_{a,\text{eph}}$

⋮

Ratchet

A ratchet is a mechanical device which can only move forward.



- **forward secrecy**: protects past sessions against future compromises of *long-term* secret keys
- **future secrecy**: protects future sessions against compromises of *ephemeral* secret keys

Double Ratchet in Signal

- 3DH: a ratchet for every time the direction of exchange changes
  - needs synchronization between the two participants
  - good forward and future secrecy
- a ratchet for a sequence of messages in the same direction
  - no real future secrecy
  - plausible deniability

MRTD

Schweizer Pass
Passeport suisse
Passaporto svizzero
Passaport svizzer
Swiss passport

ICAO-MRTD Objectives

(MRTD=Machine Readable Travel Document)

more secure identification of visitors at border control

- $\rightarrow$ biometrics
- $\rightarrow$ contactless IC chip
- $\rightarrow$ digital signature + PKI

maintained by UN/ICAO (International Civil Aviation Organization)

MRTD History

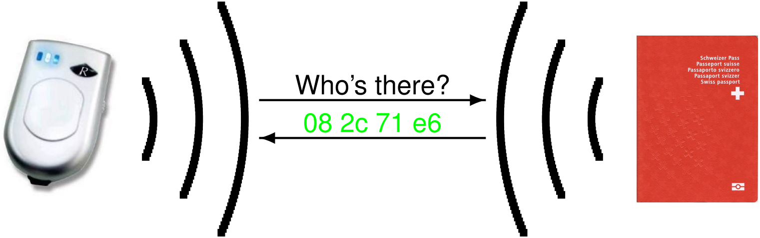
- 1968: ICAO starts working on MRTD
- 1980: first standard (**Machine Readable Zone (MRZ)**)
- 1997: ICAO-New Tech. WG starts working on biometrics
- 2001 9/11: US want to speed up the process
- 2002 resolution: ICAO adopts **facial recognition** (+ optional fingerprint and iris recognition)
- 2003 resolution: ICAO adopts **contactless IC media** (instead of e.g. 2D barcode)
- **2004: version 1.1** of standard with ICC
- 2005: deployment of epassports in several countries
- 2006: **extended access control** in the EU
- now part of Doc9303

MRZ Example

```
PMCHEDUPONT<<JEAN<<<<<<<<<<<<<<<<<<<<<<<<<<<
X337803X<6CHE7208066M1308147<<<<<<<<<<<<<<<3
```

- document type
- issuing country
- holder name
- doc. number + CRC
- nationality
- date of birth + CRC
- gender
- date of expiry + CRC
- options + CRC

ISO 14443 (RFID)



- frequency: 13.56MHz
- typical range: 2cm
- reported range (with legal equipment): 12m

RFID

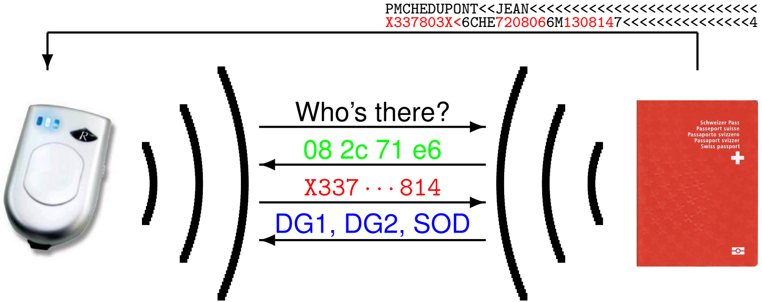
Advantages

- robust
- large storage capacity
- dynamic

Problems

- leaks information
- answers to anyone

ICAO (MRTD): BAC and Passive Authentication



- DG1: official name, citizenship, X337 · · · 814, gender
- DG2: facial picture
- SOD: signature by authorities of the hash of DG's

Identity Example

DG1 `PMCHEDUPONT<<JEAN<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<`
`X337803X<6CHE7208066M1308147<<<<<<<<<<<<<<<4`

DG2

SOD

Hashes:

- DG1: 4e1249fb72c8e70ba72f488dc1f91394e57f9f83
- DG2: a3853c3c7261c2788fc2c4b9db372c5875f5c91d

Signature:

```
54a4 a626 4ee1 c0ab e022 3f1d e673 75d4
7c89 7e7f d8fb acd6 abbf d568 b178 7171
652d e730 43c2 9495 6134 680c 7070 9028
1caa 2364 17e8 ffa0 9ee7 c8be 4c32 908c
```

Certificate:

