Lecture Notes

electronic passports & biometrics

Michael Nüsken

b-it

(Bonn-Aachen International Center
for Information Technology)

summer 2009


trust
ICAO
Passport
authentication
identification
special paper with specific fine pattern
unique id-number
biometry
hologram
issuing authority (state)
total number of pages
trusted parties
seal
signature
· remarks
· accompanying persons...
date of validity & (expiration) of issuance
immigration rules
emigration rules
visas
fingerprint
person
photograph
eyes:
· retina scan
· iris scan
electronic
cryptographic
nationality
country
eye color
- hair color
voice
chip → RFID
→ contact chip
name
- weight
height
MRZ machine readable zone
date of birth
gender
- magnetic stripe
place of birth
- barcode
- current address
signature
electronic signature
- father's name
- mother's name
encryption

Table IIIA-1. Summary of security recommendations

Threats	Basic features	Additional features
Counterfeiting		
Paper substrates (5.1.1)	– controlled UV response – two-tone watermark – chemical sensitizers – appropriate absorbency and surface characteristics	– registered watermark – invisible UV fibres/ planchettes – visible UV fibres/planchettes – embedded or window thread
Label substrates (5.1.2)	– controlled UV response – chemical sensitizers – invisible UV fibres/planchettes – visible UV fibres/planchettes – non-peelable adhesive	– embedded or window thread
Plastic/synthetic substrates (5.1.4)	– as per paper or substitute – security features providing an equivalent level of security in plastic	– optically variable feature(OVF)
Security printing (5.2)	– two-colour guilloche background – rainbow printing – anti-scan pattern – microprinting – unique biodata page design	– intaglio printing – latent image – duplex pattern – 3-D design feature – front-to-back register feature – deliberate error in microprint – unique design on every page – tactile feature
Numbering (5.2.3)	– unique document number	– perforated document number – special typefonts
Inks (5.2.2):	– UV inks on all pages – reactive inks	– optically variable properties – metallic inks – penetrating numbering ink – metameric inks – infrared dropout ink – thermochromic ink – photochromic ink – infrared fluorescent ink – phosphorescent ink – tagged ink

Table IIIA-1. Summary of security recommendations

Threats	Basic features	Additional features
Photo-substitution (5.4.4)	– integrated biodata page – guilloche overlapping portrait – secure laminate or equivalent	– OVF over the portrait – digital signature in document – embedded image – secondary portrait image – storage and retrieval system for digital portrait images – biometric feature
Alteration of the biodata (5.4.4)	– reactive inks – secure laminate or equivalent	– chemical sensitizers in substrate – secondary biodata image – OVF over the biodata
Page substitution (5.5.3/4)	– lock stitch or equivalent – unique biodata page design	– programmable sewing pattern – fluorescent sewing thread – serial number on every page – page folio numbers in guilloche – index marks on every page – biodata on inside page
Deletion/removal of stamps and labels (5.5.5)	– reactive inks – chemical sensitizers – high-tack adhesives (labels) – permanent inks (stamps)	– over-lamination – high absorbency substrates – frangible substrate (labels)
Document theft (5.7.1):	– good physical security arrangements – control of all security components – serial numbers on blank documents – secure transport of blank documents – internal fraud protection system – international exchange on lost and stolen documents	– CCTV in production areas – centralized production – digital signature – embedded image

Cryptographic passports

Security?

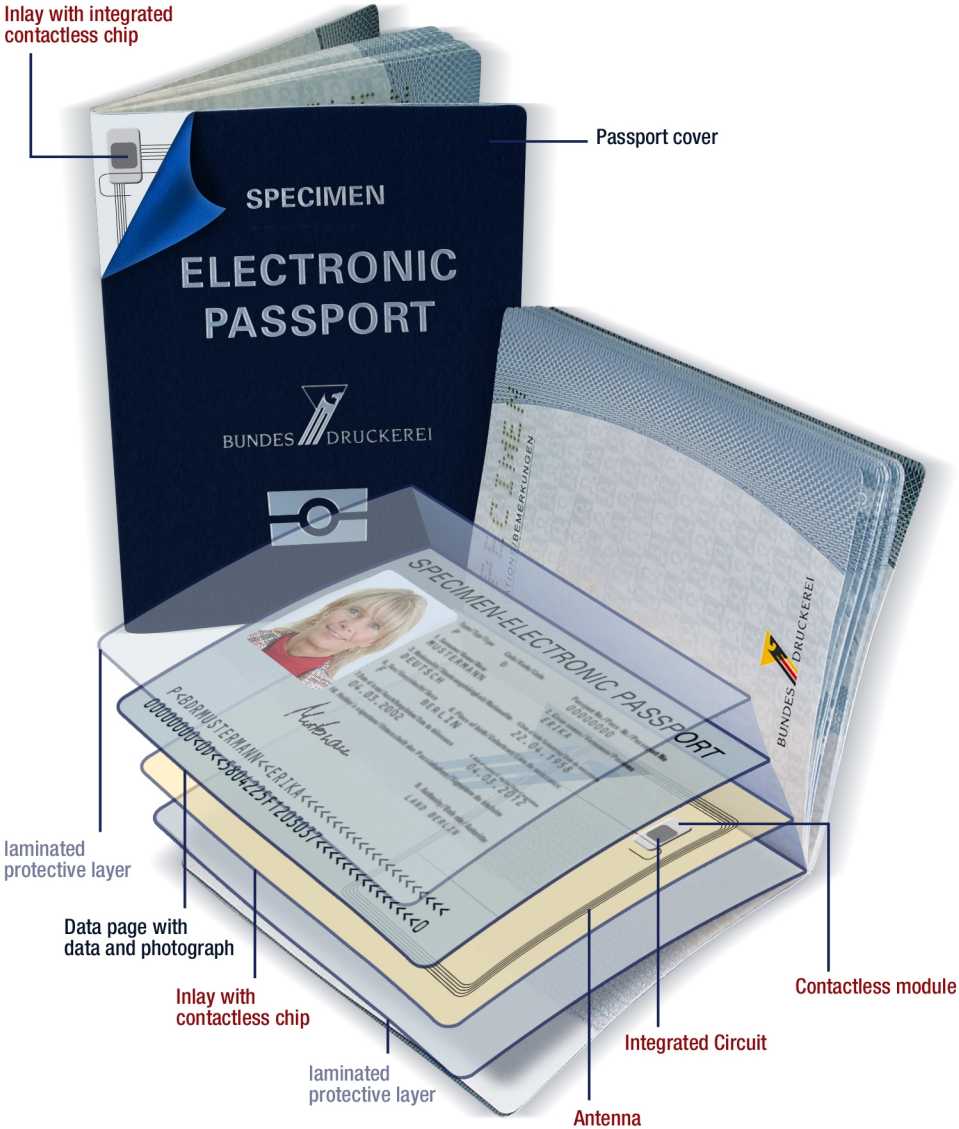
- difficult to forge
- impossible to forge — Unrealistic
- difficult to ~~access~~ read for unauthorized parties
- difficult to track
- database?
- discrimination?

Reliability?

- easy to use

biometric
false accept rate
false reject rate

→ Acceptance?
→ Accessibility? Costs?
→ Robust?



Source-Bundesdruckerei GmbH

▶ The contactless chip can be integrated into either the cover page or the data page.

cpb
21.4.09
(2)

New threat due to contactless chip:

invisible reading process,

radio waves are not visible!

→ USE CRYPTOGRAPHY!

Starting ideas:

→ use MRZ for basic access control
... How? Details ...

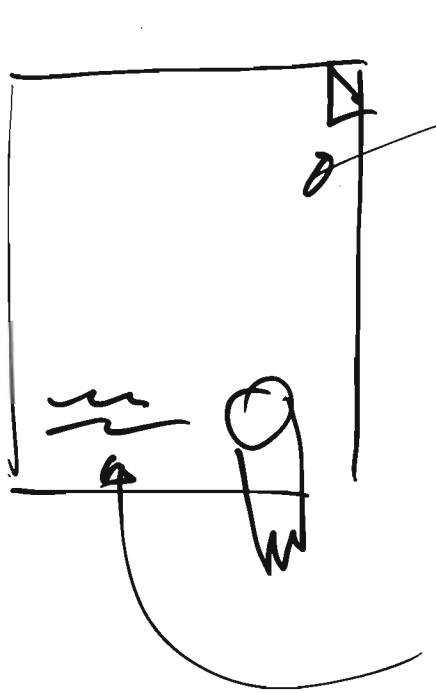
→ use additional barriers for access to sensitive data, like e.g. finger prints.

Signatures til ECDSA

cpb
22.4.09
(1)

What's a signature?

- identification of a person
  (unique, every person has a different one)
  use biometrics...
- authenticates a document
  (commitment, sign a document)
  - Link between person and document
    which cannot be seperated.
- integrity of the document



unchanged integrity

link — authenticity

signature — identity

ElGamal type signatures, first try

$$a^b \; b^\gamma = g^m$$

verification equation

What is necessary to give this some sense?

- → need a multiplication for the parts/elements a^b, b^γ,
- → need an exponentiation to compute a^b, b^γ, g^m.

Constraint: everything must be finite!

Use a finite group!

Excursion: finite groups.

Computer scientist's definition:

(P)roper: there is an implementation which allows a finite set G of values and has an operation $\cdot : G \times G \longrightarrow G$ and (optionally) an operation $inv = ?^{-1} : G \longrightarrow G$

and (optionally) an element

$$1 \in G .$$

(A)ssociative: for any $a, b, c \in G$:

$$(a \cdot b) \cdot c = a \cdot (b \cdot c)$$

(N)eutral element: for any $a \in G$ we have

$$1 \cdot a = a = a \cdot 1$$

(I)nverses: for any $a \in G$ there exists $b \in G$

$$a \cdot b = 1 = b \cdot a$$

or equiv: for any $a \in G$ we have

$$a \cdot a^{-1} = 1 = a^{-1} \cdot a$$

optional:

(C)ommutative: for any $a, b \in G$ we have

$$a \cdot b = b \cdot a$$

Examples:

$(\mathbb{Z}_N, +)$: elements are coded as integer $0, 1, 2, \dots, N-1$ and the operation is defined by:

$$a + b := (\check{a} +_{\mathbb{Z}} \check{b}) \bmod N \in \mathbb{Z}_N$$

[in contrast to: $(\check{a} +_{\mathbb{Z}} \check{b}) \bmod N \in \mathbb{Z}$.]

in $\mathbb{Z}_{15}$: $9 + 7 = 1$,

cp b
22.4.09
(4)

$(\mathbb{Z}_N, \cdot)$, $N = 6$. Not a group:

no inverse for 2:

$2 \cdot 0 = 0$, $2 \cdot 3 = 0$,
$2 \cdot 1 = 2$, $2 \cdot 4 = 2$
$2 \cdot 2 = 4$, $2 \cdot 5 = 4$.

$(\mathbb{Z}_N, \cdot)$, $N = 7$. Not a group!

Beh.: $2 \cdot 4 = 1$.
$3 \cdot 5 = 1$,
$1 \cdot 1 = 1$,
$6 \cdot 6 = 1$.

$0 \cdot a = 0$ never 1 → no inverse.

$(\mathbb{Z}_N \setminus \{0\}, \cdot)$, $N = 7$.

$\cdot$	1	2	3	4	5	6
1	(1)	2	3	4	5	6
2	2	4	6	(1)	3	5
3	3	6	2	5	(1)	4
4	4	(1)	5	2	6	3
5	5	3	(1)	6	4	2
6	6	5	4	3	2	(1)

Proper: ✓ (Check the table!)

Associative: We know that multiplication for integers is associative:

$$(a \cdot b) \cdot c = (\check{a} \cdot_{\mathbb{Z}} \check{b} \underbrace{\text{ rem } N}) \cdot_{\mathbb{Z}} \check{c} \quad \text{mod } N$$

$$= (\check{a} \cdot_{\mathbb{Z}} \check{b} \underbrace{- k \cdot_{\mathbb{Z}} N}) \cdot_{\mathbb{Z}} \check{c} \quad \text{mod } N$$

$(\mathbb{Z}_6 \setminus \{0\}, \cdot)$ not a group!

Still: 2 has no inverse.

Also: $2 \cdot 3 \notin \mathbb{Z}_6 \setminus \{0\}$.

$(\mathbb{Z}_N^\times, \cdot)$

↑ set of invertible elements in $\mathbb{Z}_N$.

P? A ✓ N: 1 is invertible I:? ✓

→ $1 \cdot 1 = 1$ (invertible)

$1 \cdot a = a$ (neutral)

I: We have to check that if a is invertible, then its inverse b is also invertible.

But if: $a \cdot b = 1 = b \cdot a$

Then: $b \cdot a = 1 = a \cdot b$

so a is the inverse of b

and thus b is invertible.

P: Given a, b invertible.

Check that $a \cdot b$ is invertible.

But: $(a \cdot b) \cdot (b^{-1} \cdot a^{-1}) = 1$

$(b^{-1} \cdot a^{-1}) \cdot (a \cdot b) = 1$ ✓

This is a group! We call it:

the unit group of integers modulo N.

cpb 22.4.09 (5)

$$(a \cdot b) \cdot c = (\check{a} \cdot_{\mathbb{Z}} \check{b}) \cdot_{\mathbb{Z}} \check{c} - \underbrace{(k \cdot_{\mathbb{Z}} N) \cdot_{\mathbb{Z}} \check{c}}_{\text{is a multiple of } N} \bmod N$$

$$= (\check{a} \cdot_{\mathbb{Z}} \check{b}) \cdot_{\mathbb{Z}} \check{c} \bmod N$$

$$= \check{a} \cdot_{\mathbb{Z}} (\check{b} \cdot_{\mathbb{Z}} \check{c}) \bmod N$$

$$= a \cdot (b \cdot c)$$

Neutral: $1 \in \mathbb{Z}_N \setminus \{0\}$.

$$1 \cdot a = (\check{1} \cdot_{\mathbb{Z}} \check{a}) \bmod N$$

$$= \check{a} \bmod N = a.$$

Inverses: Every row has a 1. ✓

Commutative? Table symmetric wrt to the diagonal.

Here: $$a \cdot b = (\check{a} \cdot_{\mathbb{Z}} \check{b}) \bmod N$$

$$= (\check{b} \cdot_{\mathbb{Z}} \check{a}) \bmod N$$

$$= b \cdot a.$$

Summary: $(\mathbb{Z}_7 \setminus \{0\}, \cdot)$ is a comm. group. !

Example $\mathbb{Z}_6^\times = (\{1,5\}, \cdot)$

$\cong (\mathbb{Z}_2, +)$

$\cdot$	1	5
1	1	5
5	5	1

$+$	0	1
0	0	1
1	1	0

$(\mathbb{Z}_2, +) \longrightarrow (\mathbb{Z}_6^\times, \cdot)$

$0 \longmapsto 1$

$1 \longmapsto 5$

$a \longmapsto 5^{\check{a}}$

$\check{0} = 0 \in \mathbb{Z}$

$\check{1} = 1 \in \mathbb{Z}$

fast?!

$\longleftarrow$?? we do not know a fast algorithm!

Example

$\mathbb{Z}_5^\times = (\{1,2,3,4\}, \cdot) \cong (\mathbb{Z}_4, +)$

$\mathbb{Z}_4^\times = (\{1,3\}, \cdot) \cong (\mathbb{Z}_2, +)$

$\mathbb{Z}_7^\times = (\{1,2,3,4,5,6\}, \cdot) \cong (\mathbb{Z}_6, +)$

$\mathbb{Z}_8^\times = (\{1,3,5,7\}, \cdot) \cong (\mathbb{Z}_2 \times \mathbb{Z}_2, +)$

$\mathbb{Z}_9^\times = (\{1,2,4,5,7,8\}, \cdot) \cong (\mathbb{Z}_6, +)$

1/2

$(\mathbb{Z}_2 \times \mathbb{Z}_3, +)$

How to compute and/or decide inverses?

Task: Given $a \in \mathbb{Z}_N$.
Find $b \in \mathbb{Z}_N$ such that

(*) $a \cdot b = 1$ (in $\mathbb{Z}_N$),

or prove there is no such b.

Notice that (*) is equivalent to

$$\exists k \in \mathbb{Z}: \check{a} \cdot_{\mathbb{Z}} \check{b} + k \cdot_{\mathbb{Z}} N = 1$$

So we look for $\check{b}, k \in \mathbb{Z}$: $\check{b}\,\check{a} + k \cdot N = 1$ in $\mathbb{Z}$

Rephrased: find b, k such that

$$b\,\check{a} + k\,N$$

is (positive and) as small as possible!

For example:

$$1 \cdot \check{a} + 0 \cdot N = \check{a}$$
$$0 \cdot \check{a} + 1 \cdot N = N$$

Example: $N = 9$, $\check{a} = 2$. Starting:

(1)	$1 \cdot 2 + 0 \cdot 9$	$= 2$
(2)	$0 \cdot 2 + 1 \cdot 9$	$= 9$
(2)-4(1)	$(-1) \cdot 2 + 1 \cdot 9$	$= 7$
(2)-4(1)	$(-4) \cdot 2 + 1 \cdot 9$	$= 1$

Extended Euclidean Algorithm

pb
22.4.09
(3)

$15 \in \mathbb{Z}_{87}$. Inverse?

	r	q	s	t	$s \cdot 87 + t \cdot 15 = r$
	87	.	1	0	$1 \cdot 87 + 0 \cdot 15 = 87$
→	15	5	0	1	$0 \cdot 87 + 1 \cdot 15 = 15$
(87) − 5·(15):	12	1	1	−5	$1 \cdot 87 + (-5) \cdot 15 = 12$
(15) − 1·(12):	3	4	−1	6	$(-1) \cdot 87 + 6 \cdot 15 = 3$
	0		5	−29	$5 \cdot 87 + (-29) \cdot 15 = 0$.

$14 \in \mathbb{Z}_{92}$ Inverse?

Eurocrypt 2009

Conference Program 26. - 30. April 2009

April 26, 2009 Sunday	
10:00 – 17:00	**Board Meeting (only IACR Board members)**
17:00 – 21:00	**Welcome Reception and Registration**
April 27, 2009 Monday	
08:30	**Registration Desk open**
09:00 – 09:15	**Welcome / Opening Remarks**
Session 1 09:15 – 10:55	**Security, Proofs and Models I**
09:15 – 09:40	**Possibility and Impossibility Results for Encryption and Commitment Secure under Selective Opening.** *Mihir Bellare, Dennis Hofheinz, Scott Yilek*
09:40 – 10:05	**Breaking RSA Generically is Equivalent to Factoring** *Divesh Aggarwal, Ueli Maurer*
10:05 – 10:30	**Resettably Secure Computation** *Vipul Goyal, Amit Sahai*
10:30 – 10:55	**On the Security Loss in Cryptographic Reductions** *Chi-Jen Lu*
10:55 – 11:25	**Coffee Break**
Invited Talk 11:25 – 12:25	**Practice-Oriented Provable-Security and the Social Construction of Cryptography Phillip Rogaway**
12:45 – 13:45	**Lunch**
Session 2 13:45 – 15:25	**Hash Cryptanalysis**
13:45 – 14:10	**On Randomizing Hash Functions to Strengthen the Security of Digital Signatures** *Praveen Gauravaram, Lars R. Knudsen*
14:10 – 14:35	**Cryptanalysis of MDC-2** *Lars R. Knudsen, Florian Mendel, Christian Rechberger, Soeren S. Thomsen*

14:35 – 15:00	**Cryptanalysis on HMAC/NMAC-MD5 and MD5-MAC** *Xiaoyun Wang, Hongbo Yu, Wei Wang, Haina Zhang, Tao Zhan*
15:00 – 15:25	**Finding Preimages in Full MD5 Faster than Exhaustive Search** *Yu Sasaki, Kazumaro Aoki*
15:25 – 16:25	**Coffee Break + Poster Session Slot**
Session 3 16:25 – 17:40	**Group and Broadcast Encryption**
16:25 – 16:50	**Asymmetric Group Key Agreement** *Qianhong Wu, Yi Mu, Willy Susilo, Bo Qin, Josep Domingo-Ferrer*
16:50 – 17:15	**Adaptive Security in Broadcast Encryption Systems (with Short Ciphertexts)** *Craig Gentry, Brent Waters*
17:15 – 17:40	**Traitors Collaborating in Public: Pirates 2.0** *Olivier Billet, Duong-Hieu Phan*
April 28, 2009 Tuesday	
08:30	**Registration Desk open**
Session 4 09:00 – 10:15	**Cryptosystems I**
09:00 – 09:25	**Key Agreement from Close Secrets over Unsecured Channels** *Bhavana Kanukurthi, Leonid Reyzin*
09:25 – 09:50	**Order-Preserving Symmetric Encryption** *Alexandra Boldyreva, Nathan Chenette, Younho Lee, Adam O'Neill*
09:50 – 10:15	**A Double-Piped Mode of Operation for MACs, PRFs and PROs: Security beyond the Birthday Barrier** *Kan Yasuda*
10:15 – 11:10	**Coffee Break + Poster Session Slot**
Session 5 11:10 – 12:25	**Cryptanalysis**
11:10 – 11:35	**On the Security of Cryptosystems with Quadratic Decryption: The Nicest Cryptanalysis** *Guilhem Castagnos, Fabien Laguillaumie*
11:35 – 12:00	**Cube Attacks on Tweakable Black Box Polynomials** *Itai Dinur, Adi Shamir*
12:00 – 12:25	**Smashing SQUASH-0** *Khaled Ouafi, Serge Vaudenay*
12:25 – 13:30	**Lunch**

Extended Euclidean Algorithm

pb 22.4.09 (3)

$15 \in \mathbb{Z}_{87}$. Inverse?

	r	q	s	t	$s \cdot 87 + t \cdot 15 = r$
	87		1	0	$1 \cdot 87 + 0 \cdot 15 = 87$
→	15	5	0	1	$0 \cdot 87 + 1 \cdot 15 = 15$
(17) − 5·(15):	12	1	1	−5	$1 \cdot 87 + (-5) \cdot 15 = 12$
(15) − 1·(12):	3	4	−1	6	$(-1) \cdot 87 + 6 \cdot 15 = 3$
	0		5	−29	$5 \cdot 87 + (-29) \cdot 15 = 0$ ← Xcheck!

$14 \in \mathbb{Z}_{93}$ Inverse?

28.4.09 (1)

r	q	s	t	
93		1	0	
14	6	0	1	
9	1	1	−6	
5	1	−1	7	
4	1	2	−13	
1	4	−3	20	→ $1 = (-3) \cdot 93 + 20 \cdot 14$
0		14	−93	i.e. $1 = (20)\,14$ in $\mathbb{Z}_{93}$

Xcheck: ok!

$0 = 14 \cdot 93 + (-93) \cdot 14$

In our first example we did not find a solution. But we notice that 3 divides 87 and 15.