MIIECTCCA5GgAwIBAgIBFDAJBgcqhkjOPQQBMHExCzAJBgNVBAYTAkNIMQ4wDAYD
VQQKEwVBZG1pbjERMA8GA1UECxMIU2VydmljZXMxIjAgBgNVBAsTGUNlcnRpZmlj
YXRpb24gQXV0aG9yaXRpZXMxGzAZBgNVBAMTEmNzY2Etc3dpdHplcmxhbmQtMTAe
Fw0wODA1MTkwODA4NDVaFw0xNDA2MjEwODA4NDVaMG0xCzAJBgNVBAYTAkNIMQ4w
DAYDVQQKEwVBZG1pbjERMA8GA1UECxMIU2VydmljZXMxGTAXBgNVBAsTEFNpZ25h
dHVyZS1TZXJ2ZXIxDzANBgNVBAsTBlBhc3MwNjEPMA0GA1UEAxMGZHMtMDAxMIIB
MzCB7AYHKoZIzj0CATCB4AIBATAsBgcqhkjOPQEBAiEA/////wAAAAEAAAAAAAAA
AAAAAAD///////////////8wRAQg/////wAAAAEAAAAAAAAAAAAAAAAD/////////
//////wEIFrGNdiqOpPns+u9VXaYhrxlHQawzFOw9jvOPD4n0mBLBEEEaxfR8uEs
Qkf4vOblY6RA8ncDfYEt6zOg9KE5RdiYwpZP40Li/hp/m47n60p8D54WK84zV2sx
Xs7LtkBoN79R9QIhAP////8AAAAA//////////+85vqtpxeehPO5ysL8YyVRAgEB
A0IABO8J8UthgahfN1JQKIq9a1ll/L3er54mUd1SZMnKQ2pQTbX5JwHc9ByEgw3G
5kucfGw1k2uAts+Ck+WSovy7k7GjggFBMIIBPTArBgNVHRAEJDAigA8yMDA4MDUx
OTA4MDg0NVqBDzIwMDgwODIwMDgwODQ1WjBgBgNVHSAEWTBXMFUGCGCFdAERAzQB
MEkwRwYIKwYBBQUHAgEWO2h0dHA6Ly93d3cucGtpLmFkbWluLmNoL3BvbGljeS9D
UFNfMl8xNl83NTZfMV8xN18zXzUyXzEucGRmMIGbBgNVHSMEgZMwgZCAFE7InZjJ
tOCQ9StbhZdQVr/oJOt2oXWkczBxMQswCQYDVQQGEwJDSDEOMAwGA1UEChMFQWRt
aW4xETAPBgNVBAsTCFNlcnZpY2VzMSIwIAYDVQQLExlDZXJ0aWZpY2F0aW9uIEF1
dGhvcml0aWVzMRswGQYDVQQDExJjc2NhLXN3aXR6ZXJsYW5kLTGCAQEwDgYDVR0P
AQH/BAQDAgeAMAkGByqGSM49BAEDZwAwZAIwGYMbTqjlYQnJ1DSpb//5WtQthjoy
pGrbBZW1Rqa7TXffzQX818OjQCdQ0n9tZEDlAjBPtMdS9OymxywZpXZj9Os2qO6M
6htXJKXpdKSWq75ZhQRet/or3pT2MQ56n69hqGw=

MRTD

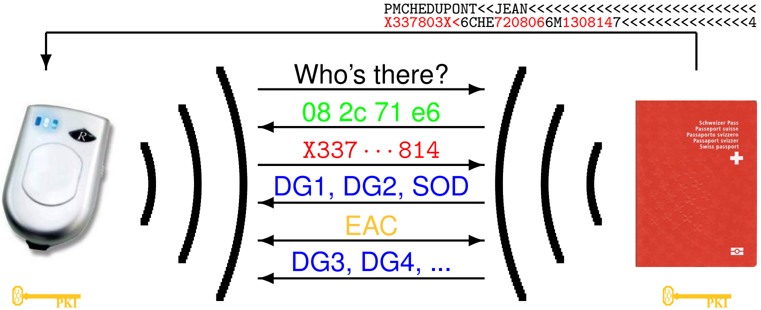
Advantages

- impossible to forge an identity
- protect against non-organized illegal immigration

Problems

- encourage identity theaft
- facial recognition is weakly reliable
- passeport cloning
- tracking people
- leakage of evidence
  - proof of official name
  - proof of wedding
  - proof of age
  - proof of gender
- anonymity loss

EAC: Access Control and Active Authentication



- EAC: chip authentication
- EAC: terminal authentication
- DG3...: fingerprint, other data

EAC

Advantages

- anti-cloning
- better access control
- better identification

Problems

- only where EAC is available
- still evidence leakages
- a new PKI

LDS Example

```
PMCHEDUPONT<<JEAN<<<<<<<<<<<<<<<<<<<<<<<<<<<
X337803X<6CHE7208066M1308147<<<<
```

DG1: same as MRZ

DG2: encoded face

DG3: encoded finger

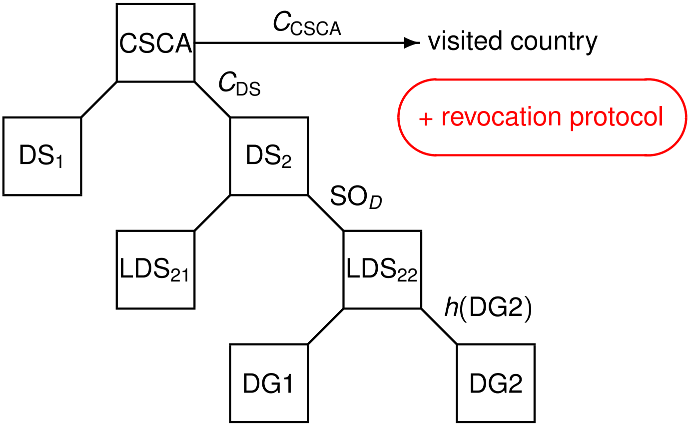
SO_D

[h(DG1), h(DG2), h(DG3)]
signature
certificate C_{DS}

LDS (MRTD Memory) Structure

- K_{ENC}, K_{MAC}, KPr_{AA}
- COM: present data groups
- DG1: same as MRZ
- DG2: encoded face
- DG3: encoded finger(s)
- DG4: encoded eye(s)
- DG5: displayed portrait
- DG6: (reserved)
- DG7: displayed signature
- DG8: data feature(s)
- DG9: structure feature(s)
- DG10: substance feature(s)
- DG11: add. personal detail(s)
- DG12: add. document detail(s)
- DG13: optional detail(s)
- DG14: security options
- DG15: KPu_{AA}
- *DG16: person(s) to notify*
- SO_D

(Country-wise) PKI



- one CSCA (*Country Signing Certificate Authority*)
- several DS (*Document Signer*) per country
- SO_D: signature of LDS
- fingerprint of a DG

Passport: From Paper to Bits

paper passport

- invisible if not shown
- hard to copy
- photocopies are non-binding
- needs human check
- access control by the holder

MRTD

- detectable, recognizable
- easy to copy with no AA
- SOD is a digital evidence
- readable automatically
- needs specific access control

MRZ_info

```
PMFRADUPONT<<<<JEAN<<<<<<<<<<<<<<<<<<<<<<<<<
74HK8215<6CHE7304017M0705121<<<<<<<<<<<<<<03
```

- document type
- issuing country
- holder name
- doc. number + CRC
- nationality
- date of birth + CRC
- gender
- date of expiry + CRC
- options + CRC

Basic Access Control

Authenticated Key Exchange Based on MRZ_info

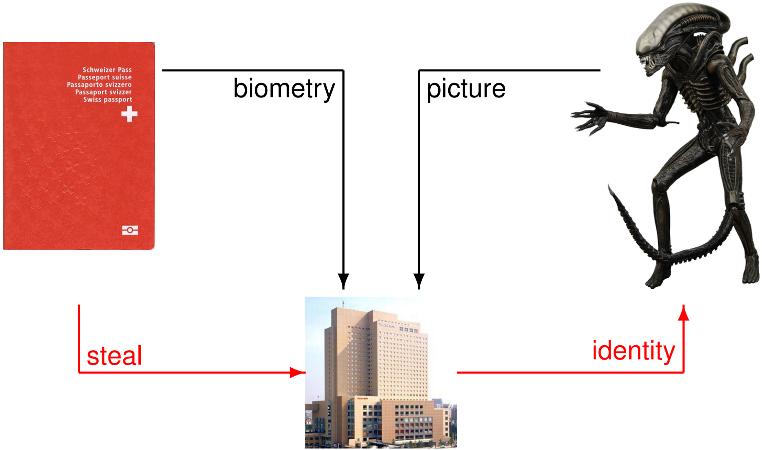
IFD		ICC
	(derive K_{ENC} and K_{MAC} from MRZ_info)	
	GET CHALLENGE $\longrightarrow$	
pick RND.IFD, K.IFD	$\longleftarrow$ RND.ICC	pick RND.ICC, K.ICC
$S \leftarrow \text{RND.IFD}\Vert\text{RND.ICC}\Vert\text{K.IFD}$	$[S]_{K_{\text{ENC}},K_{\text{MAC}}}$ $\longrightarrow$	check RND.ICC
check RND.IFD	$\longleftarrow$ $[R]_{K_{\text{ENC}},K_{\text{MAC}}}$	$R \leftarrow \text{RND.ICC}\Vert\text{RND.IFD}\Vert\text{K.ICC}$