So any combination

$$b \cdot 15 + k \cdot 87$$

is divisible by 3. BUT 1 is not divisible by 3. So there is no solution!

Thm

The Extended Euclidean Algorithm finds a solution (s,t) of

$$s \cdot a + t \cdot b = 1$$

or produces a non-trivial divisor of a,b which proves that no solution exists.

Actually, the last non-zero remainder r is a (the) greatest common divisor of a and b.

Moreover: over $\mathbb{Z}$ the runtime is $O(n^2)$. $\square$

(n = bitsize of a and b)

As a corollary we obtain a description of the invertible elements in $\mathbb{Z}_N$:

$$\mathbb{Z}_N^\times = \{ a \in \mathbb{Z}_N \mid \overbrace{\exists b \in \mathbb{Z}_N : b \cdot a = 1}^{a \text{ is invertible}} \}$$

$$= \{ a \in \mathbb{Z}_N \mid \gcd(a, N) = 1 \}$$

↑ greatest common divisor

Chinese Remainder Theorem

Assume $N = m_1 \cdot m_2$,

$\gcd(m_1, m_2) = 1$.

Then

$$\mathbb{Z}_N \xrightarrow{\simeq} \mathbb{Z}_{m_1} \times \mathbb{Z}_{m_2}$$

$$a \longmapsto (a \bmod m_1, a \bmod m_2)$$

is an isomorphism!

Consider: $N = 15 = 3 \cdot 5$

$\mathbb{Z}_3$ \ $\mathbb{Z}_5$	0	1	2	3	4
0	0	6	12	3	9
1	10	1	7	13	4
2	5	11	2	8	14

counterexample

$N = 4 = 2 \cdot 2$

$\mathbb{Z}_2$ \ $\mathbb{Z}_2$	0	1
0	0,2	?
1	?	1,3

How to find $a \in \mathbb{Z}_N$ mapping to $(b, c) \in \mathbb{Z}_{m_1} \times \mathbb{Z}_{m_2}$?

Step 0: Use that the CRT-map is a homomorphism and write

$$(b, c) = \underbrace{\check{b} \cdot (1,0)}_{\text{repeated addition.}} + \check{c} \cdot (0,1).$$

Assume that we have

$$a_1 \longmapsto (1,0),$$
$$a_2 \longmapsto (0,1).$$

Then

$$\check{b} \cdot a_1 + \check{c} \cdot a_2 \longmapsto \check{b} \cdot (1,0) + \check{c} \cdot (0,1)$$
$$\| \qquad\qquad\qquad\qquad\qquad\qquad \| \qquad\qquad\qquad\qquad\qquad\qquad \|$$
$$(b, c) \qquad\qquad\qquad\qquad\qquad\qquad (b, c)$$

Step 1: Reformulate the question: we look for a_1 such that

$$a_1 = 1 - k_1 \cdot m_1 \text{ for some } k_1$$

and

$$\| \quad a_2 = 0 + k_2 \cdot m_2 \text{ for some } k_2,$$

i.e.

$$\underbrace{k_1 \cdot m_1}_{} + \underbrace{k_2 \cdot m_2}_{} = 1.$$

We can find k_1, k_2 by EEA(m_1, m_2). Then $a_1 = k_2 m_2$. And actually $a_2 = k_1 m_1$.

This works since by assumption the gcd of m_1 and m_2 is 1.

We get

$$a = \check{b} \cdot a_1 + \check{c} \cdot a_2$$

$$= \check{b} \cdot k_2 m_2 + \check{c} \cdot k_1 m_1 .$$

Example Find $a \in \mathbb{Z}_{15}$ such that

$$a \longmapsto (2,2) \in \mathbb{Z}_3 \times \mathbb{Z}_5 .$$

Run EEA $(3,5)$ and obtain

$$\underbrace{2 \cdot 3}_{a_2} + \underbrace{(-1) \cdot 5}_{a_1} = 1 .$$

$\check{a}_1 \bmod 3 = 1$	$\check{a}_2 \bmod 3 = 0$	2
$\check{a}_1 \bmod 5 = 0$	$\check{a}_2 \bmod 5 = 1$	2

$\cdot 2$ $\cdot 2$

$$a = 2 \cdot ((-1) \cdot 5) + 2 \cdot (2 \cdot 3) = 2 .$$

Why don't we stop here?
What's the basic difference
between $(\mathbb{Z}_N, +)$,
$(\mathbb{Z}_N, \cdot)$?

Actually, for any finite group G
and any element g,
there is a number $\ell > 0$ such
that
$$g^\ell = 1.$$
(In $g^0, g^1, g^2, \dots, g^N, \dots$ you find
somewhere twice the same result, i.e.
$\ell_1 > \ell_2 : g^{\ell_1} = g^{\ell_2}$. Since we can divide we have
$g^{\ell_1 - \ell_2} = 1$.)

And so the map
$$\begin{array}{ccc} \{g^\alpha \mid \alpha \in \mathbb{Z}\} & & \\ \| & & \\ (\mathbb{Z}_\ell, +) & \longrightarrow & \langle g \rangle \subset (G, \cdot) \\ \alpha \bmod \ell & \longmapsto & g^\alpha \end{array}$$
is a homomorphism and
actually an isomorphism between
$(\mathbb{Z}_\ell, +)$ and $(\langle g \rangle, \cdot)$.

cpb
28.4.09
(7)

Compute this?

Silly answer: compute

$$g^x = \underbrace{g \cdot g \cdot \ldots \cdot g}_{x \text{ of them}}$$

$(x-1)$ operations in G.

But $x \approx 2^n$ where n = #bits in q.

so the runtime is exponential.

:(

Better: use square & multiply.

Example Compute g^{25} cleverly!

1 10 100 1000 10000 11000 11001

$g, g^2, g^4, g^8, g^{16}, g^{24}, g^{25}$

store

Square as often as possible,
multiply with most latest fitting til done.

runtime: $O(n)$ op's in G.

memory: $O(n)$ elt's in G.

Example Compute g^{25} by square & multiply:

1 10 11 110 1100 11000 11001

$g, g^2, g^3, g^6, g^{12}, g^{24}, g^{25}$

time: $O(n)$

space: $3 \in O(1)$

What about finding α

given $a = g^{\alpha}$?

This is called the

Discrete Logarithm Problem.

And: for $(\mathbb{Z}_N, +)$ it's easy,

for $(\mathbb{Z}_N^{\times}, \cdot)$ it's often probably difficult.

14:00 – 18:00	**Social Program - Meeting Point 13:40 in the foyer!** • **City Walking Tour (2,5 h)** • **City Bike Tour (3h)** • **Chocolate Museum (1h)** • **Wallraf-Richartz Museum (1h)** • **Boat Cruise (1h)**
18:00 – 23:00	**Rump Session**
April 29, 2009 Wednesday	
09:00 – 9:15	**Best Paper Award Ceremony**
Session 6 **09:15 – 10:30**	**Cryptosystems II**
09:15 – 09:40	**Practical Chosen Ciphertext Secure Encryption from Factoring** *Dennis Hofheinz, Eike Kiltz*
09:40 – 10:05	**Realizing Hash-and-Sign Signatures under Standard Assumptions** *Susan Hohenberger, Brent Waters*
10:05 – 10:30	**A Public Key Encryption Scheme Secure against Key Dependent Chosen Plaintext and Adaptive Chosen Ciphertext Attacks** *Jan Camenisch, Nishanth Chandran, Victor Shoup*
10:30 – 11:25	**Coffee Break + Poster Session Slot**
Invited Talk **11:25 – 12:25**	**Cryptography without (Hardly any) Secrets ?** *Shafi Goldwasser*
12:25 – 13:45	**Lunch**
Session 7 **13:45 – 15:25**	**Security, Proofs and Models II**
13:45 – 14:10	**Salvaging Merkle-Damgard for Practical Applications** *Yevgeniy Dodis, Thomas Ristenpart, Thomas Shrimpton*
14:10 – 14:35	**On the Security of Padding-Based Encryption Schemes (Or: Why we cannot prove OAEP secure in the Standard Model)** *Eike Kiltz, Krzysztof Pietrzak*
14:35 – 15:00	**Simulation without the Artifical Abort: Simplified Proof and Improved Concrete Security for Waters' IBE Scheme** *Mihir Bellare, Thomas Ristenpart*
15:00 – 15:25	**On the Portability of Generalized Schnorr Proofs** *Jan Camenisch, Aggelos Kiayias, Moti Yung*
15:25 – 15:45	**Coffee Break**

April 29, 2009 Wednesday	
Session 8 **15:45 – 16:35**	**Side Channels**
15:45 – 16:10	**A Unified Framework for the Analysis of Side-Channel Key Recovery Attacks** *Francoix-Xavier Standaert, Tal Malkin, Moti Yung*
16:10 – 16:35	**A Leakage-Resilient Mode of Operation** *Krzysztof Pietrzak*
16:45 – 18:00	**IACR Membership Meeting**
19:00 – 23:00	**Conference Dinner - Boat Cruise** **(2h, the boat leaves the pier of K&D at 20:00)**
April 30, 2009 Thursday	
Session 9 **09:00 – 10:40**	**Curves**
09:00 – 09:25	**ECM on Graphics Cards** *Daniel Bernstein, Tien-Ren Chen, Chen-Mou Cheng, Tanja Lange, Bo-Yin Yang*
09:25 – 09:50	**Double-Base Number System for Multi-Scalar Multiplications** *Christophe Doche, David Kohel, Francesco Sica*
09:50 – 10:15	**Endomorphisms for Faster Elliptic Curve Cryptography on a Large Class of Curves** *Steven Galbraith, Xibin Lin, Michael Scott*
10:15 – 10:40	**Generating Genus Two Hyperelliptic Curves over Large Characteristic Finite Fields** *Takakazu Satoh*
10:40 – 11:10	**Coffee Break + Poster Session Slot**
Session 10 **11:10 – 12:25**	**Randomness**
11:10 – 11:35	**Optimal Randomness Extraction from a Diffie-Hellman Element** *Pierre-Alain Fouque, Sebastien Zimmer, David Pointcheval, Celine Chevalier*
11:35 – 12:00	**Verifiable Random Functions from Identity-based Key Encapsulation** *Michel Abdalla, Dario Catalano, Dario Fiore*
12:00 – 12:25	**A New Randomness Extraction Paradigm for Hybrid Encryption** *Eike Kiltz, Krzysztof Pietrzak, Martijn Stam, Moti Yung*
12:25 – 12:40	**Closing Remarks**

cpb
28.4.09
(8)

What about finding α

given $a = g^\alpha$?

This is called the

Discrete Logarithm Problem.

And: for $(\mathbb{Z}_N, +)$ it's easy,

for $(\mathbb{Z}_N^\times, \cdot)$ it's often probably difficult.

12.5.09
(7)

$\exp_g$: sends $\mathbb{Z}_\ell \to \langle g \rangle \subset G$ (subgroup of G generated by g; G any finite group)

$\mathbb{Z}_\ell$	$\langle g \rangle$
$+$	$\circ$
α	g^α

isomorphism!

$\ell = \mathrm{ord}(g) = \min\{\lambda > 0 ; g^\lambda = 1\} = \#\langle g \rangle$.

Triangle note:
It's injective. Assume you find α, β with $g^\alpha = g^\beta$. Then $g^{\alpha-\beta} = 1$
But of course $-\ell < \alpha - \beta < \ell$, wlog. $0 \le \alpha - \beta < \ell$
(otherwise exchange α, β). But $g^\lambda \neq 1$ for $0 < \lambda < \ell$.
Thus $\alpha = \beta$. That is: $\exp_g$ is injective! ∎

Compute the inverse of $\exp_g$ means to solve a

Discrete Logarithm Problem

Given $a \in \langle g \rangle$

Look for $\alpha \in \mathbb{Z}_e$ s.t. $a = g^{\alpha}$.

Example

$G = (\mathbb{Z}_N, +)$: DLP solvable in time $O((\log_2 N)^2)$ using the Extended Euclidean Algorithm

$G = (\mathbb{Z}_N^{\times}, \cdot)$:

DLP solvable in time $\tilde{O}(\sqrt{e})$

where e = largest prime factor of $\#\mathbb{Z}_N^{\times}$.

$\#\mathbb{Z}_N^{\times} = \#\{a \bmod N \mid \gcd(a,N) = 1\}$ ← decide by EEA!

Case: $N = p$ prime

$\#\mathbb{Z}_p^{\times} = p - 1$.

eg.: $p = 101$. $p - 1 = 100 = 2^2 \cdot 5^2$.

Thus there is a DLOG-algorithm of runtime roughly $\sqrt{5} \ll \sqrt{100}$.

Solution for DLOG in time $\sqrt{e}$:

Given $g \in G$, of order e.

Given $a \in \langle g \rangle$.

Find $\alpha \in \mathbb{Z}_e$ s.t. $g^\alpha = a$.

"Meet in the middle" attack

→ Baby step - giant step.

Choose $b = \lceil \sqrt{e} \rceil$ (as $b = 2^{\lceil \frac{1}{2} \log_2 e \rceil}$.)

Write $\alpha = \alpha_1 \cdot b + \alpha_0$ with $0 \le \alpha_1, \alpha_0 < b$.

Thus we have to solve this:

$$(g^b)^{\alpha_1} \cdot g^{\alpha_0} = a$$

$$(g^b)^{\alpha_1} = a\, g^{-\alpha_0}.$$

Compute a table with

$$[\alpha_0,\ a \cdot g^{-\alpha_0}] \quad \text{for } 0 \le \alpha_0 < b.$$

Then run through possible $0 \le \alpha_1 < b$

and compute $(g^b)^{\alpha_1} = (g^b)^{\alpha_1 - 1} \cdot g^b$

and check whether it is in the table.

deterministic runtime

$O(b) = O(\sqrt{e})$ operations (& space)

Better:

Pollard-ρ

runs in heuristic expected time $O(\sqrt{e})$

and space $O(1)$.

Compute many

$$g^{\alpha} a^{\beta}.$$

If we find a collision i.e.

$$g^{\alpha_1} a^{\beta_1} = g^{\alpha_2} a^{\beta_2}$$

then

$$g^{\alpha_1 - \alpha_2} = a^{\beta_2 - \beta_1}$$

i.e.

$$g^{\frac{\alpha_1 - \alpha_2}{\beta_2 - \beta_1}} = a \quad \text{if } \beta_2 - \beta_1 \in \mathbb{Z}_e^{\times}.$$

We expect to choose $O(\sqrt{e})$ many (α, β) until we find a collision.

Floyd's trick saves memory!

Fix a function $(\alpha, \beta, g^{\alpha} a^{\beta})$ and maps it to a new one of this form.

Now proceed:

Pick $\alpha_0, \beta_0 \in_R \mathbb{Z}_e$,

compute $g^{\alpha_0} a^{\beta_0} =: x_0 =: y_0$

~~Repeat~~: $x_0 := (\alpha_0, \beta_0, g^{\alpha_0} a^{\beta_0})$,

$y_0 := (\alpha_0, \beta_0, g^{\alpha_0} a^{\beta_0})$.

$i := 1$

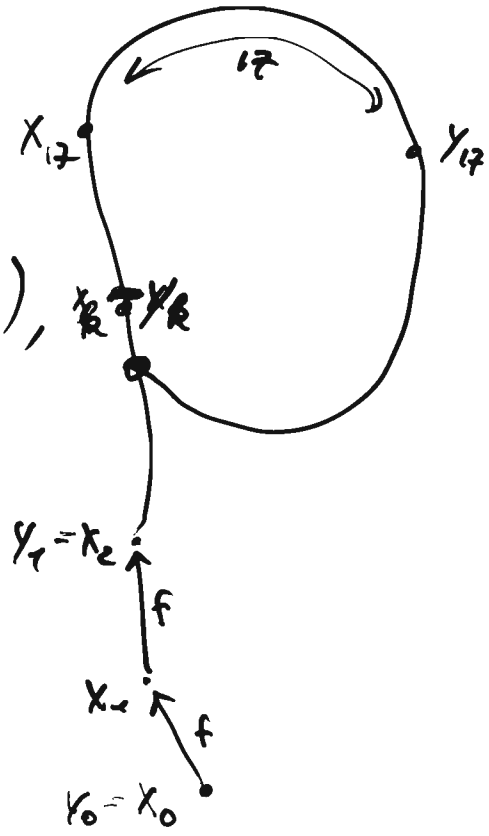
Repeat

$x_i = f(x_{i-1})$,

$y_i = f(f(y_{i-1}))$,

$i = i + 1$

until $(x_i)_3 = (y_i)_3$.



We lost the precise analysis, but in practice it works!

So heuristic expected runtime $O(\sqrt{e})$.

and space $O(1)$

Can we use specific properties of $\mathbb{Z}_N^\times$ to get an even better algorithm?

Yes: $c \cdot \sqrt{\log_2 N \ \log_2 \log_2 N}$

runtime 2
(& space)

This is subexponential but not polynomial in $\log_2 N$.

Actually, we can use that elements in $\mathbb{Z}_N^\times$ come from integers in $\mathbb{Z}$ which have a unique prime factorization!

For practical situations we want that the runtime of any known attack is large.

This leads to $\log_2 N \approx 2048$ (3072).

to ensure runtime $\approx 2^{128}$

128 - bit security.

So $\mathbb{Z}_p^\times$ is ok,

but maybe there's sth better?

Remember: DLP can be solved in time $O(\sqrt{e})$

if $e = \mathrm{ord}(g)$ prime.

for any group.

As just seen: within $G = \mathbb{Z}_p^\times$ we can do faster.

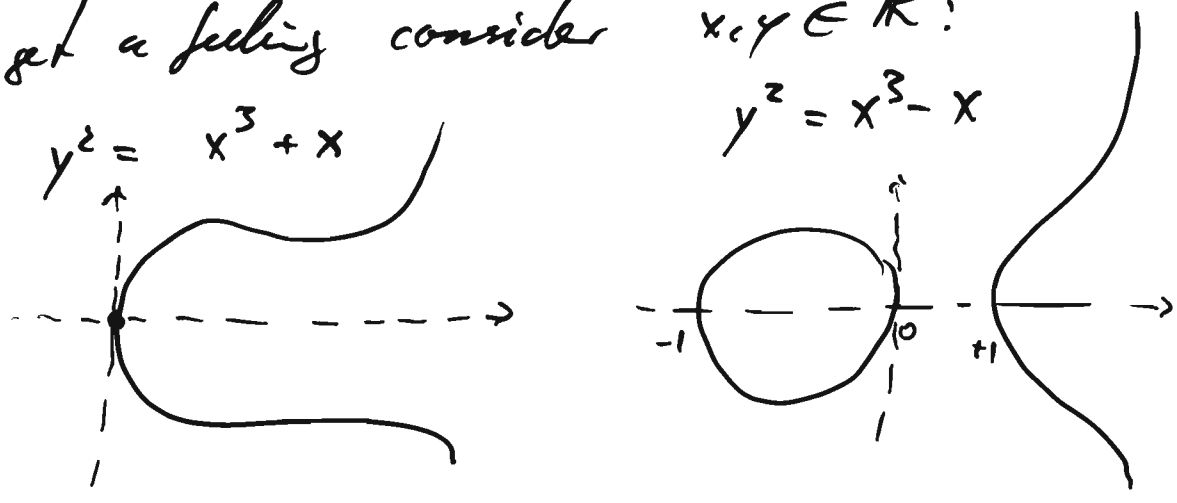
Q: Are there groups where we can not do faster?

Answer: Probably, yes!

Elliptic Curves

An elliptic curve is (most of the time) given by an equation $y^2 = x^3 + ax + b$ where $x, y \in \mathbb{Z}_p$ with p prime.

To get a feeling consider $x, y \in \mathbb{R}$:



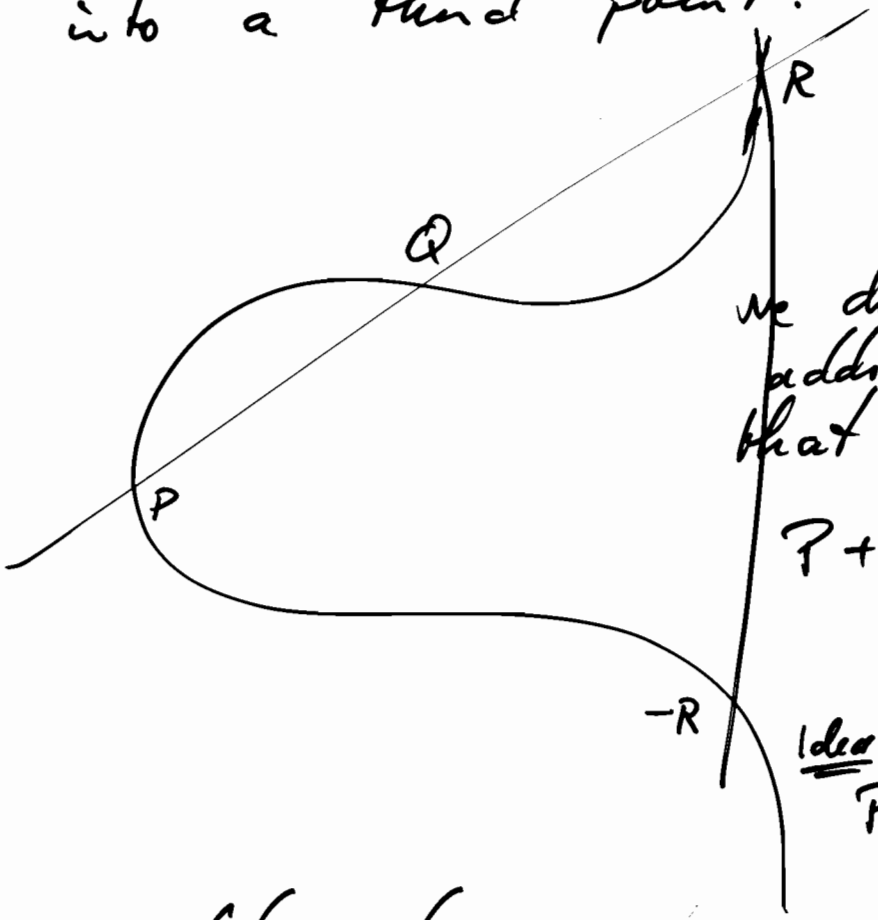
Group structure?

Basic observation:

it's a degree 3 curve

ie. any line intersects the curve in (at most) three points.

We would to combine two points into a third point.



We define the addition such that

$$P + Q + R = 0.$$

$\Leftrightarrow$

Idea $P + Q := -R$

We would need more to define $R \longmapsto -R$.

Notice: $R \longmapsto -R \longmapsto +R = R$.

Vertical lines intersect only twice.

(Remember: $y^2 = x^3 + ax + b$.)

We add a point 0 at infinity lying on any vertical line. Then $R + (-R) = -0 = 0$.

cp 6
13.05.09
(1)

Formulas for the group operation:

(1) Negation

$R = (x_3, y_3)$: $y_3^2 = x_3^3 + a x_3 + b$

Then $-R = (x_3, -y_3)$

Then $R, -R$ lie on the same vertical line and both on the curve.

Obviously $-(-R) = R$.

(2a) Addition of different points P, Q, where $P \neq -Q$, $P \neq O \neq Q$

$P = (x_1, y_1)$, $Q = (x_2, y_2)$

$\ell_{P,Q}$: $y = \alpha x + \beta$

we find $\alpha = \frac{y_2 - y_1}{x_2 - x_1}$ which is nice since by assumption $x_1 \neq x_2$.

and $\beta = y_1 - \alpha x_1$.

So we look for the third point R on $\ell_{P,Q}$ and the curve E: $R = (x_3, y_3)$

$$y_3^2 = x_3^3 + a x_3 + b$$

$$y_3 = \alpha x_3 + \beta .$$

Thus

$$p(x_3) = -(\alpha x_3 + \beta)^2 + (x_3^3 + a x_3 + b) = 0$$

We already know two solutions for this equation: x_1, x_2.

Thus can compute

$$p(x) = \underline{(x-x_1)(x-x_2)} \cdot (x-x_3) + 0.$$

ie.

the x^2-coefficient of p must be $-(x_1+x_2+x_3)$.

On the other hand it is by def. of p:

$$-\alpha^2$$

Thus $$x_3 = \alpha^2 - x_1 - x_2.$$

and $$y_3 = \alpha x_3 + \beta = \alpha(x_3 - x_1) + y_1$$

Thus if we let $P+Q = (x_4, y_4)$,

then $$\alpha = \frac{y_2 - y_1}{x_2 - x_1},$$

$$x_4 = \alpha^2 - x_1 - x_2,$$

$$y_4 = -y_1 + \alpha(x_1 - x_4).$$

(26) Addition of P with itself: $P+P$.

→ use tangent: $\alpha = \frac{3x_1^2 + a}{2y_1}$.

Everything else is as above.

(2c) Addition of P and $Q = -P \neq O$

$$P + (-P) := O$$

(2d) Addition of P and O

$$P + O := P$$

$$O + P := P$$

$$O + O := O$$

Now we have

(P)

- a set $\quad E = \{ (x,y) \in \mathbb{F}^2 \mid y^2 = x^3 + ax + b \} \dot\cup \{O\}$
- and a negation $\quad - : E \longrightarrow E$, $P \longmapsto -P$, $O \longmapsto O$
- and a addition $\quad + : E \times E \longrightarrow E$, $(P, Q) \longmapsto P + Q$.
- and a neutral element: $\quad O \in E$.

A: defer ...

N: O is neutral by definition.

I: $P + (-P) = O$.

C: Inspect the definition: $P + Q = Q + P$.

cp5
15.5.09
(4)

Associativity?

Tricky.

- Historical: Divisors, Picard-group and Riemann-Roch. ☺
- Geometrical: Possible, but... cumbersome.
- Algebraically?

Fix $P = (x_1, y_1)$, $P \neq \pm Q$

$Q = (x_2, y_2)$, $Q \neq \pm R$

$R = (x_3, y_3)$ $P+Q \neq \pm R$

non-special! $Q+R \neq \pm P$

Find a formula for

$(P+Q)+R$

and

$P+(Q+R)$

and compare the formulas.

Not really satisfying.

But it works!

Our goal is:
construct groups with
a point of

known order.

In particular, we should be able to
compute the size of the group.

$$\# \mathbb{Z}_N^+ = N$$

$$\# \mathbb{Z}_p^\times = p - 1 \qquad \text{(assuming } p \text{ prime)}$$

$$\# E_{a,b} = ? \qquad (a, b \in \mathbb{F}_q)$$

$$\| \{ (x,y) \in \mathbb{F}_q^2 \mid y^2 = x^3 + ax + b \} \cup \{\mathcal{O}\}$$

↑ # of elements in the field

(Side remark: for every prime power q
there exists essentially one field
with q elements. If p is prime,
then $\mathbb{F}_p = \mathbb{Z}_p$.)

Count ver 0: Try all pairs $(x,y) \in \mathbb{F}_q$
$O(q^2)$ and check if $y^2 = x^3 + ax + b$.
Output count if true.

cp6
19.5.09
(2)

Count ver 1: Try all values $x \in \mathbb{F}_q$ and decide whether 2, 1, 0 solutions exists with $y^2 = x^3 + ax + b$.

If you can: take square root of $x^3 + ax + b$.

1 solution $\Leftrightarrow$ $\neq 0$.

$O(q)$

Deciding whether $x^3 + ax + b$ is a square modulo p can be done in time $O(n^2)$ where n = bitlength(p).

Jacobi-symbol:

$$\left(\frac{c}{d}\right) := \prod_i \left(\frac{c}{d_i}\right)^{e_i}$$

$d = d_i^{e_i}$, d_i are all prime

and $\left(\frac{c}{p}\right) = \begin{cases} +1 & \text{if } c \text{ square mod } p \\ 0 & \text{if } c = 0 \\ -1 & \text{if } c \text{ non-square} \end{cases}$

Cool thing:

$$\left(\frac{c}{d}\right) = \left(\frac{c \bmod d}{d}\right)$$

$$= (-1)^{\cdots} \left(\frac{d}{c}\right).$$

$= (-1)^{\cdots} \left(\frac{d \bmod c}{c} \right)$

until $\left(\frac{2}{p}\right) = \ldots$, $\left(\frac{-1}{p}\right) = \ldots$

NOTE: This is very similar to the Euclidean algorithm.

→ same runtime. ⌟

In particular, we learn that

$$\# E_{a,b} \leq 2q + 1.$$

Hasse proves much more:

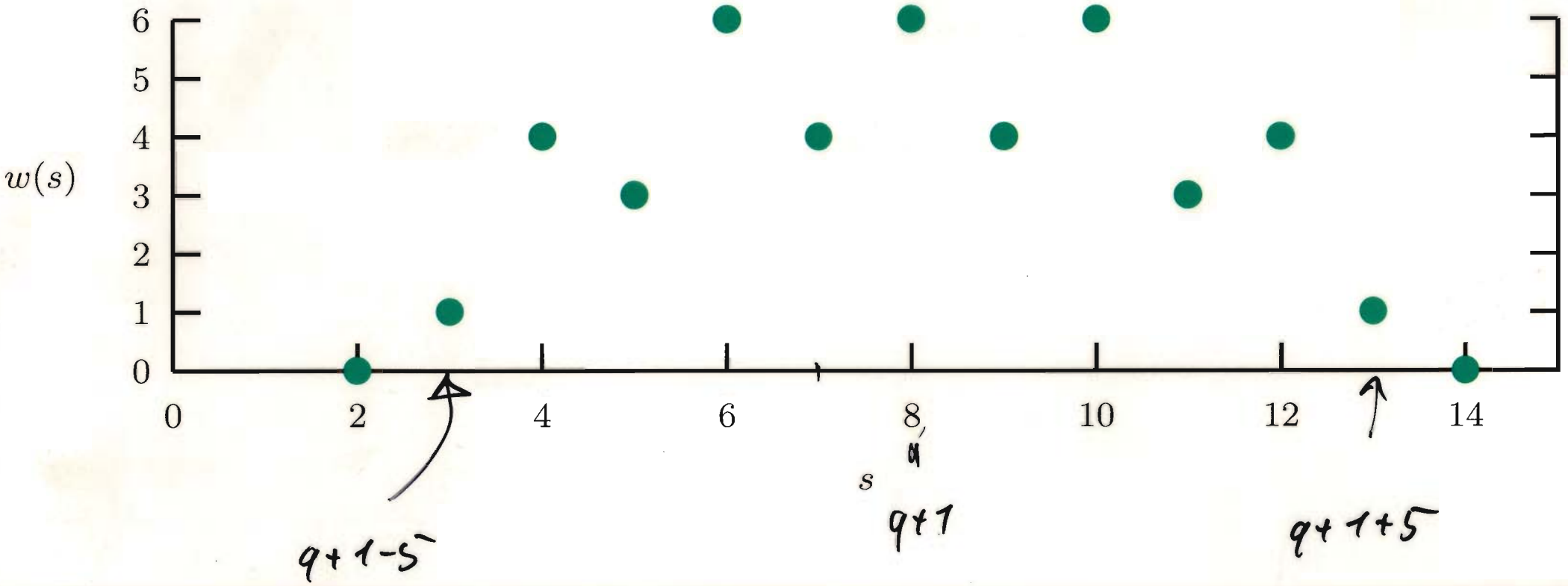
$$\# E_{a,b} = q + 1 - t$$

where $$|t| \leq 2\sqrt{q}.$$

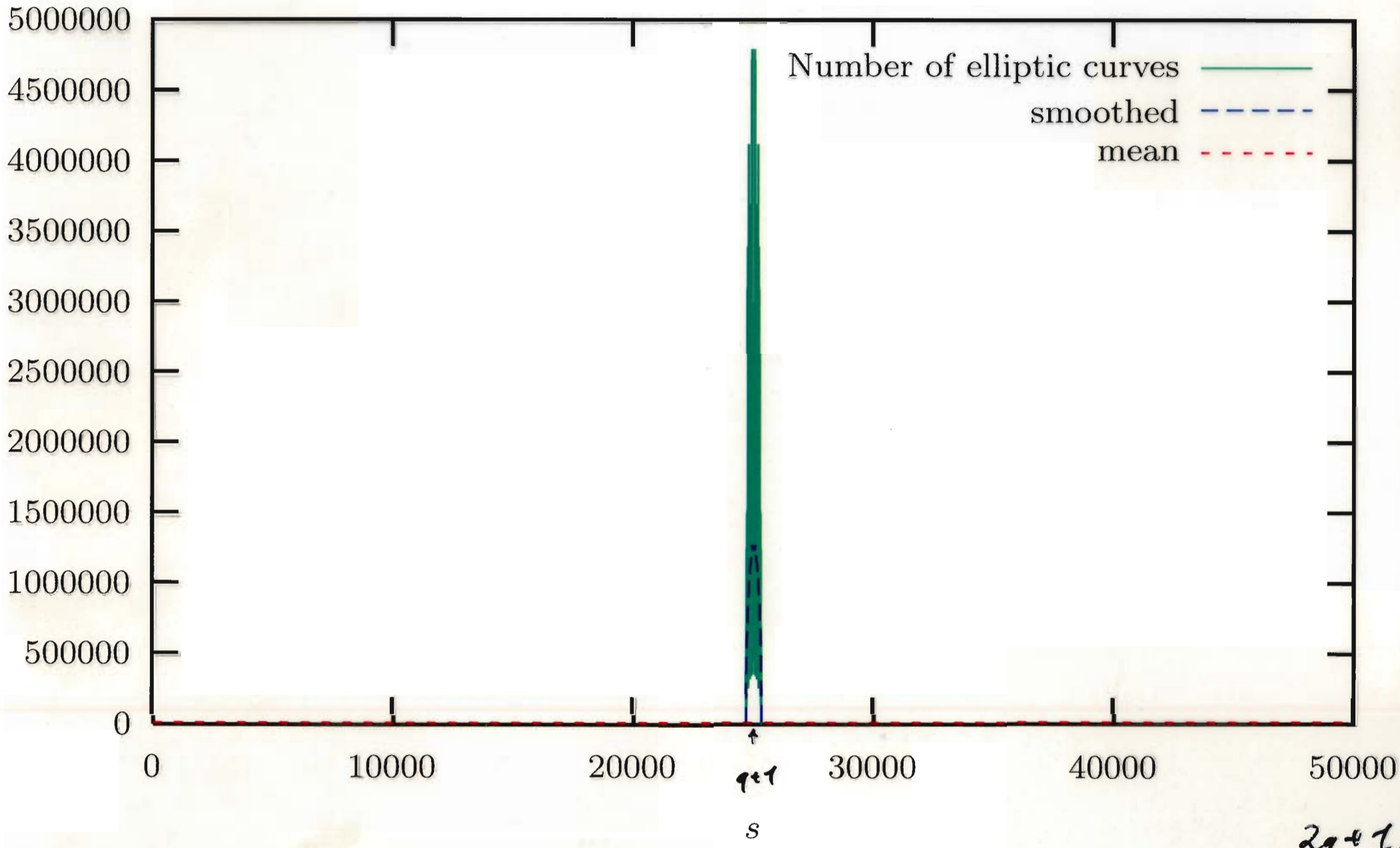
Extra observation: for every curve $E_{a,b}$ there is another one $E_{c,d}$ such that $\# E_{a,b} = q+1-t$, $\# E_{c,d} = q+1+t$.

$p = q = 7$

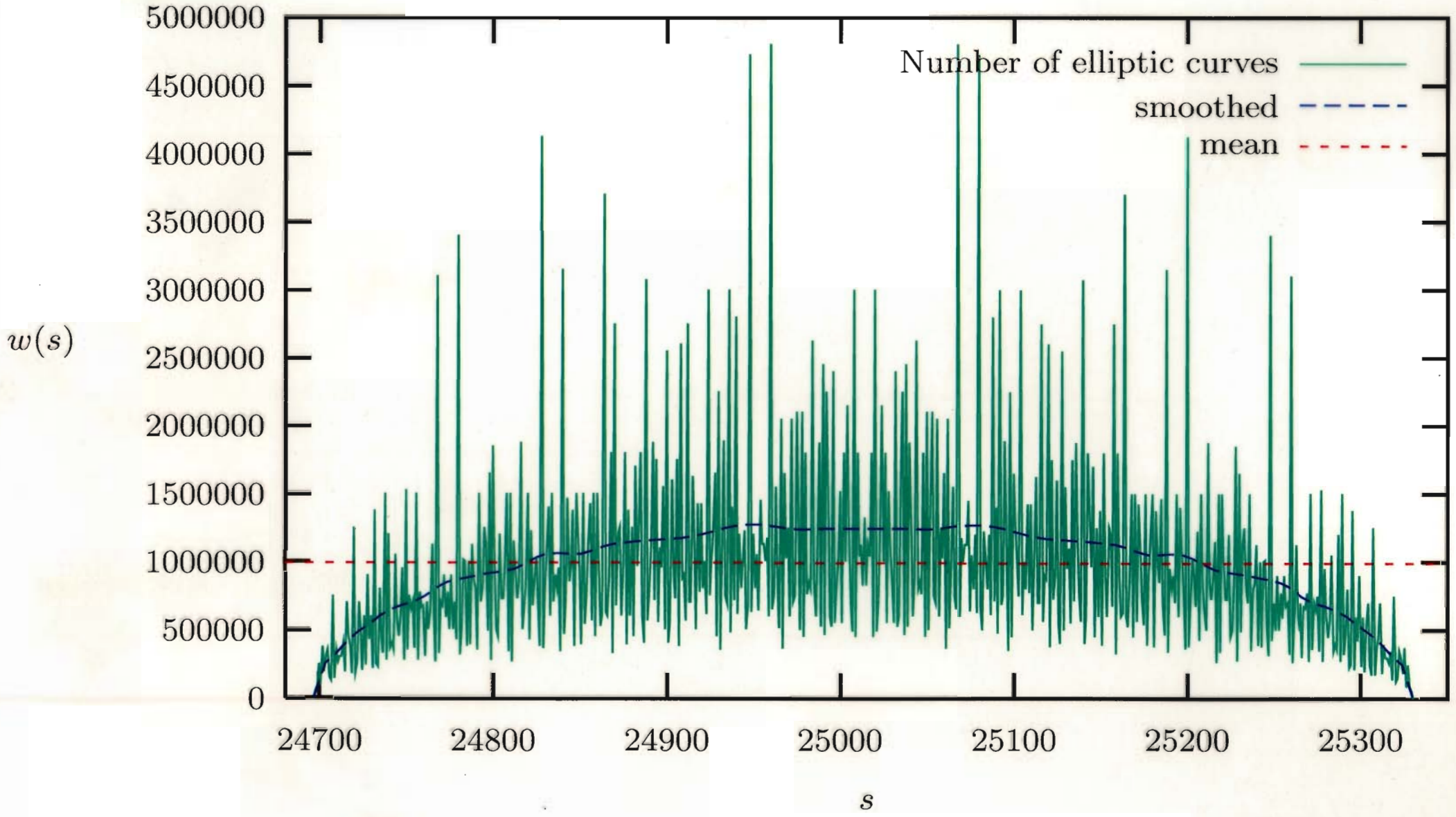
$\lfloor 2\sqrt{7} \rfloor = 5$



p=q = 25013. # curves



2q+1.


5000000
4500000
4000000
3500000
3000000
2500000
2000000
1500000
1000000
500000
0
$w(s)$
24700
24800
24900
25000
25100
25200
25300
s
Number of elliptic curves
smoothed
mean

pb
19.5.09
(4)

So we have a way of computing $\#E_{a,b}$
in time $O(q \cdot n^2)$

$q \leq 2^n$; $q \geq 1$

Schoof: pre-Schoof:

Thm (Lagrange) G group, finite.
Then for any $a \in G$ we have $a^{\#G} = 1$.
In particular, $\mathrm{ord}(a) \mid \#G$.

Pf of the "in particular":
We know that $a^{\mathrm{ord}(a)} = 1$, $a^{\#G} = 1$.
Use the EEA to ~~write~~ compute

$$g = s \cdot \mathrm{ord}(a) + t \cdot \#G$$

with $g = \gcd(\mathrm{ord}(a), \#G) \leq \mathrm{ord}\, a$.

Now:
$$a^g = \underbrace{\left(a^{\mathrm{ord}(a)}\right)^s}_{=1} \cdot \underbrace{\left(a^{\#G}\right)^t}_{=1} = 1.$$

Since $g > 1$, $g \leq \mathrm{ord}(a)$, only $g = \mathrm{ord}(a)$ is possible of the order.

□

So just guess a point T on E.

Compute $\alpha \cdot P$ for

$$\alpha \in q+1-\lfloor 2\sqrt{q} \rfloor \ \ldots \ q+1+\lfloor 2\sqrt{q} \rfloor .$$

runtime: $O(\sqrt{q} \cdot n^3)$

$= 2^{n/2}$ still exponential !

Schoof:

SEA $O(n^8)$

Try to determine

$t \bmod \ell$

for some prime ℓ.

That is possible! and in time poly(n).

Do this for $O(n)$ many primes

so that $\prod \ell \geq 4\sqrt{q}$

and then plug things with the

help of the chinese remainder theorem.

Lakes Elkies and Atkies did many improvements.

We will usually need
a group G and
a point $g \in G$ of known order.

If $G = E$ with $\#E$ = prime
then just pick any point $P \in E \setminus \{O\}$.

Good news:
for the DLP on a 'random' elliptic curve nothing better but the generic algorithms are known.

So: choose q a 256-bit,
choose $a, b \in \mathbb{F}_q$,
compute $\#E_{a,b}$
eventually repeat until that is prime
pick $P \in E_{a,b} \setminus \{O\}$.

This will give us 128-bit security (unless...)

Actually, in the verification we have a = public key
b, s = signature
m = message.

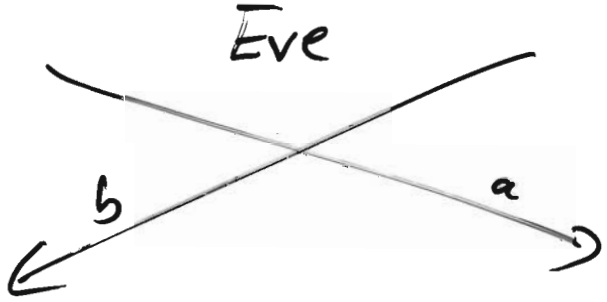
A first crypto algorithm:

Pick G and $g \in G$ of known order e.

Aim: agree on a common key

Diffie-Hellman key agreement

Aline	Boris
$\alpha \in_R \mathbb{Z}_e$.	$\beta \in_R \mathbb{Z}_e$.
$a = g^\alpha$	$b = g^\beta$



$k_1 = b^\alpha$ $\qquad$ $k_2 = a^\beta$

Observation: $k_1 = k_2$.

$$k_1 = b^\alpha = (g^\beta)^\alpha = g^{\beta \cdot \alpha}$$

$$k_2 = a^\beta = (g^\alpha)^\beta = g^{\alpha \cdot \beta}$$

Eavesdropper: has g, a, b

wants k.

$$(g, g^\alpha, g^\beta) \longmapsto g^{\alpha\beta}.$$

Diffie-Hellman-Problem (DHT) in G

$(g, g^\alpha, g^\beta) \longmapsto g^{\alpha\beta}$

Obviously:

If Eve can solve DLP in G
then she can solve DHP in G.

The other direction is open ... almost ...

Thus DH is insecure
if DLP is easy.

We would like to have
if ... is difficult then DH is secure

DHPV

DLP?