(derive KS_{ENC} and KS_{MAC} from $K_{\text{seed}} = \text{K.ICC} \oplus \text{K.IFD}$)

Security and Privacy Issues

- collision avoidance discrepancies
  $\rightarrow$ deviating from standard induce leakages
- MRZ_info entropy
  $\rightarrow$ online attack or offline decryption from skimming
- underestimated wireless range limits
  $\rightarrow$ claimed to be possible at a distance of 25m
- identity theft (by stealing/cloning MRTD)
  $\rightarrow$ facial recognition is weak
- remote passport detection
  $\rightarrow$ nice to find passports to steal
- relay attacks
- denial of services
- ...

Identity Theft



a few 100 customers are enough

Extended Access Control (EAC)

- **PACE** $>$ BAC
- **Chip Authentication**
- **Terminal Authentication** to access non-mandatory data
- more biometrics (finger) for more secure identification

- using state-of-the-art cryptography
(public-key crypto, PAKE, elliptic curves)
- secure access control but requires a heavy PKI for readers

- in-process standard: protocols with different versions, variants, described in different documents, with different notations...

Terminal Authentication Issues

Terminal revocation issue:

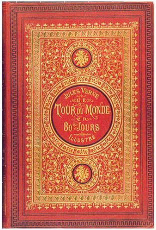
- MRTDs are not online!
- MRTDs have no reliable clock

Information Leakage

- SO_D leaks the digest of protected DGs before passing EAC
- could be used to recover missing parts from exhaustively search
- could be used to get a proof if DG is known

Conclusion on MRTD

- **LDS**: contains too much private information
- **passive authentication**: leaks evidence for LDS
- **BAC**: does a poor job
- **secure messaging**: OK
- **AA**: leaks digital evidences, subject to MITM
- **EAC**: much better, but still leaks + revocation issue
- **RFID**: leaks
- **biometrics**: leaks template



"Les passeports ne servent jamais qu'à gêner les honnêtes gens et à favoriser la fuite des coquins."

Jules Verne, 1872
Le tour du monde en 80 jours

Other Useful Primitives

- zero-knowledge (for privacy)
- property-preserving encryption (for databases): searchable, order-revealing, format-preserving
- homomorphic things (for privacy)
- multiparty computation
- identity-based cryptography
- obfuscation

Intrinsic Threats to Cryptography

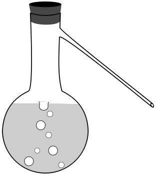
- Moore law
  natural increase of the computational power of computers
  $\rightarrow$ security gracefully decreases
- Shor algorithm
  quantum computer
  $\rightarrow$ security will collapse in an earthquake
- non-guaranteed hypotheses
  finally, factoring could be easy
  $\rightarrow$ security fall can occur at any time

Lack of Cryptodiversity

Erroneous Security Proofs

[illustration by Fred]

Bad Random Sources

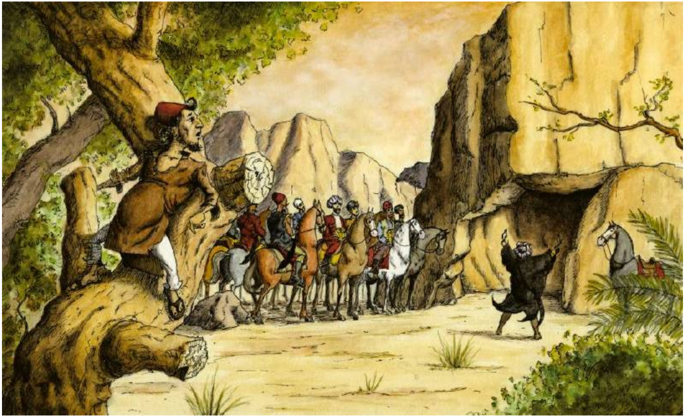


- the secret key of servers can be guessed
- the secret key of routers can be guessed
- play stations can be cracked
- bitcoins can be stolen
- bad algorithms may be (maliciously) imposed as standards

Lack of Composability Results

$$\text{secure} + \text{secure} \stackrel{?}{=} \text{secure}$$

Leakage Due to Hardware



power consumption	response to stress	time of computation
radio emanation	vibration	branch prediction
cache fails	format validation	...