(No) I Am in the middle attack

cpb
20.5.09
①

Alice

$\alpha \in_R \mathbb{Z}_e$

$a = g^{\alpha}$

(Eve)
Wilma

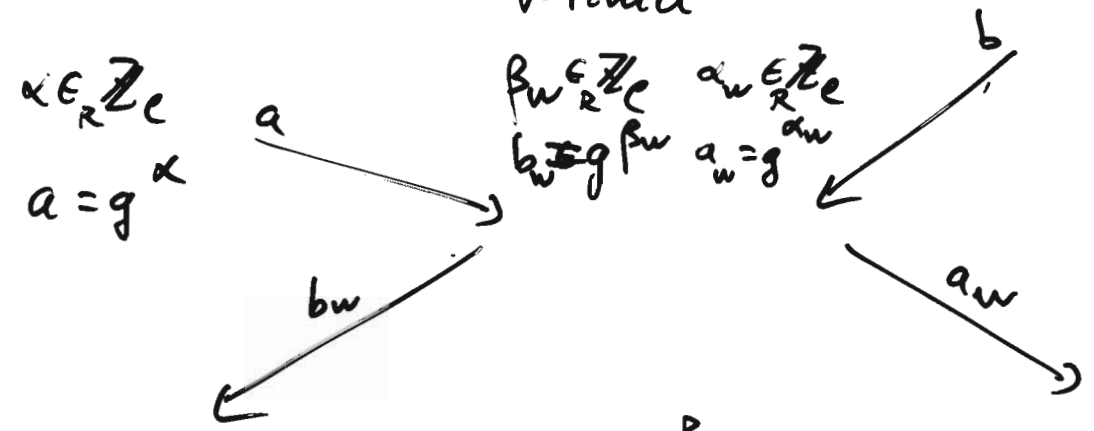
$\beta_w \in_R \mathbb{Z}_e \quad \alpha_w \in_R \mathbb{Z}_e$

$b_w = g^{\beta_w} \quad a_w = g^{\alpha_w}$

Boris

$\beta \in_R \mathbb{Z}_e$

$b = g^{\beta}$



$k_1 = b_w^{\alpha} = g^{\alpha\beta_w}$

$k_{2,w} = a^{\beta_w} = g^{\alpha\beta_w} = k_1$

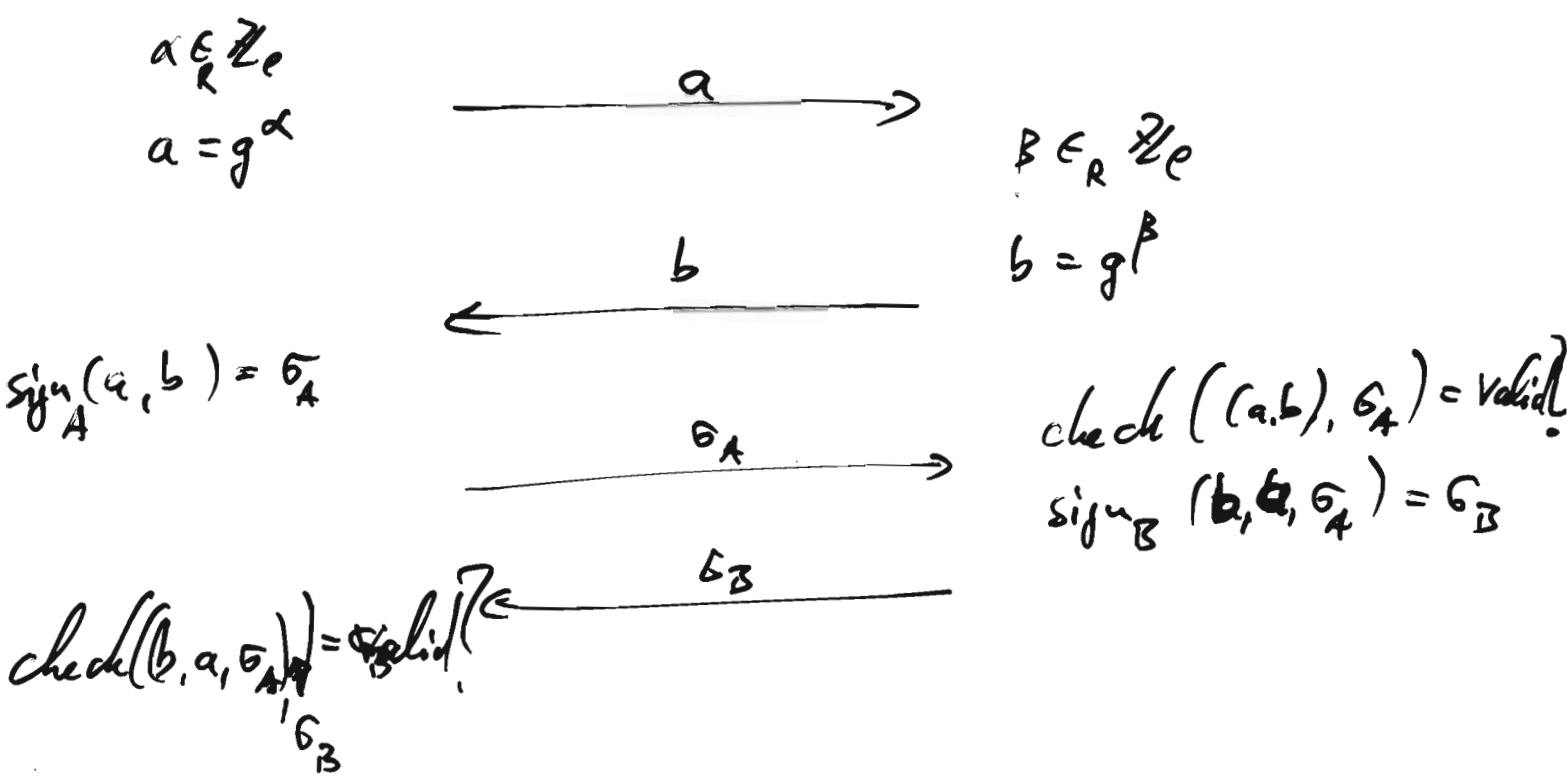
$k_2 = a_w^{\beta} = g^{\alpha_w \cdot \beta}$

$k_{1,w} = b^{\alpha_w} = g^{\alpha_w\beta} = k_2$

Now, Wilma acting as a proxy can read and manipulate every thing.

Authenticated key exchange

cpb
20.5.09
(2)



Now, Wilma cannot play the previous attack !

:)

ElGamal type signatures

cpb 20.5.09 (3)

$$a^{b^*} \; b^{\delta} = g^{\text{hash}(m)}$$

verification equation.

We now have groups where $a, b, g \in G$ can live. And we can make sure that the DLP is reasonably difficult (unless...). Assume further that #G is prime.

Actually, we always require $a, b, g \in \langle g \rangle$.

So where does δ live?

$$\delta \in \mathbb{Z}_\ell \quad \text{where} \quad \ell = \text{ord}\, g.$$

Next: $m \in ? \; \mathbb{Z}_\ell$!

But $\ell \approx 256$-bits

Most messages do not fit in 256 bits. → Either enlarge the group. Or: compress the message?

So no, we only need

$$\text{hash}(m) \in \mathbb{Z}_\ell.$$

We cannot

raise a group element a

to the power of a group element b

.

So we have to replace the exponent there!

Instead of

$$a^{b}$$

we put

$$a^{b^*}$$

For ~~the~~ any ElGamal type signature scheme

first

Setup: Choose a group G

and an element g

of ~~best prime, but certainly~~

known order ℓ.

Make sure that the DLP to the

basis g can be difficult.

(In particular, ℓ must contain

a large prime factor.)

Fix a hash function

$$\text{hash}: \{0,1\}^* \longrightarrow \mathbb{Z}_\ell .$$

pb
20.5.09
(5)

Choose a simple, easy to compute function

$$*: G \longrightarrow \mathbb{Z}_e .$$

Examples:

(1) $G = \mathbb{Z}_p^{\times}$, $g \in G$ of order $e = p-1$.

$\rightarrow$ ElGamal signatures.

(2) $G = \mathbb{Z}_p^{\times}$, $g \in G$ of order $e \mid p-1$, e prime!

where $p \approx 3072$ Bits,

$e \approx 256$ Bits.

$\rightarrow\rightarrow$ DSA (Digital Signature Algorithm)

(3) $G = E_{a,b}$ elliptic curve give by $a, b \in \mathbb{F}_q$,

$P \in G$ of order $e \mid \# E_{a,b}$, e prime!

where $q \approx 256$ bits

$l \approx 256$ bits

$\rightarrow\rightarrow$ EC DSA (Elliptic Curve Digital Signature Algorithm)

(i) only 80-bit security
(ii) broken.

ALGORITHMUS. SHA-1.
Eingabe: Eine Nachricht $x \in \{0,1\}^*$.
Ausgabe: Ein Hackwert $H \in \{0,1\}^{160}$.

Konstanten und Rundenfunktionen:

1. $h \leftarrow$ (67452301, EFCDAB89, 98BADCFE, 10325476, C3D2E1F0).

$$K_j \leftarrow \begin{cases} \texttt{5A827999}, & 0 \le j < 20, \quad (32 \text{ Bits von } \sqrt{2}) \\ \texttt{6ED9EBA1}, & 20 \le j < 40, \quad (32 \text{ Bits von } \sqrt{3}) \\ \texttt{8F1BBCDC}, & 40 \le j < 60, \quad (32 \text{ Bits von } \sqrt{5}) \\ \texttt{CA62C1D6}, & 60 \le j < 80. \quad (32 \text{ Bits von } \sqrt{7}) \end{cases}$$

$$f_j(B,C,D) = \begin{cases} (B \wedge C) \vee (\overline{B} \wedge D), & 0 \le j < 20, \\ B \oplus C \oplus D, & 20 \le j < 40, \\ (B \wedge C) \vee (C \wedge D) \vee (D \wedge B), & \\ B \oplus C \oplus D, & 60 \le j < 80. \end{cases}$$

Vorberechnungen:

2. Auffüllen: $\tilde{x} \leftarrow x|1|0^d|\langle|x|\rangle_{64}$ mit $0 \le d < 512$ so, daß $|\tilde{x}|$ ein Vielfaches von $512 = 16 \cdot 32$ ist.
3. Zerlege $\tilde{x}$ in 32-Bitworte: $\tilde{x} = x_0 x_1 x_2 \ldots x_{16m-1}$.
4. Initialisiere: $(H_1, H_2, H_3, H_4, H_5) \leftarrow h$.

Hauptberechnung:

5. For $i = 0..m-1$ do 6–13
6. For $j = 0..15$ do $W_j \leftarrow x_{16i+j}$.
7. For $j = 16..79$ do
8. $W_j \leftarrow (W_{j-3} \oplus W_{j-8} \oplus W_{j-14} \oplus W_{j-16}) \lll 1$.
9. $(A,B,C,D,E) \leftarrow (H_1, H_2, H_3, H_4, H_5)$.
10. For $j = 0..79$ do 11–12
11. $t \leftarrow A \lll 5 + f_j(B,C,D) + E + W_j + K_j$.
12. $(A,B,C,D,E) \leftarrow (t, A, B \lll 30, C, D)$.
13. $(H_1, \boldsymbol{H_2}, \boldsymbol{H_3}, H_4, H_5) \leftarrow$
$(H_1 + A, H_2 + B, H_3 + C, H_4 + D, H_5 + E)$.
14. Antworte $H_1|H_2|H_3|H_4|H_5$.

In the verification we have:

	bits	①	②	③
the message	m			
the public key	a			
the signature	b, δ	6144	3328	512

Back to the scheme

26.5.09 ①

Verification:

Input: a public key $a_1 \in G$

a message $m \in \{0,1\}^*$

a signature $b, \delta \in G \times \mathbb{Z}_e$.

1. Return ($a^{b^*} b^{\delta} = g^{hash(m)}$ in G)

where $G = \langle g \rangle$, and $g = e$.

Setup: Input: security parameter

Output: G a group,

$g \in G$ element,

$e = ord(g)$.

User-setup: Input: G, g, e.

Output: $a \in G$, $\alpha \in \mathbb{Z}_e$

Choose $\alpha \in_R \mathbb{Z}_e$.

$a := g^{\alpha}$

Return (a, α)

How to 'solve' the verification equation for (b, γ) ?

...

Most things are already powers of g, with exponents known to the signer... not yet b but we can choose it as a power:

Signing

Input: a message m
a secret key α
global setup G, g, ℓ.

1. Choose $\beta \in_R \mathbb{Z}_\ell$, and compute $b := g^\beta$
2. Find $\gamma \in \mathbb{Z}_\ell$ as the (a) solution of
$$\alpha b^* + \beta\gamma = \text{hash}(m) \text{ in } \mathbb{Z}_\ell$$
for γ.
3. Return (b, γ).

CORRECT?

Then of course $a^{b^*} b^\gamma = g^{\alpha b^* + \beta\gamma} = g^{\text{hash}(m)}$,
ie. it is a signature!

EFFICIENT?

Ex

SECURITY?

Let's play a little...

The attacker tries to solve the verification equation w/o the secret key.

First try:

1. Choose any msg m.
2. Choose any b, if you want as $b = g^{\beta}$.
3. Compute the dlog

$$b^{\gamma} = g^{hash(m)} a^{-b^{*}} \quad \vdots \quad 1$$

3'. Compute the dlog $a = b^{\alpha}$ $\vdots$ 1

and then ...

Second try

1. Choose m.
2. Choose γ.
3. Solve

$$a^{b^{*}} b^{\gamma} = g^{hash(m)} \quad \vdots \quad 1 \quad ?$$

Last try

Last try

cp6
26.5.09
(4)

1. Force hash = id.

2. Then there are
nice formula
in same parameters

$$a$$
$$b = f(\ldots)$$
$$\gamma = g(\ldots)$$
$$m = h(\ldots)$$

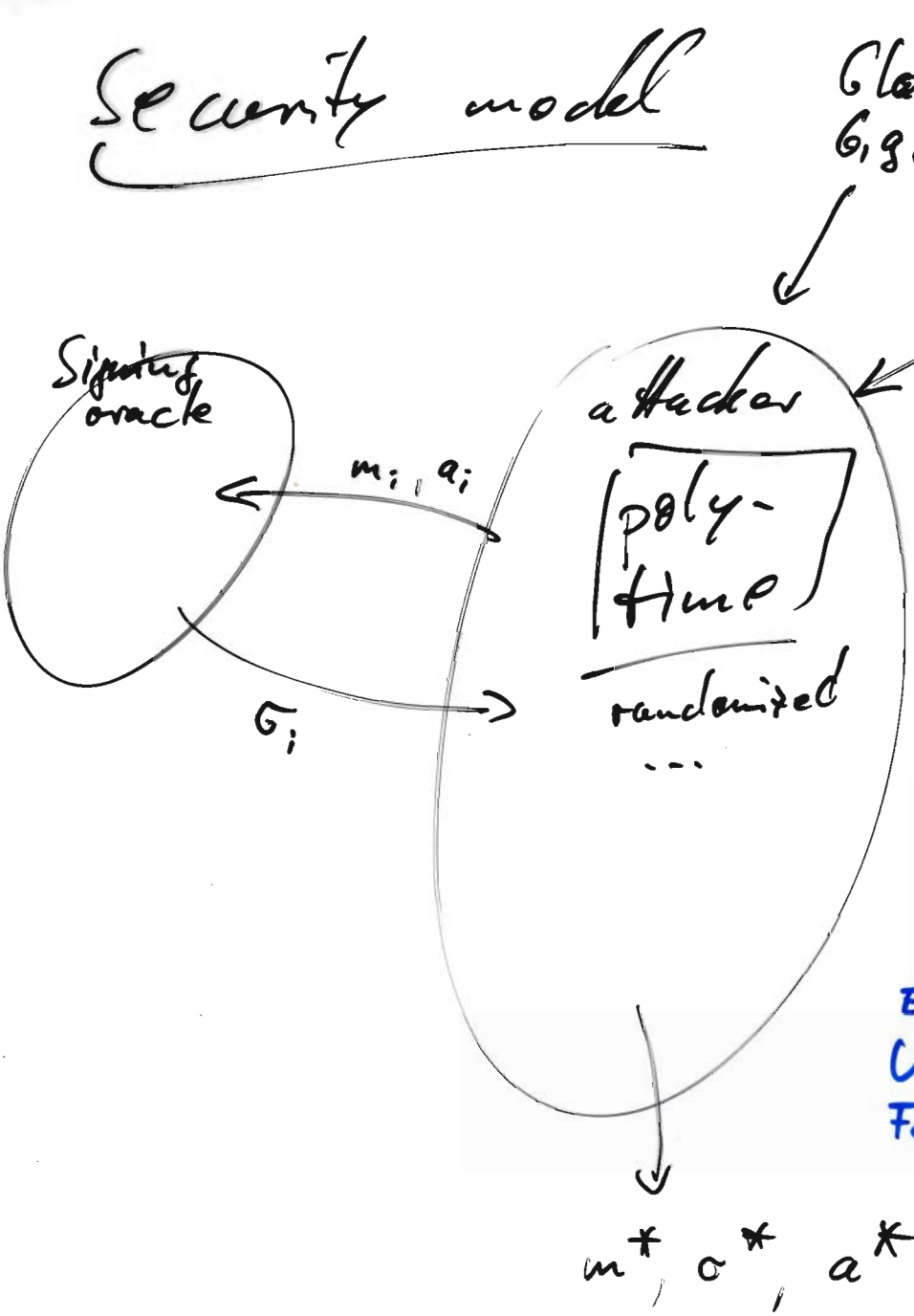
This construct an EXISTENTIAL FORGERY.

However, this attacker has
no way to influence
which message is signed.

All this is digging in the fog...

Security model

CpB 26.5.09 (5)



EF

EUF – CMA

Existential Universal Forgery

Chosen Message Attacks

Aim of the attacker:

$$\text{ver}_{a^*}(m^*, \sigma^*) = \text{TRUE}$$

& (m^*, a^*) never queried.

The attacker wins if his success probability is significantly larger than guessing.

SECURITY GOAL: No such attacker exists

ECDSA Setup:

- Implement an elliptic curve E (so we can add and subtract points, multiply points with an integer) and choose a point P of prime order q. (In fact, we work in the q-element group $\langle P \rangle$ generated by P.) We assume that q is a, say, 160-bit prime.
- Fix a cryptographic hash function HASH. The DSS chooses SHA-1 which produces a 160-bit value.
- Alice chooses a secret key $\alpha \in_{\mathrm{R}} \mathbb{Z}_q$ and publishes her public key $A = \alpha \cdot P \in E$.

Algorithm. ECDSA sign.

Input: Global parameters: the base point P, Alice' public key A, her secret key α and a message $m \in \{0,1\}^*$.

Output: A signature $(b^*, c) \in \mathbb{Z}_q$.

1. Calculate $e = \mathrm{HASH}(m) \in \mathbb{Z}_q$.
2. Repeat 3–5
3. Choose $\beta \in_{\mathrm{R}} \mathbb{Z}_q^\times$ at random.
4. Calculate $\beta \cdot P = (x_1, y_1)$ in the elliptic curve E and let $b^* = x_1 \operatorname{rem} q$.
5. Solve $\beta c = e + \alpha b^*$ in $\mathbb{Z}_q$ for $c \in \mathbb{Z}_q$.
6. Until $b^* \neq 0$ and $c \neq 0$
7. Return (b^*, c).

ALGORITHM. ECDSA verify.

Input: Global parameters: the base point P, Alice' public key A, the message $m \in \{0,1\}^*$ and the signature $(b^*, c) \in \mathbb{Z}_q \times \mathbb{Z}_q$.

Output: A boolean value stating whether the signature is valid.

1. Verify that $b^*, c \in \mathbb{Z}_q$. Else Return FALSE.
2. Calculate $e = \mathrm{HASH}(m) \in \mathbb{Z}_q$.
3. Calculate $(x_1, y_1) = c^{-1}e \cdot P + c^{-1}b^* \cdot A \in E$.
4. If $x_1 \operatorname{rem} q \neq b^*$ then Return FALSE
5. Else Return TRUE

If both parties are honest then

$$\begin{aligned} c^{-1}e \cdot P + c^{-1}b^* \cdot A &= c^{-1}(e + b^*\alpha) \cdot P \\ &= c^{-1}\beta c \cdot P = \beta \cdot P \end{aligned}$$

and so the x-coordinate of this point is the same that was used to define b^* and the signature is valid.

SECURITY REDUCTION

Prove a statement like:

If Problem X is difficult

then our scheme is EUF-CMA-secure.

We won't do anything like this here...

CONSEQUENCES?

Statements like

If our scheme is ... -secure

then ... the hash-function used has property A.

Thm If ElGamal type signatures for $(G, g, q, *, \text{hash})$ are secure

then the DLP for $g \in G$ must be difficult.

Proof Construct an attacker!

Eg. 1. Compute α from a by the DLP-subroutine

2. Generate signature...

□

Thm If ElGamal type signatures for $(G, g, \ell, *, hash)$ are secure then the hash function hash is collision-resistant.

Def A hash function family $\left(\{0,1\}^* \to \{0,1\}^k\right)_k$ is collision-resistant if it is difficult to find

$$x, y \in \{0,1\}^*,$$
$$x \neq y,$$
$$hash(x) = hash(y).$$

[difficult ~~=~~ not difficult = poly time and significant success prob.]

Proof Assume that the attacker has an efficient subroutine that often can produce collisions for the hash function.

1. Call this subroutine and get $m_1 \neq m_2$, $hash(m_1) = hash(m_2)$.
2. Call the signing oracle on m_1 and obtain σ_1.
3. Output (m_2, σ_1).

... □

Thm If ElGamal type signatures for $(G, g, \ell, *, \text{hash})$ are secure then the hash function is one-way.

one-way = difficult to find preimages

Proof (Ex) □

ElGamal type → DSA

Change verification so that only b^* is needed!

We had $a^{b^*} \boxed{b}^{\gamma} = g^{\text{hash}(m)}$

Solve for this position

$$b = b^1 = b^{\gamma\gamma^{-1}} = \left(g^{\text{hash}(m)}\, a^{-b^*}\right)^{\gamma^{-1}}$$ in the exponent

Thus

$$b^* = \left[\left(g^{\text{hash}(m)}\, a^{-b^*} \right)^{\gamma^{-1}} \right]^*$$

Use this as verification! → Save size in signatures.

MRTDs as described/required by ICAO

cpb
18.5.09
(1)

1. based on RFID technology

→ consequence that one can communicate with the chip from distances of up to 1.5m (or 10m with special equipment)

Problem: data on the chip is sensible and must not be handed out

(a) unless the reader physical access to the MRTD

machine readable travel document.

and

(b) unless the reader is authorized to read.

<p5
10.6.09
(2)

side remarks:

RFID = radio frequency identification

two variants:
- without power supply or battery → long lifetime & short distance
- with battery → short lifetime & long distance (~100 m)

Basic Access Control

→ MUST use the information written in the machine readable zone (MRZ) to produce some keying material

→ all further communication MUST be secured with "secure messaging" ie. encryption & authentication

Passive Authentication

→ chip has to verify a certificate given by the reader. This certificate is a signed electronic document containing
- access rights

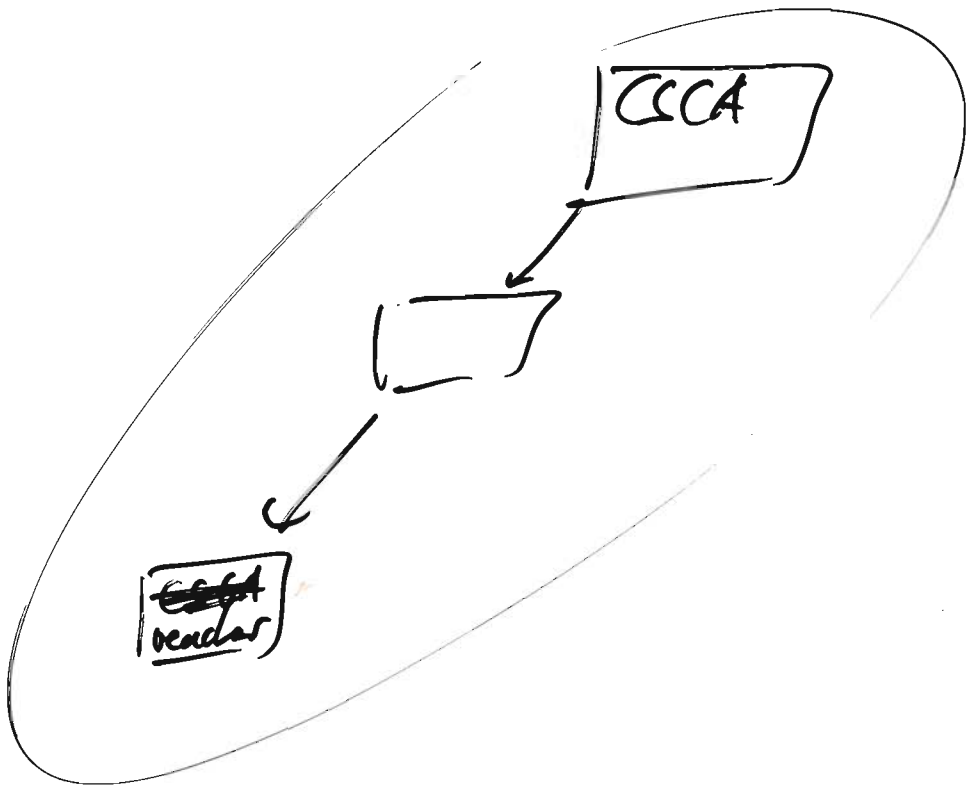
cpb
10.6.09
(3)

- id information
- public key
- signature of a higher authority

The chip has the

- Country Signing CA certificate
  contents similar info.
  public key: 3072 bit RSA key, or
  256 bit EC key

For complete verification the chip needs the certificate chain up to the country signing CA.



- Document Signer CA certificate with CSCA signature
  (2048 bit RSA key or 224 bit EC key)

cpb
10.5.09
(4)

Active Authentication

→ visual access to MRZ in order to do the basic access control.

Extended Access Control (OPTIONAL)

→ to protect information like finger prints which is not found in the non-electronic part of the MRTD.

→ biometrics may be additionally encrypted! [g. Diffie-Hellman]

BAC

(i) inspection of the MRZ

MRZ_information ← Document Number || Date_of_birth || Date_of_expiry

(ii) derive keys from this

(iii) authentication & key establishment

(iv) ~~the~~ after that use the established keys for secure messaging

Annex E Basic Access Control and Secure Messaging

E.1 Key Derivation Mechanism

The computation of 2 key 3DES keys from a key seed (K_{seed}) is used in both the establishment of the Document Basic Access Keys (K_{ENC} and K_{MAC}) and the establishment of the Session keys for Secure Messaging.

A 32 bit counter c is used to allow for deriving multiple keys from a single seed. Depending on whether a key is used for encryption or MAC computation the following values MUST be used:

- c = 1 (i.e. '0x 00 00 00 01') for encryption.
- c = 2 (i.e. '0x 00 00 00 02') for MAC computation.

The following steps are performed to derive 2 key 3DES keys from the seed K_{seed} and c:

1. Let D be the concatenation of K_{seed} and c ($D = K_{seed} \| c$).
2. Calculate H = SHA-1(D) the SHA-1 hash of D.
3. Bytes 1..8 of H form key K_a and bytes 9..16 of H form key K_b.
4. Adjust the parity bits of keys K_a and K_b to form correct DES keys.

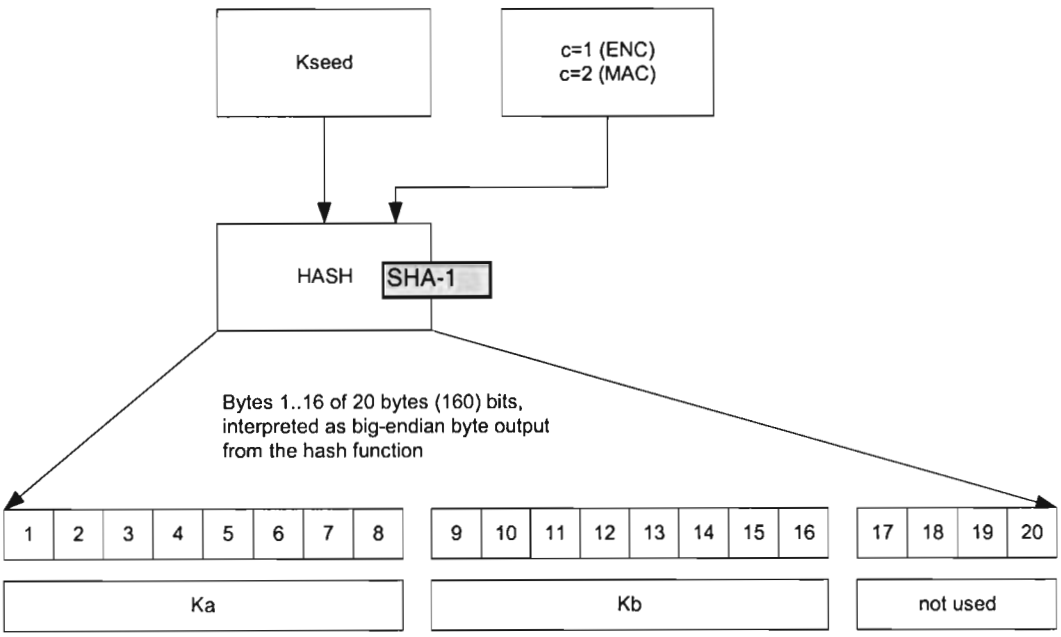


Figure 1: Compute keys from key seed scheme

E.2 Authentication and Key Establishment

Authentication and Key Establishment is provided by a three pass challenge-response protocol according to ISO 11770-2 Key Establishment Mechanism 6 using 3DES as block cipher. A cryptographic checksum according to ISO/IEC 9797-1 MAC Algorithm 3 is calculated over and appended to the ciphertexts. The modes of operation described in Annex E.4 MUST be used. Exchanged nonces MUST be of size 8 bytes, exchanged keying material MUST be of size 16 bytes. Distinguishing identifiers MUST NOT be used.

In more detail, IFD and ICC perform the following steps:

Key generation

Produces based on Kseed
(which is derived from the NRZ)
a 112 bit ~~key~~ 3 DES for
encryption
and a 112 bit key 3DES for
authentication

Observe:

3DES is used.

Known (at the time):

DES is broke.

- by differential or linear cryptanalysis but only theoretically.
- by brute force:

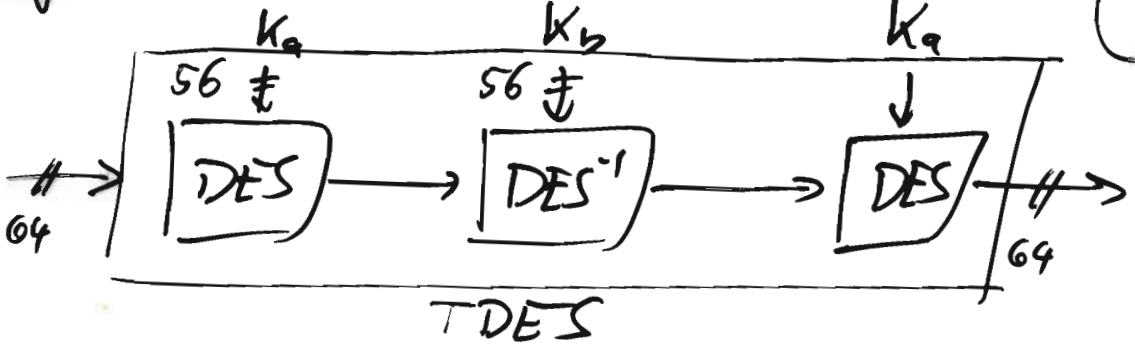
  EFF using a $250'000 special (?2001?) hardware in runtime 2 days.

  Uni Bochum/Kiel using $10'000 standard (?2004/5?) hardware in runtime 9 days

But: 3DES uses 112 bit key
and is supposed to be secure.
It is slower than 128-bit AES

cpb
10.6.09
(7)

3DES is just triple execution of DES:



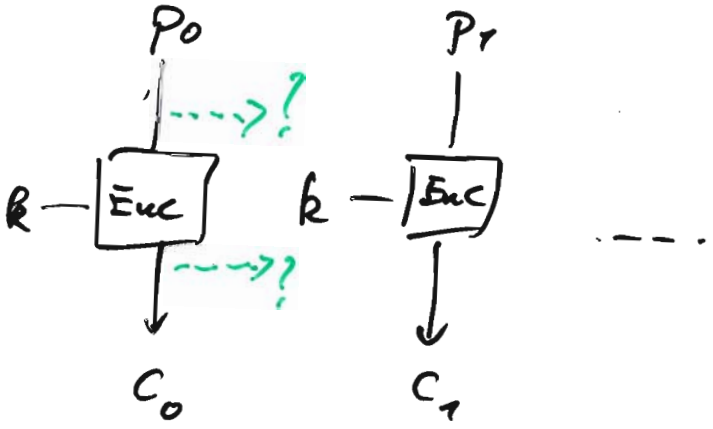
16.6.09
(1)

Block cipher into stream cipher
(like TDES, or AES)

Modes of operation

Easiest solution:

ECB Electronic codebook



+ Fast self-synchronization
- Simple to modify a single block
- Simple to get plain text - cipher text pairs
- Same plain text $\rightarrow$ same cipher text

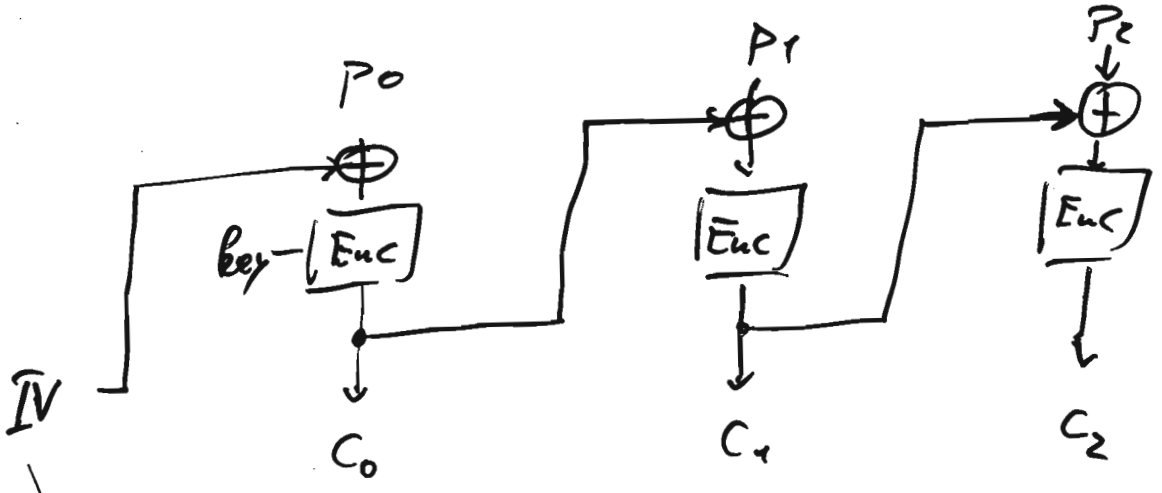
cp5
16.5.09
(2)

Ways out:

(a) reuse bits from p_0 or c_0 in the encryption of p_1 (and so on ...)

(b) use some MAC (message authentication code = kind of a signature) to detect changes

CBC-mode (Cipher Block Chaining)



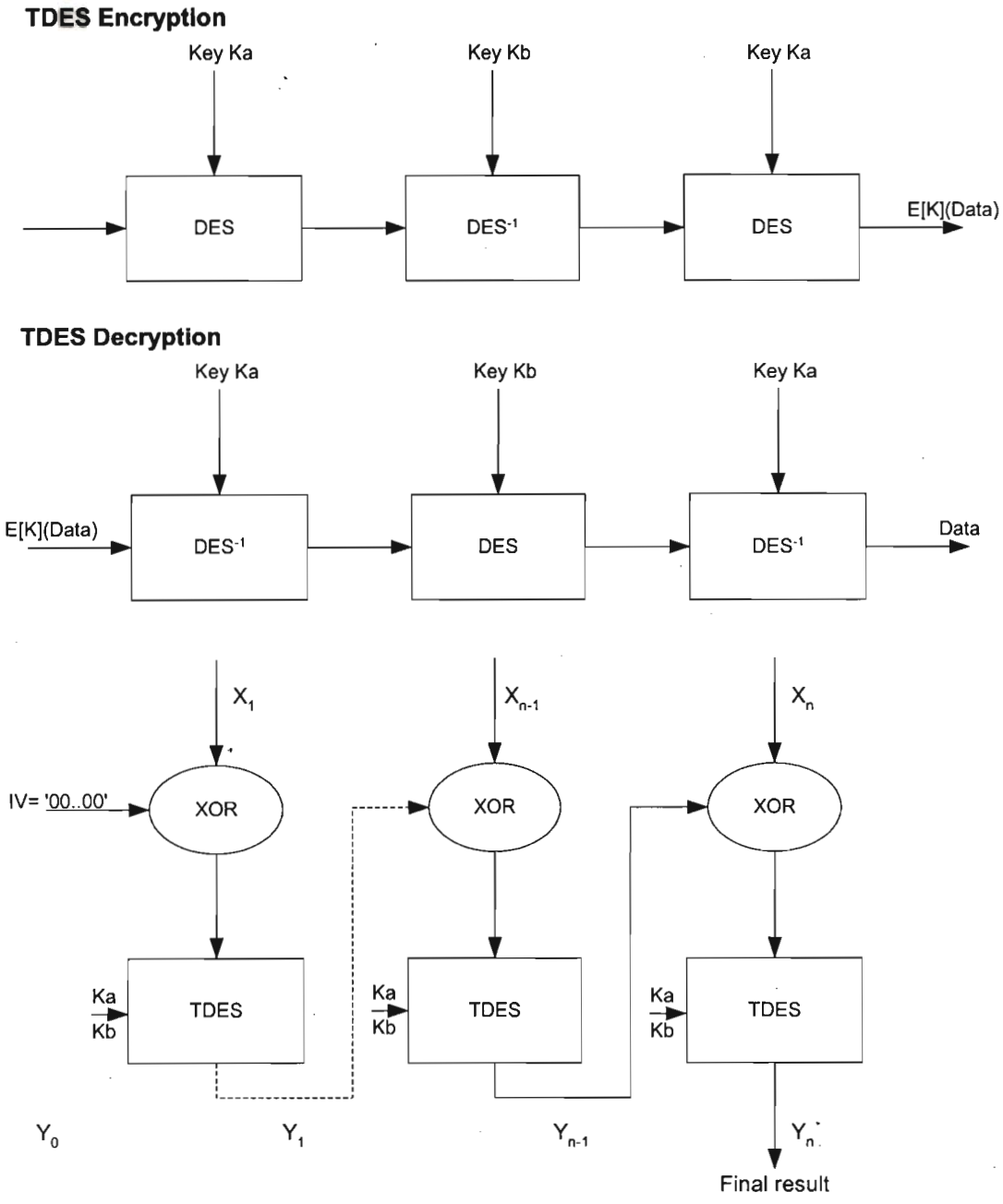
IV
can be agreed on
or transmitted & chosen at random
or fixed

+ same plaintext yields different ciphertext if we vary the IV
+ fast self-synchronization
± simple to get plaintext-ciphertext-pairs

E.4 3DES Modes of Operation

E.4.1 Encryption

Two key 3DES in CBC mode with zero IV (i.e. 0x00 00 00 00 00 00 00 00) according to ISO 11568-2 is used (see diagrams below). No padding for the input data is used when performing the MUTUAL AUTHENTICATE command. During the computation of SM APDUs, padding according to ISO 9797-1 padding method 2 is used.



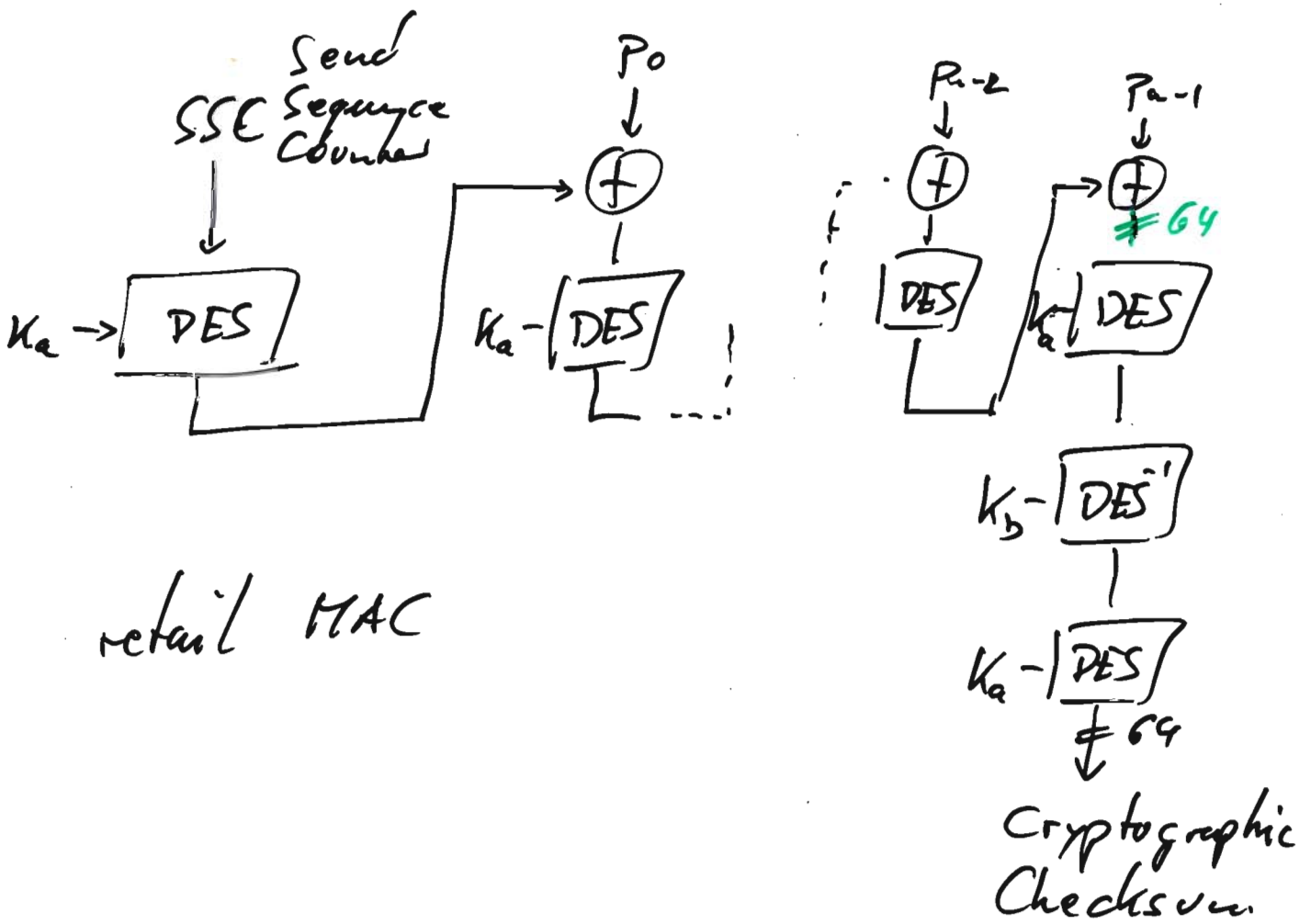
IV	=	zero initialization vector
'X_1||....||X_n'	=	plain text (message to encrypt) where each block X_i is 64-bit long
'Y_1||....||Y_n'	=	resulting cryptogram (encrypted message) where each block Y_i is 64-bit long

Figure 4: 3DES Encryption/Decryption in CBC Mode

And what about integrity?

cpb
10.6.09
(3)

Specially DES-TDES adapted CBC-like MAC:



retail MAC

→ compare to other constructions

XCBC-AES or HMAC-SHA1

...

Security: We should prove something like:

if we can forge a cryptographic checksum then we can break DES or even TDES.

E.4.2 Message Authentication

Cryptographic checksums are calculated using ISO/IEC 9797-1 MAC algorithm 3 with block cipher DES, zero IV (8 bytes), and ISO9797-1 padding method 2. The MAC length MUST be 8 bytes.

After a successful authentication the datagram to be MACed MUST be prepended by the Send Sequence Counter. The Send Sequence Counter is computed by concatenating the four least significant bytes of RND.ICC and RND.IFD respectively:

SSC = RND.ICC (4 least significant bytes) || RND.IFD (4 least significant bytes).

The Send Sequence Counter is increased every time before a MAC is calculated, i.e. if the starting value is x, in the next command the value of SSC is x+1. The value of the first response is then x+2.

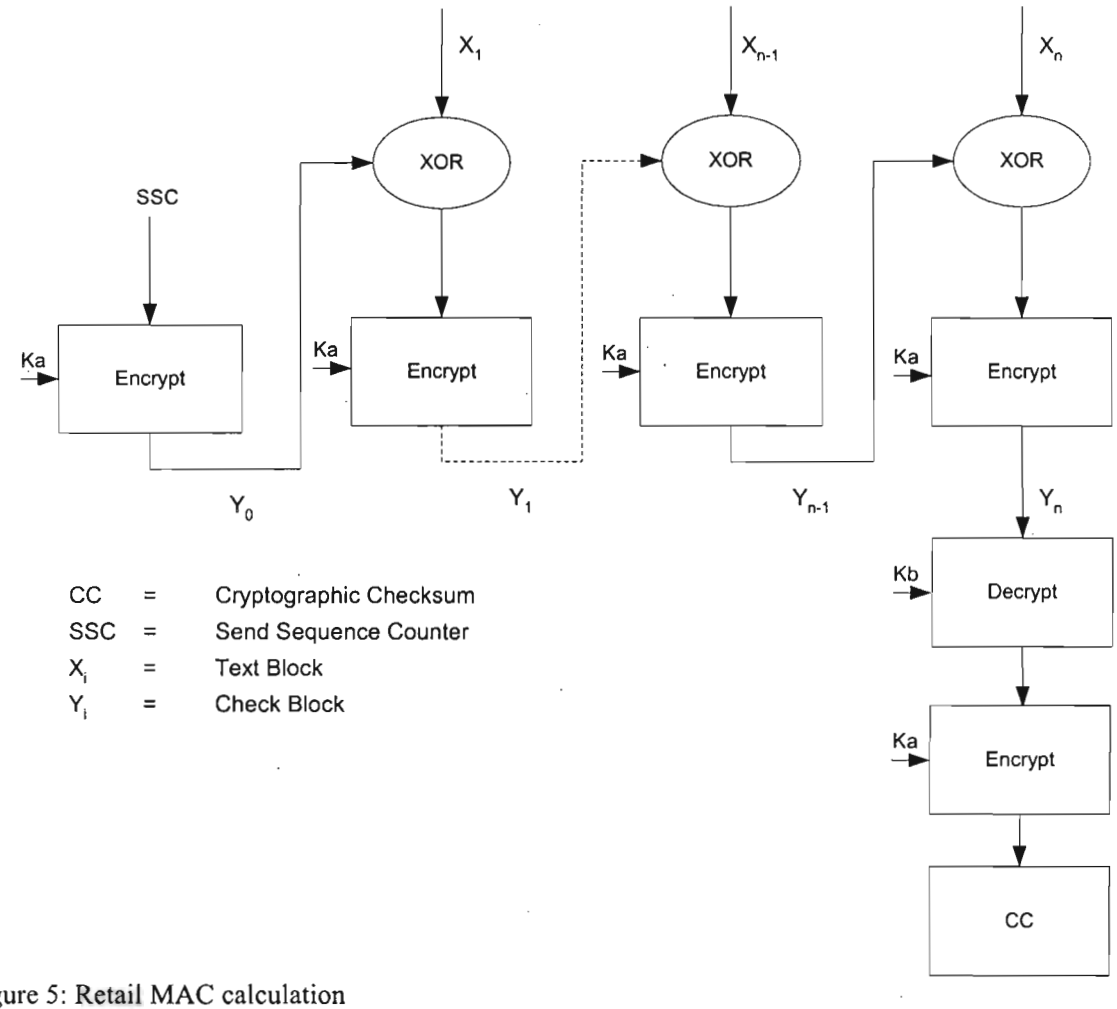
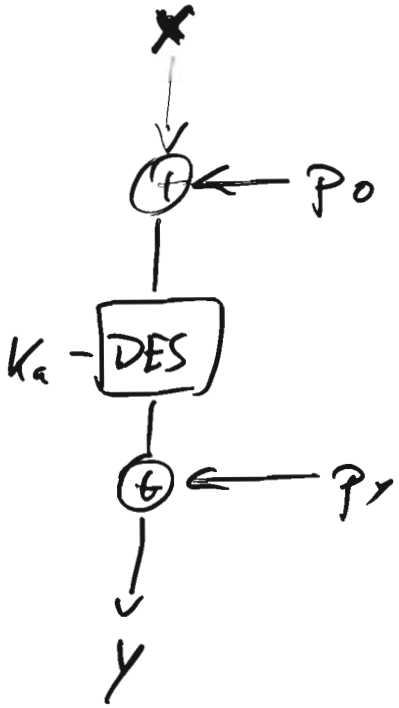


Figure 5: Retail MAC calculation

cpb
10.6.09
(4)

We should consider the effect of changes to the message.