Trust in Security Infrastrutures

[Disney illustration]

Hot Issues

- good authenticated encryption
- side channel mitigation
- leakage resiliency
- randomness generation
- postquantum cryptography

Principles of Quantum Mechanics

- randomness is inherent
- **state** of an isolated system: unit vector with complex coordinates
- transformations of an isolated system are reversible
- **observing** a system degrades the state of the system
- weird things:
  - **superposition state**
  - **entangled state**
  - **no-cloning principle**

Superposition State

- elementary states defined with the $|\cdot\rangle$ notation
- example: the aliveness of Shrödinger's cat

$$\begin{aligned} X &= |\text{alive}\rangle \\ Y &= |\text{dead}\rangle \end{aligned}$$

  a state could be a combination $\alpha X + \beta Y$
- observing would end up in state $\begin{cases} X \text{ with probability } |\alpha|^2 \\ Y \text{ with probability } |\beta|^2 \end{cases}$

Entangled State

- with two particules A and B which can be either "up" (1) or "down" (0), we have the orthogonal states

$$|00\rangle \quad |01\rangle \quad |10\rangle \quad |11\rangle$$

- we could have the state

$$\frac{1}{\sqrt{2}}\left(|00\rangle + |11\rangle\right)$$

- observing one particule affects the other!

Quantum Computing

- qbits of memory are complex vectors of dimension two: $\alpha|0\rangle + \beta|1\rangle$ with $|\alpha|^2 + |\beta|^2 = 1$
- a quantum memory of n qbits is a combination $\sum_{b_1 \cdots b_n} \alpha_{b_1 \cdots b_n} |b_1 \cdots b_n\rangle$
- operations are unitary linear operations
- if we have a classical circuit to compute f, we have an equivalent quantum circuit to transform $|x\ y\rangle$ into $|x\ y + f(x)\rangle$
- we can create $\frac{1}{\sqrt{2^{\text{length}(x)}}} \sum_x |x\ f(x)\rangle$ (free parallelism)
- observing one qbit is a linear projection

Main Algorithms

Shor algorithm

- factoring in quasi-linear time (instead of sub-exponential)
  † RSA
- discrete logarithm in quasi-linear time
  † Diffie-Hellman, DSA, elliptic curves

→ need alternate cryptography: postquantum cryptography

Grover algorithm

- finds a needle in a haystack in square root time

→ need to double key lengths in symmetric cryptography

Quantum Cryptography

by transmitting photon in a quantum state, we can make a key agreement

- unconditionally secure
  (if the adversary can only see or modify the transmitted photons and any other classical communication)
- not secure against side-channel attacks
- the key can only be used with the Vernam cipher
- needs a quantum channel

Bennett–Brassard Protocol

Alice		**Bob**
pick $x, e \in \{0,1\}$		pick $d \in \{0,1\}$
$\vert A\rangle = H^e \vert x\rangle$	$\xrightarrow{\vert A\rangle}$	$y = \text{measure}_{d=0?Z:X}(\vert A\rangle)$
erase x if $e \neq d$	$\xleftarrow{d}$	
	$\xrightarrow{\text{kept or erased}}$	erase y is Alice erased
	$(x = y)$	

$$\Pr[\text{measure}_{d=0?Z:X}(H^e|x\rangle) = x] = \begin{cases} 0 \text{ if } d = e \\ \frac{1}{2} \text{ if } d \neq e \end{cases}$$

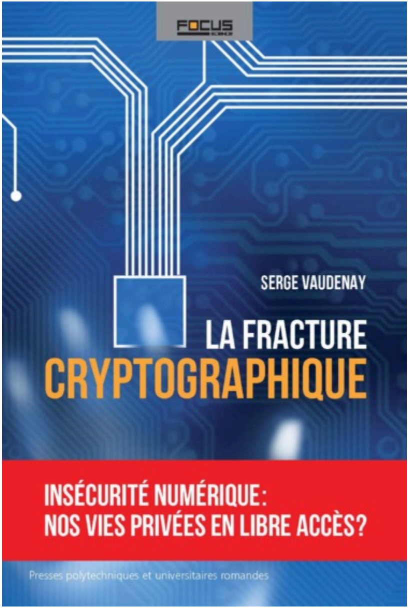
this must be followed by a verification of having received enough bits and of correctness

Conclusion

- many techniques and algorithms to protect information
- almost always relying on some hardness assumption
- does not compose
- delicate to use
- quite multidisciplinary and lively!
- a science for

malicious behaviors and protection techniques

Pub


FOCUS
SERGE VAUDENAY
LA FRACTURE
CRYPTOGRAPHIQUE
INSÉCURITÉ NUMÉRIQUE:
NOS VIES PRIVÉES EN LIBRE ACCÈS?
Presses polytechniques et universitaires romandes