Changing without effect on y?

$$DES_{K_a}(p_0 \oplus x) \oplus p_1$$

$$\| $$

$$DES_{K_a}(p_0' \oplus x) \oplus p_1'$$

ie. we would have to manipulate p_1' like an encryption.

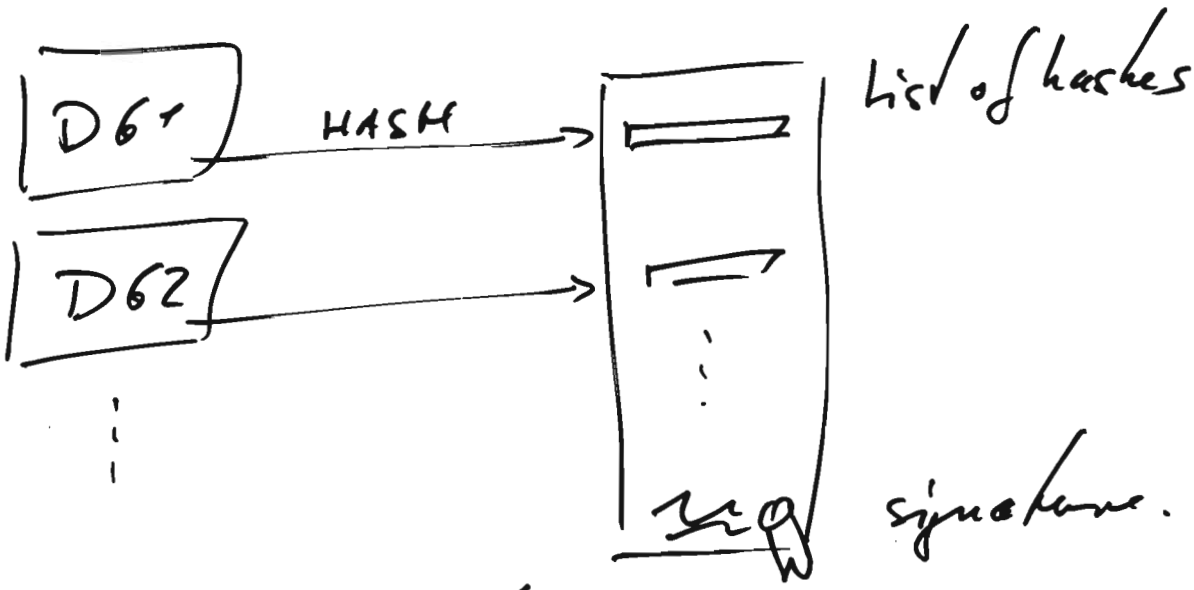
But we cannot encrypt without the key.

So it seems that we need to break at least DES to forge the crypto-checksum.

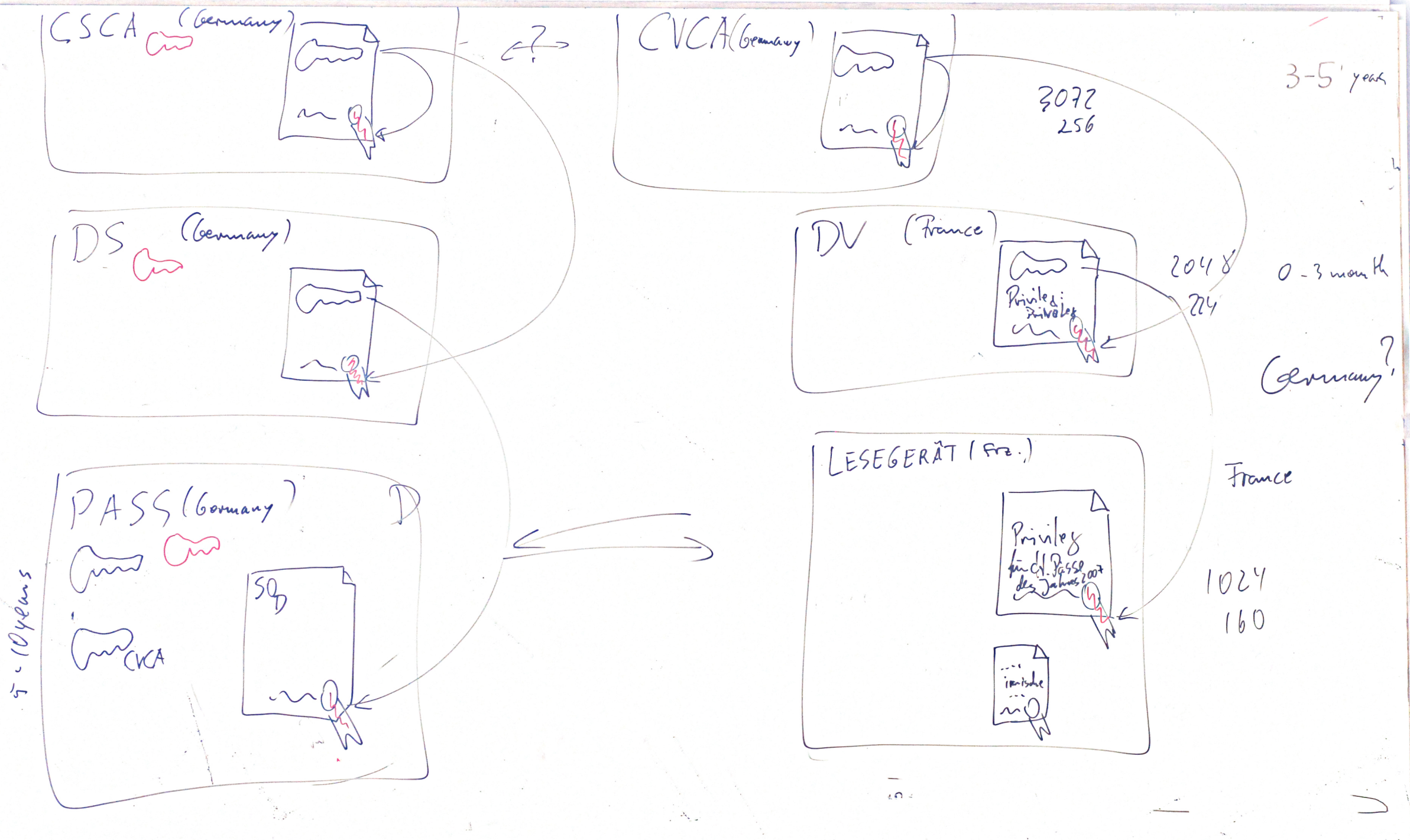
Logical Data Structure

cp6
17.6.09
(4)

- Precise definitions down to byte level are given in Doc 9303 about which and how the data are stored.
- All this data has to be 'secured' by a signature. The construction is as follows:



This grants that the reader can check integrity and authenticity of each single Data Group separately.


CSCA (Germany)
CVCA (Germany)
3072
256
3-5 years
DS (Germany)
DV (France)
Privileg
2048
224
0-3 month
Germany?
LESEGERÄT (Frz.)
France
PASS (Germany)
SOD
CVCA
5-10 years
Privileg
1024
160

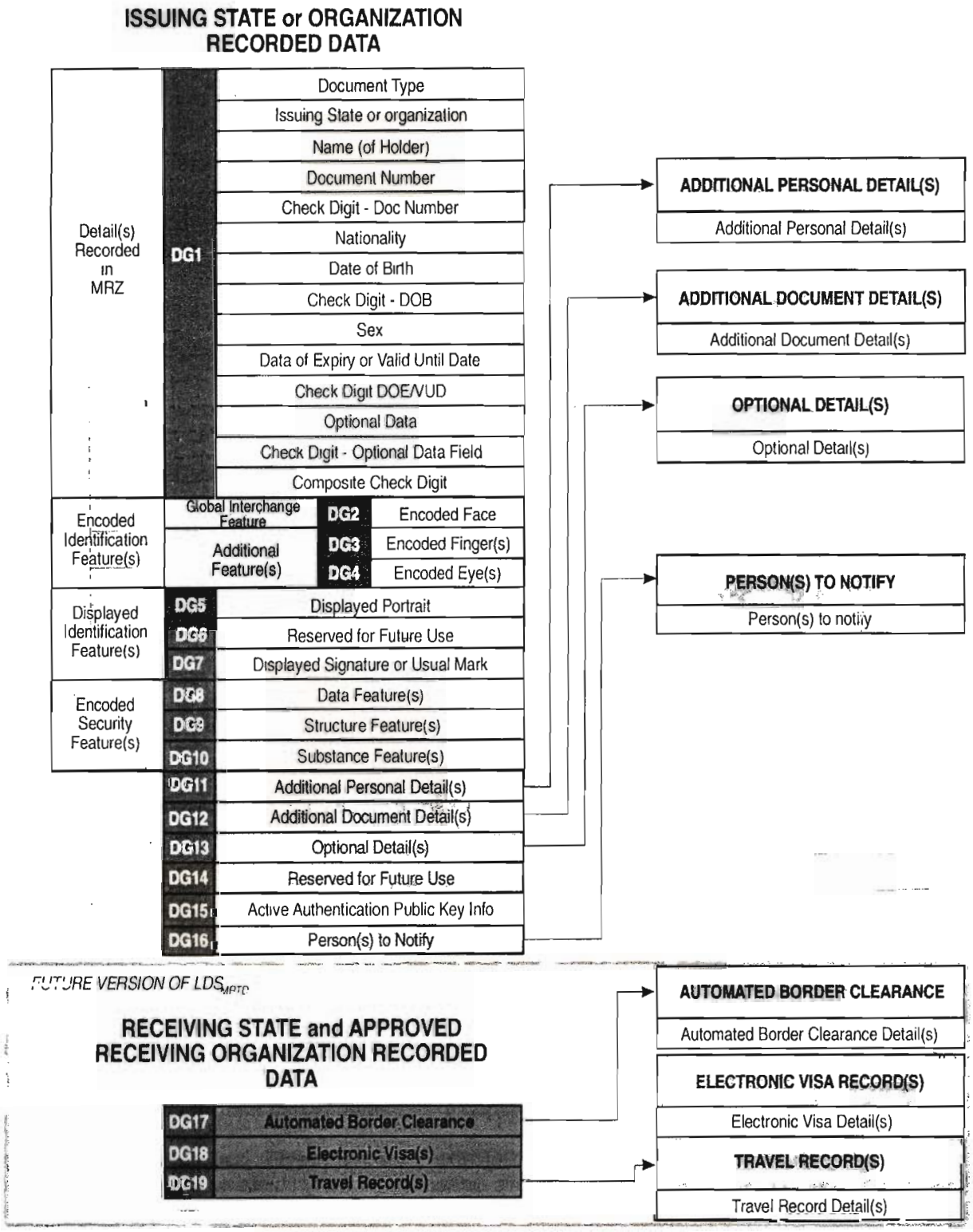


Figure III-2. Data group reference numbers assigned to LDS

Why ICAO Selected the Face as Primary Biometric Identifier specified to ePassports

by ICAO Secretariat

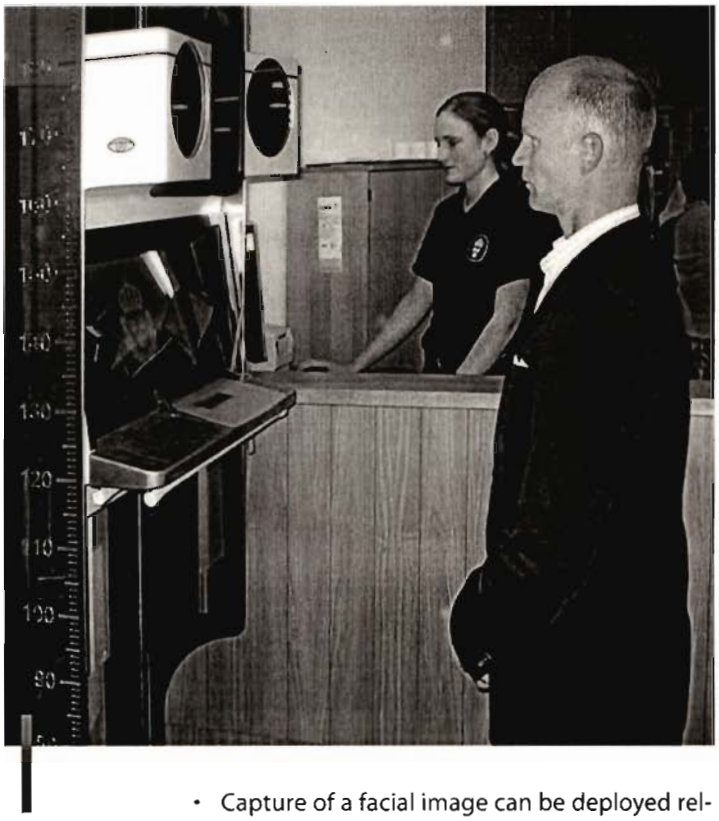
It has long been recognized that names and honour are not sufficient to guarantee that the holder of an identity document (such as a Machine Readable Passport - MRP) assigned to that person by the issuing State is guaranteed to be the person purporting, at a receiving State, to be the same person to whom that document was issued.

The only method of relating a person irrevocably to his travel document is to have a physiological characteristic of that person associated with the travel document in a tamper-proof manner. This physiological characteristic is a biometric.

After a five-year investigation into the operational needs for a biometric identifier which combines suitability for use in the MRP issuance procedure and in the various processes in cross-border travel consistent with the privacy laws of various States, ICAO has specified that facial recognition shall become the globally interoperable biometric technology. A State may also optionally elect to use fingerprint and/or iris recognition in support of facial recognition.

In reaching this conclusion, ICAO observed that for the majority of States the following advantages applied to facial images:

- Facial photographs do not disclose information that the person does not routinely disclose to the general public.
- The photograph (facial image) is already socially and culturally accepted internationally.
- The facial image is already collected and verified routinely as part of the MRP application form process in order to produce a passport to Doc 9303 standards.
- The public is already aware of the capture of a facial image and its use for identity verification purposes.
- The capture of a facial image is non-intrusive. The end user does not have to touch or interact with any physical device for a substantial timeframe to be enrolled.
- Facial image capture does not require new and costly enrollment procedures to be introduced.



- Capture of a facial image can be deployed relatively immediately, and the opportunity to capture facial images retrospectively is also available.

- Many States have a legacy database of facial images captured as part of the digitized production of passport photographs which can be encoded into facial templates and verified for identity comparison purposes.

- In appropriate circumstances, as decided by the issuing State, a facial image can be captured from an endorsed photograph, not requiring the person to be physically present.

- For watch lists, a photograph of the face is generally the only biometric available for comparison.

- Human verification of the biometric against the photograph/person is relatively simple and a familiar process for border control authorities.

Optional additional biometrics

States can optionally provide additional data input to their (and other States) identity verification processes by including multiple biometrics in travel documents, i.e. a combination of face and/or fingerprint and/or iris. This is especially relevant where States may have existing fingerprint or iris databases in place against which they can verify the biometrics proffered to them; for example, as part of an ID card system.

Storage of an optional fingerprint biometric

There are three classes of fingerprint biometric technology: finger image-based systems, finger minutiae-based systems, and finger pattern-based systems. Whilst standards have been developed within these classes to make most systems interoperable amongst their class, they are not interoperable between classes. Three standards for fingerprint interoperability are therefore emerging: storage of the image data, storage of the minutiae data and storage of the pattern data. Where an issuing State elects to provide fingerprint data in its ePassport, the storage of the fingerprint image is mandatory to permit global interoperability between the classes. The storage of an associated template is optional at the discretion of the issuing State.

Storage of an optional iris biometric

Iris biometrics are complicated by the dearth of proven vendors. A de facto standard for iris biometrics has therefore emerged based on the methodology of the one recognized vendor. Other vendors may in future provide iris technology, but it is likely they will need the image of the iris as their starting point, rather than the template created by the current vendor. Where an issuing State elects to provide iris data in its ePassport, the storage of the iris image is mandatory to permit global interoperability. The storage of an associated template is optional at the discretion of the issuing State.

For more on this issue, please see ICAO Doc 9303 Part 1, Volume 2 sixth edition. ◆

Chip authentication

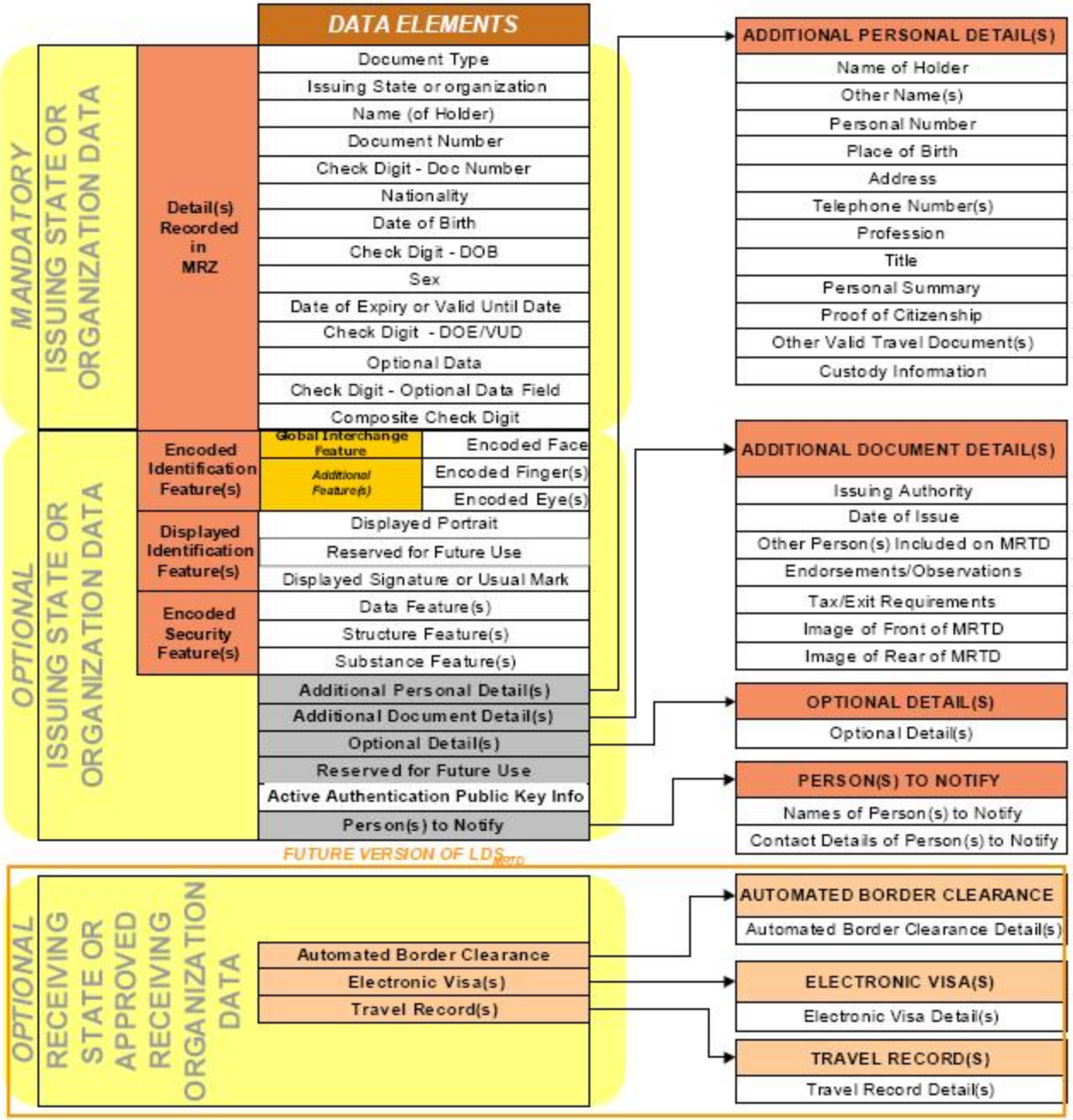
30.6.09
cp6
(1)

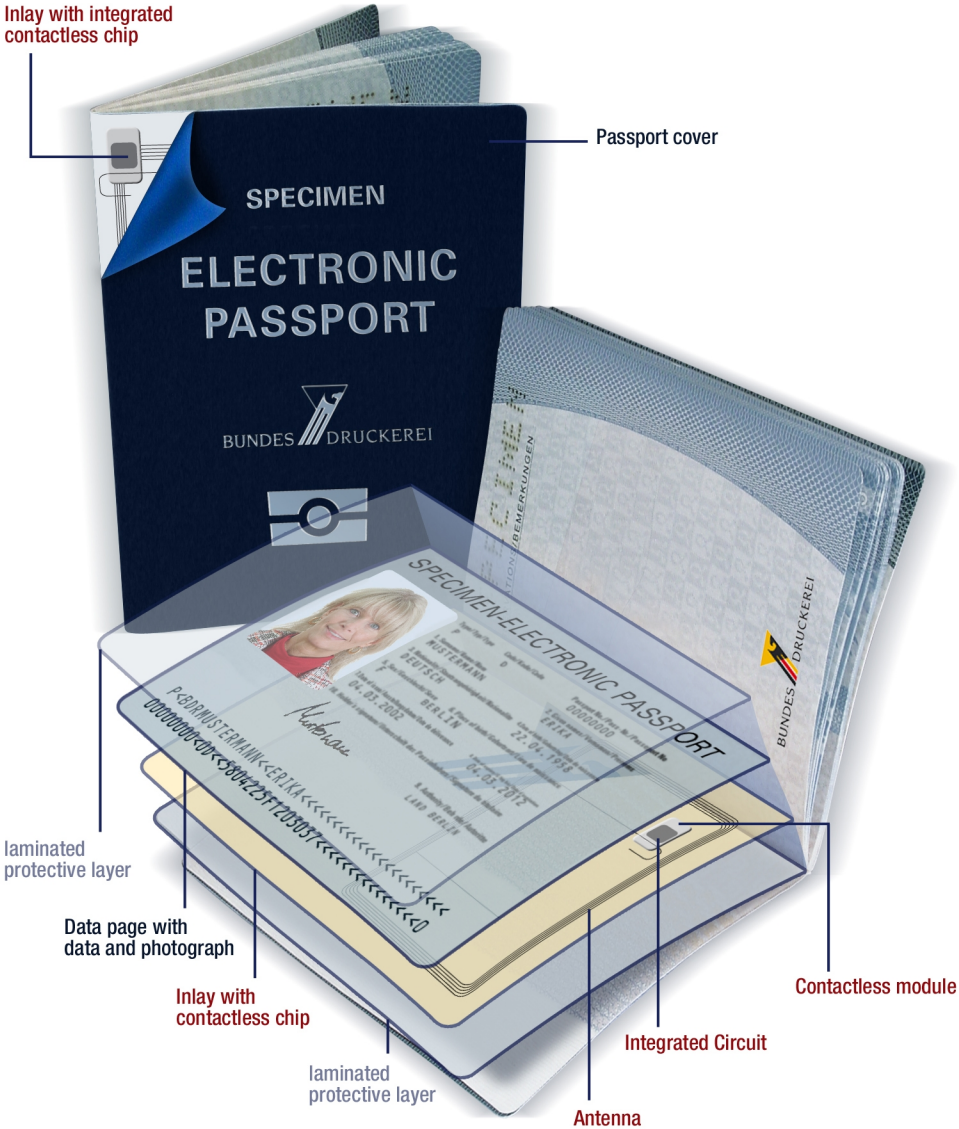
- Chip authentication
  (replaces Active Authentication)
- Terminal authentication.
-

Advanced Inspection Procedure

1. Basic Access Control (now mandatory) — MUST
2. Chip authentication (incl. key update (via DH!)) — MUST
3. Passive authentication (read the security object and check its signature) — MUST
4. Read less-sensitive data — OPT.
5. Terminal authentication — OPT.
6. Read sensitive data as far as the reader's certificate allows — OPT.

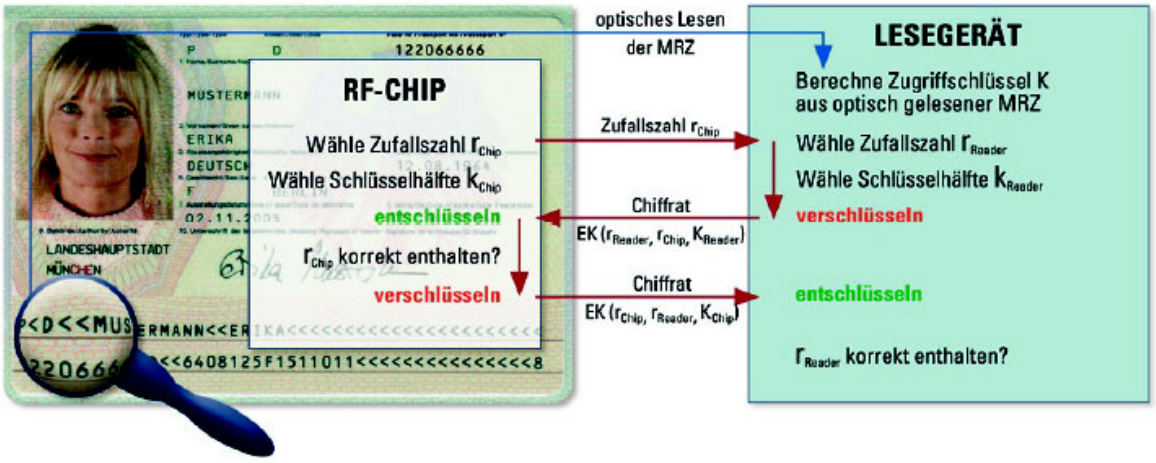
(based on TR by BSI ≥ 2004 / 2009)


DATA ELEMENTS
MANDATORY
ISSUING STATE OR ORGANIZATION DATA
Detail(s) Recorded in MRZ
Document Type
Issuing State or organization
Name (of Holder)
Document Number
Check Digit - Doc Number
Nationality
Date of Birth
Check Digit - DOB
Sex
Date of Expiry or Valid Until Date
Check Digit - DOE/VUD
Optional Data
Check Digit - Optional Data Field
Composite Check Digit
OPTIONAL
ISSUING STATE OR ORGANIZATION DATA
Encoded Identification Feature(s)
Global Interchange Feature
Additional Feature(s)
Encoded Face
Encoded Finger(s)
Encoded Eye(s)
Displayed Identification Feature(s)
Displayed Portrait
Reserved for Future Use
Displayed Signature or Usual Mark
Encoded Security Feature(s)
Data Feature(s)
Structure Feature(s)
Substance Feature(s)
Additional Personal Detail(s)
Additional Document Detail(s)
Optional Detail(s)
Reserved for Future Use
Active Authentication Public Key Info
Person(s) to Notify
FUTURE VERSION OF LDS
OPTIONAL
RECEIVING STATE OR APPROVED RECEIVING ORGANIZATION DATA
Automated Border Clearance
Electronic Visa(s)
Travel Record(s)
ADDITIONAL PERSONAL DETAIL(S)
Name of Holder
Other Name(s)
Personal Number
Place of Birth
Address
Telephone Number(s)
Profession
Title
Personal Summary
Proof of Citizenship
Other Valid Travel Document(s)
Custody Information
ADDITIONAL DOCUMENT DETAIL(S)
Issuing Authority
Date of Issue
Other Person(s) Included on MRTD
Endorsements/Observations
Tax/Exit Requirements
Image of Front of MRTD
Image of Rear of MRTD
OPTIONAL DETAIL(S)
Optional Detail(s)
PERSON(S) TO NOTIFY
Names of Person(s) to Notify
Contact Details of Person(s) to Notify
AUTOMATED BORDER CLEARANCE
Automated Border Clearance Detail(s)
ELECTRONIC VISA(S)
Electronic Visa Detail(s)
TRAVEL RECORD(S)
Travel Record Detail(s)



Source-Bundesdruckerei GmbH

▶ The contactless chip can be integrated into either the cover page or the data page.


P
D
122066666
MUSTERMANN
ERIKA
DEUTSCH
F
02.11.2005
LANDESHAUPTSTADT
MÜNCHEN
P<D<<MUSTERMANN<<ERIKA<<<<<<<<<<<<<<<<<<<<
122066666<<<6408125F1511011<<<<<<<<<<<<<<8
RF-CHIP
Wähle Zufallszahl r_{Chip}
Wähle Schlüsselhälfte k_{Chip}
entschlüsseln
r_{Chip} korrekt enthalten?
verschlüsseln
optisches Lesen der MRZ
Zufallszahl r_{Chip}
Chiffrat
EK (r_{Reader}, r_{Chip}, K_{Reader})
Chiffrat
EK (r_{Chip}, r_{Reader}, K_{Chip})
LESEGERÄT
Berechne Zugriffschlüssel K aus optisch gelesener MRZ
Wähle Zufallszahl r_{Reader}
Wähle Schlüsselhälfte k_{Reader}
verschlüsseln
entschlüsseln
r_{Reader} korrekt enthalten?

Response APDU	Mandatory if data is returned, otherwise absent.	Not used	Mandatory, only absent if SM error occurs.	Mandatory if DO‘87’ and/or DO‘99’ is present.

Table 1: Usage of SM Data Objects

Figure 2 shows the transformation of an unprotected command APDU to a protected command APDU in the case *Data* and *Le* are available. If no *Data* is available, leave building DO ’87’ out. If *Le* is not available, leave building DO ‘97’ out.

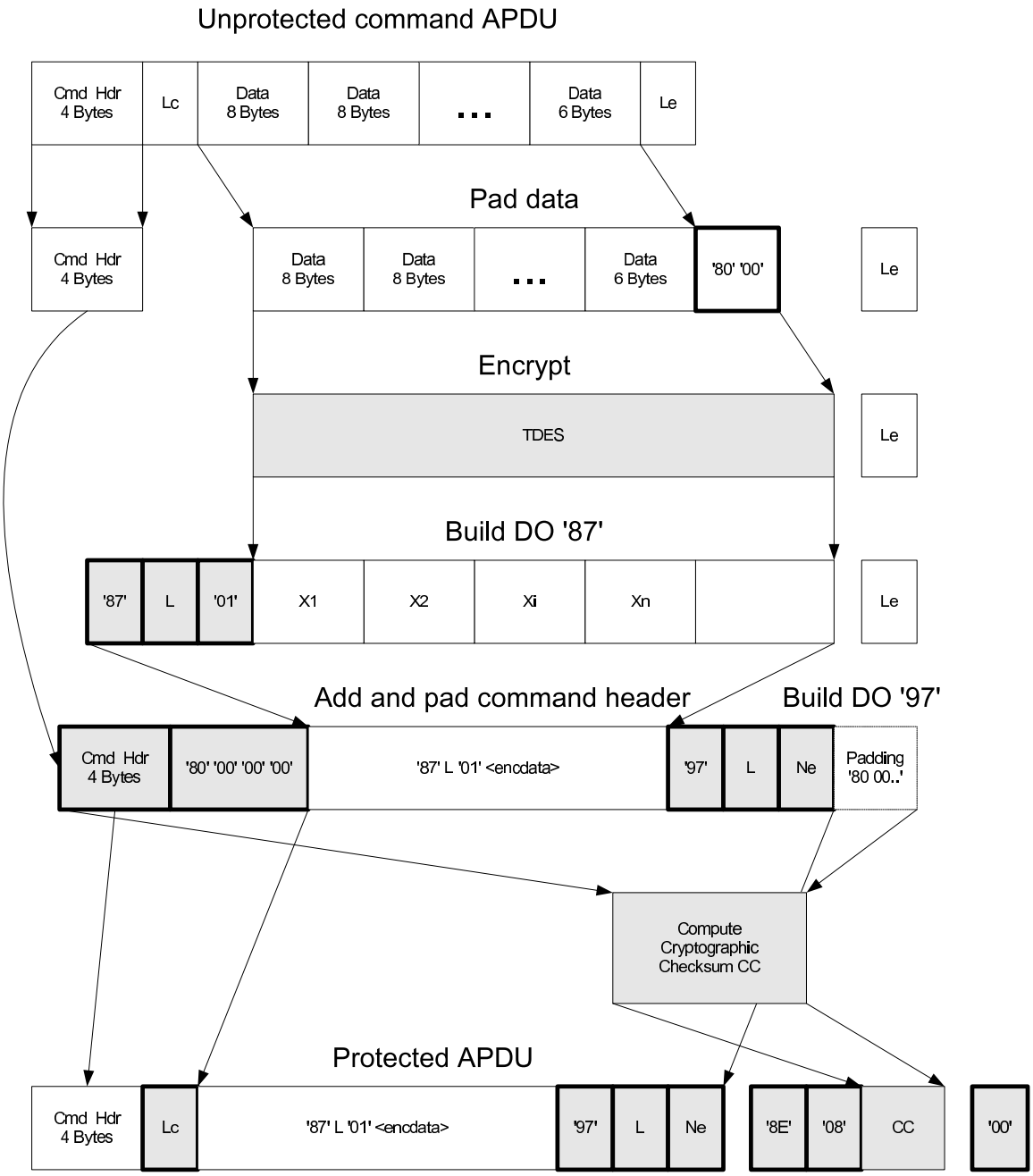
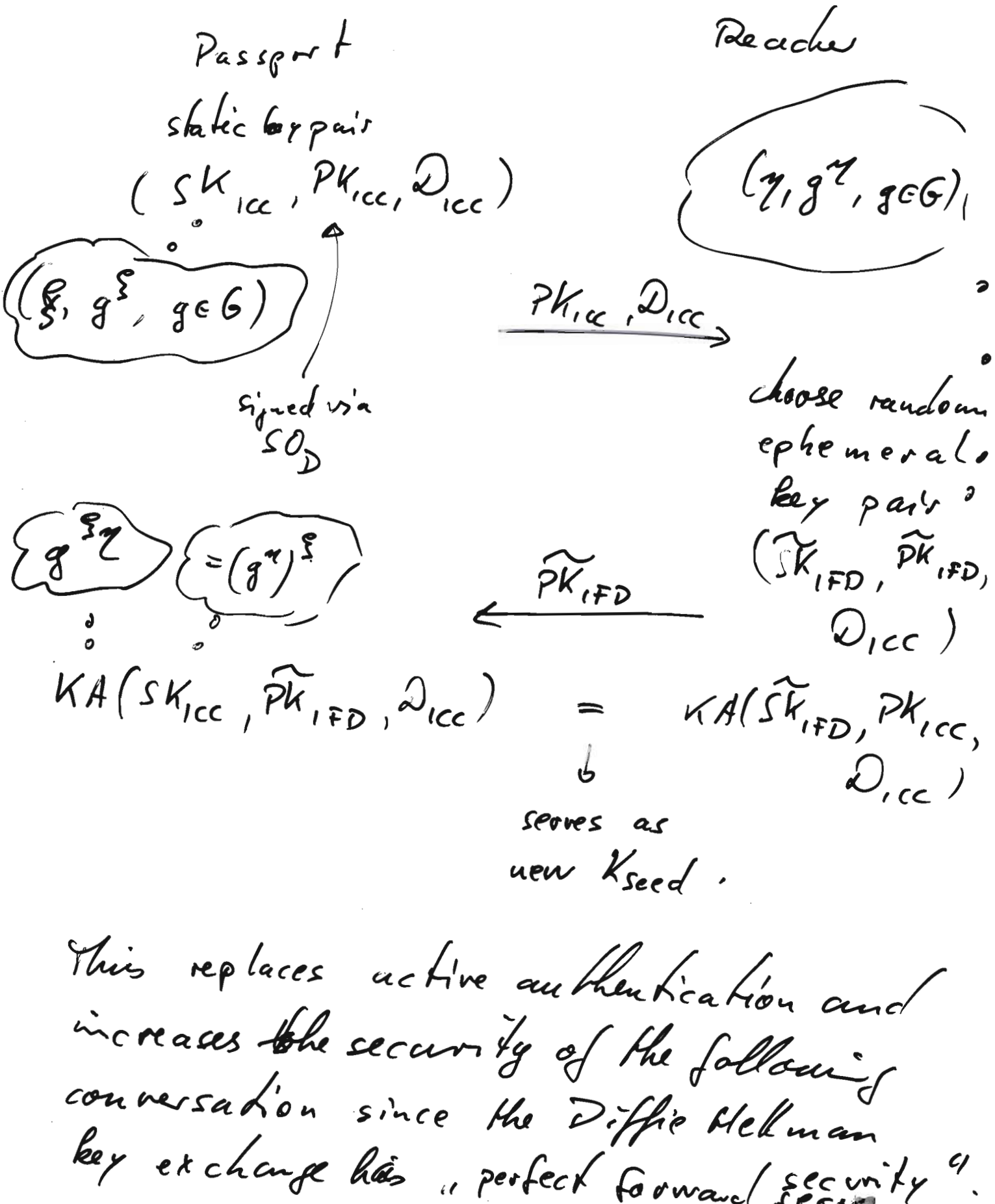


Figure 2: Computation of a SM command APDU

The protocols

Chip authentication



This replaces active authentication and increases the security of the following conversation since the Diffie Hellman key exchange has „perfect forward security".

Repetition:

cpb
1.7.09
①

HORTON's principle

We must always authenticate
the meaning of the message.

(ie. auth. the plaintext is fine,
but auth. the ciphertext is not sufficient.)

We can use either

AtE Authenticate then Encrypt.

or

E&A Encrypt & authenticate.

In the second case we have to be aware
that the authentication mechanism
gets the ciphertext, so we seem to violate
HORTON's principle.
We can authenticate ciphertext + enc.-key
determines the plaintext
or authenticate using ~~a key~~ an auth.-key
which is strongly related to the
encryption key.

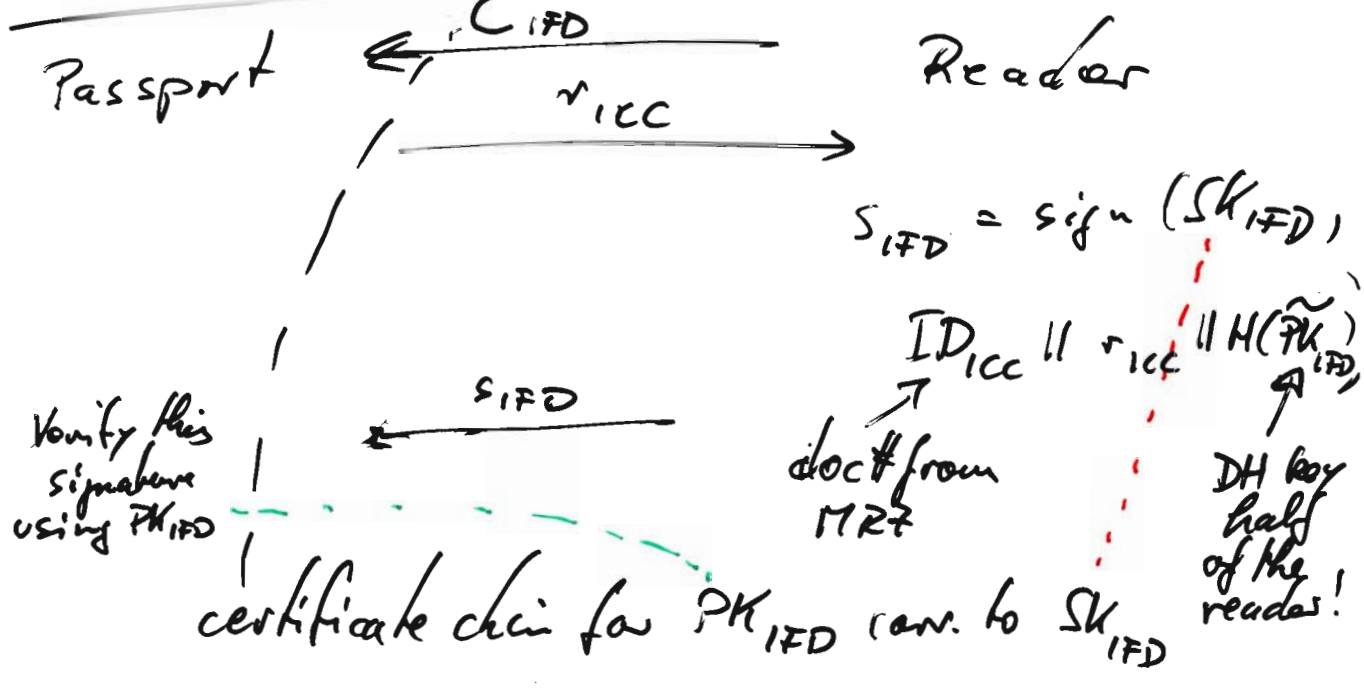
This happens in the MRTD protocols.

After setting up a secured connection via a certain key agreement protocol (for example Diffie-Hellman) we can bind authentication results to the agreed key. Thus you do need to reauthenticate every message!

The "authentication" in the chip authentication protocol is provided by subsequent verification of the SO_D, the passive authentication.

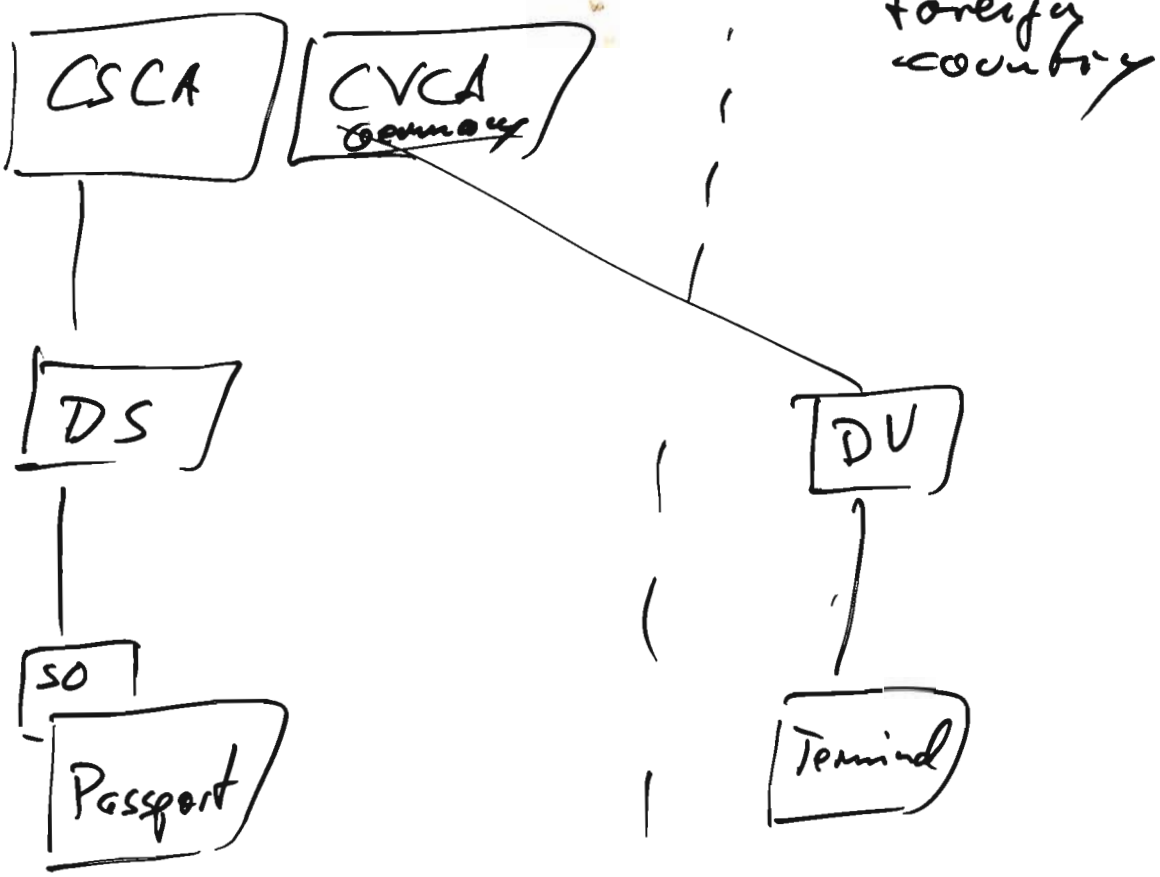
Last open point:

Terminal authentication



cpb
1.7.09
(3)

We have make sure that the chip knows the root certificate C_{CVCA} of the country verification Certification authority. So we store it on the chip during production.

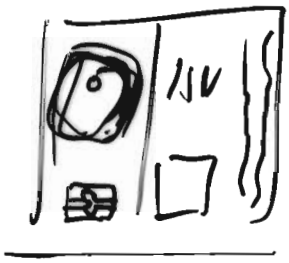


Mind sovereignity of each country!

(Eg. the CSCA and CVCA certificates are not distributed publicly but only by diplomatic means.)

cpb 1.7.08 (4)

Passport ⟷ Reader



(a) Nicely structured data

(b) various auth. & means:

BAC → grants that nobody can talk or listen without the MRZ

Passive auth. → chip info cannot be changed.

EAC : chip auth. (EDH!)
→ chip not cloned
→ perfect forward security

terminal auth
→ chip can check privileges of the reader.

Need two PKI, one for signing and one for verification.

cpb
1.7.09
(5)

Ask questions!

What prevents chip cloning?

Why is step (2) before step (3) in the Advanced Inspection Procedure?

What grants confidentiality of the conversation between chip and reader?

Can an attacker - with or without knowledge of MRZ - identify a certain passport, say to trigger a bomb?

cpl
7.7.09
①

Biometrics

focussed on fingerprint

Intentions?

→ Verify or recognize
the identity
of a living human individual.

→ automated methods.

based on physiological or
behavioral characteristics.

Examples of biometrics:

- finger prints
- iris scans
- voice
- face 2D, 3D
- gait
- hand writing
- hand geometry
- ear
- teeth
- (smell)
- DNA)
  - patterns
  - genome

History related fingerprints

cpb
7.7.09
(2)

≈ 6000 BC fingerprints on pottery,
on houses,
on business records.

1686 Marcello Malpighi
→ ridges, spirals, loops

1823 John Purkinji (Breslau)
→ nine fingerprint patterns

1880 Alphonse Bertillon (Paris)
→ system of anthropometry
to classify criminals
and to identify recidivists
→ height, weight, length of arm, leg, index finger

☺ ≈1904 case „William West"

1856 Sir William Herschel (Jungipoor, India)
→ used palmprints (and fingerprints)
to certify contracts
with native people
PLAYING on their superstition.

cpb 7.7.09 (3)

→ observed, as his collection grew, that fingerprints can prove or disprove identity.

1870 Dr Henry Faulds (British surgeon, Japan)
- noticed finger marks on ancient pottery
→ classification method.
→ first to identify greasy fingerprint on an alcohol bottle.

1880s Francis Galton
1892 book "Fingerprints"
→ fingerprints are permanent throughout life
→ no two are identical
(prob ~ 1 : 64 · 10^9)
• Galton pattern types
• minutiae, Galton details

1891 Juan Vucetich (Argentina, police officer)
→ first systematic filing of fingerprints using Galton pattern types
1892 ... identified a murder using a bloody fingerprint

1897 Sir Edward Henry (British police officer, India)

<pl 7.7.05 (4)

→ modified fingerprint classification based Galton's observations so that you can easily scan for various criteria ...

⟶ Henry classification

1901 ⟶ adopted by Scotland Yard.

1902 first : USA : NY Civil Services Comm.

1903 NY state prison system

} US military, many state & local law enforcement agencies

1920

1924 FBI establishes Identification Division by an act of Congress (based on Henry)

1971 > 200 mio fingerprints MANUALLY

Manual card files

cpb
7.7.09
(5)

Henry system
numerical weights to fingers
with a whorl pattern
→ bin number $\in \{0, \ldots, 1023\}$

Letter symbols assigned to fingers
→ subdivide bins

Problems . assigned pattern may vary on the same card

. pattern type error
→ wrong bin
→ no result

later, in early automation
FRR ≈ 25% estimated.
false reject rate

. further complication:
- distribution of pattern types is not uniform

loop 65%

whorl 30%

arch 5%

solution: subdivide large bins ...

Still: Henry (& Vucetich) system
allowed to search Large files
though the FRR might have
been high.

ad eg.: arches not further
subdivided.

Searching time was long

mid 1960s:

FBI Identification Division
processed ~25000 requests/day
in a criminal file of >20 mio fingerprints
using several thousand employees.
→ high FRR

US NBS/NIST supported by FBI
- automated digitization
of inked fingerprints
(invent scanners!)
- effect of image compression
- classification
- extract minutiae
- matching

Technology

qb
8.7.09
(1)

late 1960s

(*)
- 1.5 in x 1.5 in
- 500 dpi
- effective spotsize 0.0015 in (≙1000dp
- signal to noise ratio > 100:1
- ≥ 64 gray levels, 6-bit

early 1970s: 5 machines built,
250 fingerprints / h.
used til 1978
→ to digitize 22 mio fingerprints

early 1980s: "low cost" scanners

~ 1990 IAFIS
/
Integrated Automated Fingerprint Identification System
App-F: public specification of (*)
+ ≥ 200 effective gray levels

today:
- 1.5 x 1.5 in @ ≥ 1000 dpi
- 10 - 12 bits gray level
- S/N > 100:1
- ...

& directly from fingers "live scan"

Automated fingerprint identification

14.7.09
cpb
(1)

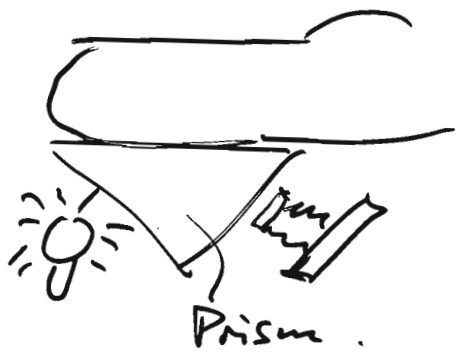
eg. IAFIS (FBI)

Principal method

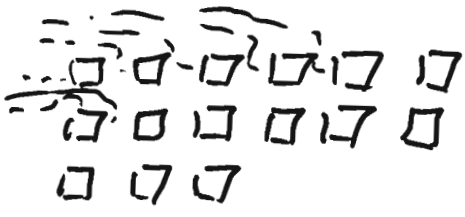
1. Finger print classification
2. Minutiae
3. • Pore positions along ridges
   • Patterns formed by the ridge edges

Sensors

- optical sensors



- capacitive silicon



- electric field sensors
- thermoelectric sweep sensor
  (iPAQ h5400 Pocket PC)
- ultrasonic sensor
- pressure array
- ---

14.7.09
cpb
(2)

Questions / criteria?

- cost
- robostness, resistance to aging
  durability
- power consumption
- size
- dot pitch
- behaviour under electro-static discharg-
- environmental / operation for sweaty and dry fingers
- usability, ...
  resistance against misuse (tiredness? ...)

Testing & identification — Algorithm?

→ FVC2002 (FingerprintVerification Competition)

4 databases { one synthetical
others from different sensors.

samplesize 90, average age ≈ 20

...

Sensor	Equal error rate	FNMR at FMR 1%	FNMR at FMR 0.1%
Optical	0,1%	0.11%	0.21%
Capacitive	0.37%	0.32%	0.61%

Research projects (of ≈2003)

14.7.09
(3)
qb

- Device interoperability
- Lightweight fingerprint verification
- Fingerprint watermarking (?)
- Secure devices
- Continuous classification systems
  → translate fingerprint to a vector so that 'close' means similar
- Pattern matching algorithms
- Fingerprint image mosaicing techniques
- Fingerprint video recognition

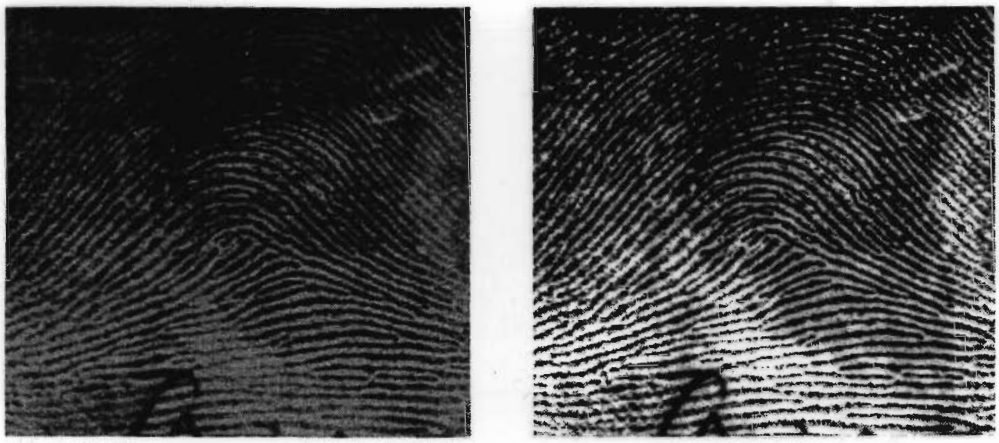
Outlook

→ PDAs
→ Laptop PCs !
→ mobile phones
→ secure mass storage
→ cars

cpb
15.7.09
(1)

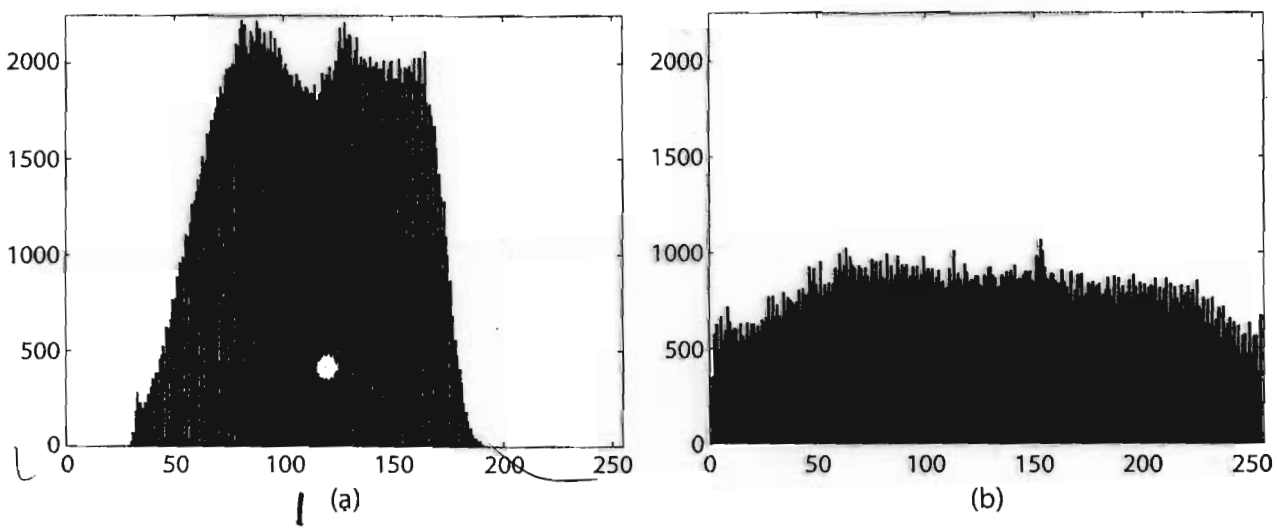
~~Local~~ Extracting information from fingerprint images

1. improve image



(a) (b)

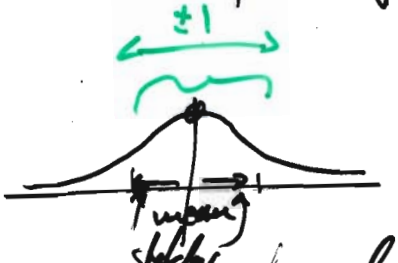
Figure 2.3 (a) An inked fingerprint image; (b) the results of the local area contrast enhancement algorithm on (a).



local area contrast enhancement (LACE)

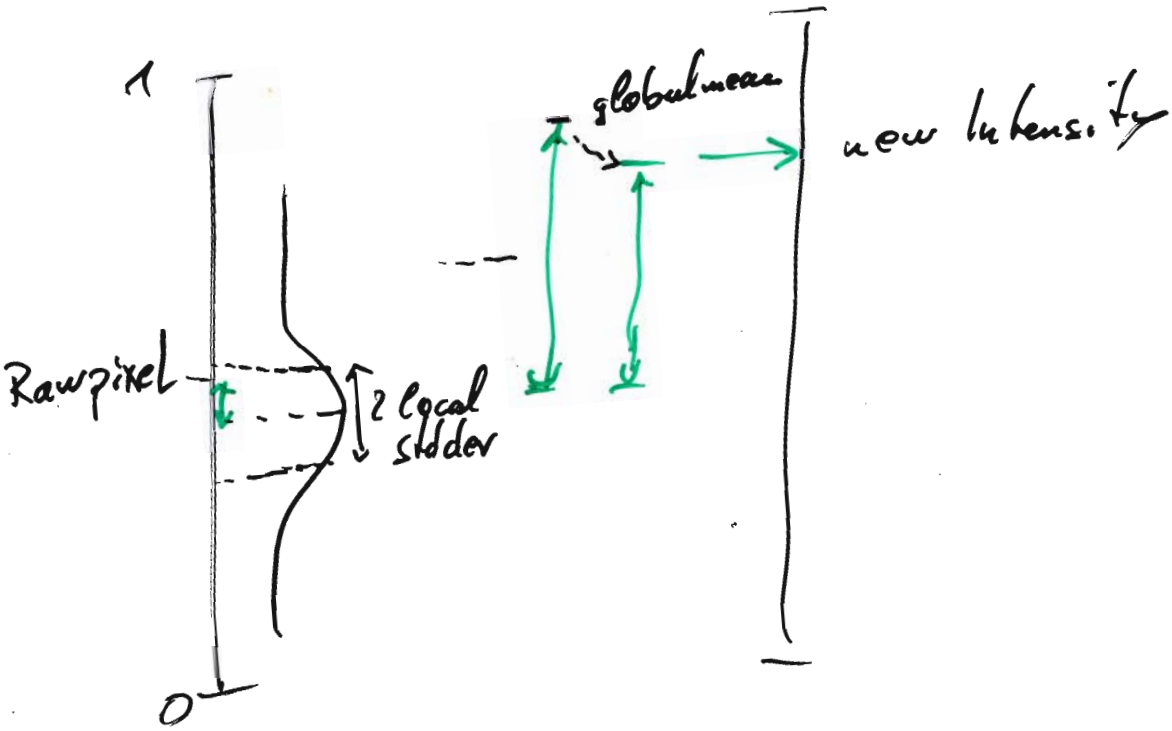
$$\text{pixelgain} = c \cdot \text{globalmean} \cdot \frac{1}{\text{local area stddev}}$$

↑ global correction, empirically determined, eg. $c = \frac{1}{2}$.



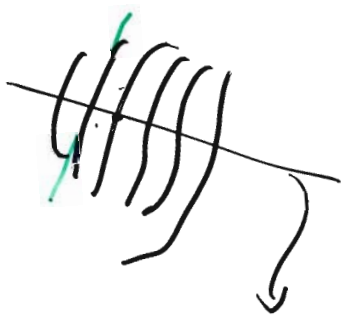
local area is a, say, 15 × 15 pixel window centered at the considered point

New Intensity
= Pixelgain (RawPixel − localmean)
+ local mean

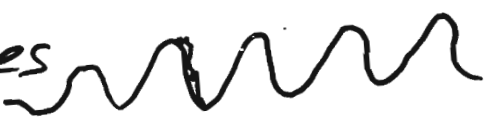


2. Contextual filtering

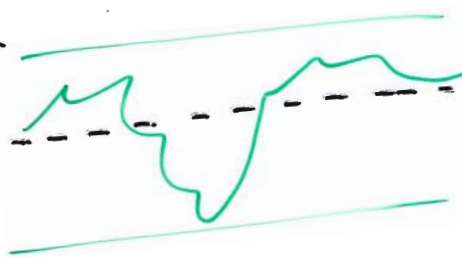
Amplify expected structure



todo : (1) provide band pass filter orthogonal to the ridges [highpass]



(2) provide low pass filter along the ridges



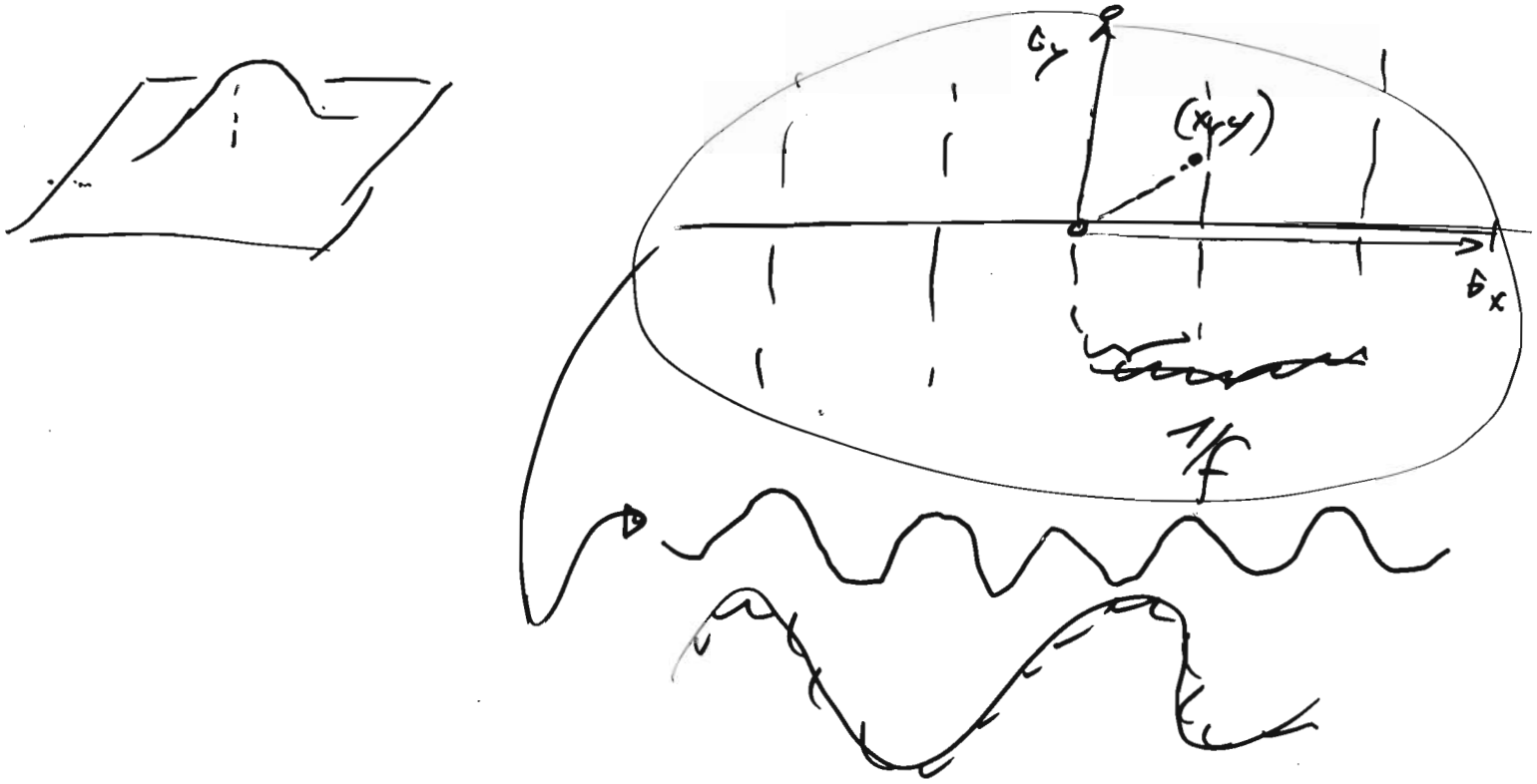
Gabor filter.

Curves :

qb
15.7.09
(3)

$$g(\underbrace{x, y}_{\text{distance to considered point}}; \underbrace{\sigma_x, \sigma_y}_{\text{damping in x and y direction}}, \underbrace{f}_{\text{frequency "of ridges"}})$$

$$= \exp\left(-\frac{1}{2}\left(\frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2}\right)\right) \cdot \cos(2\pi f \cdot x)$$



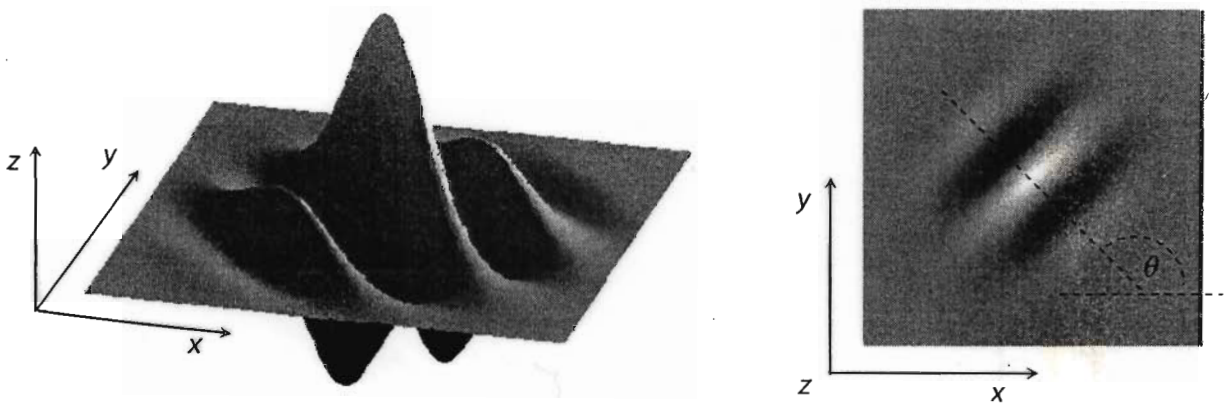


Figure 2.5 Graphical representation (lateral view and top view) of the Gabor filter defined by the parameters $\theta = 135°, f = 1/5, \sigma_x = \sigma_y = 3$ [21].

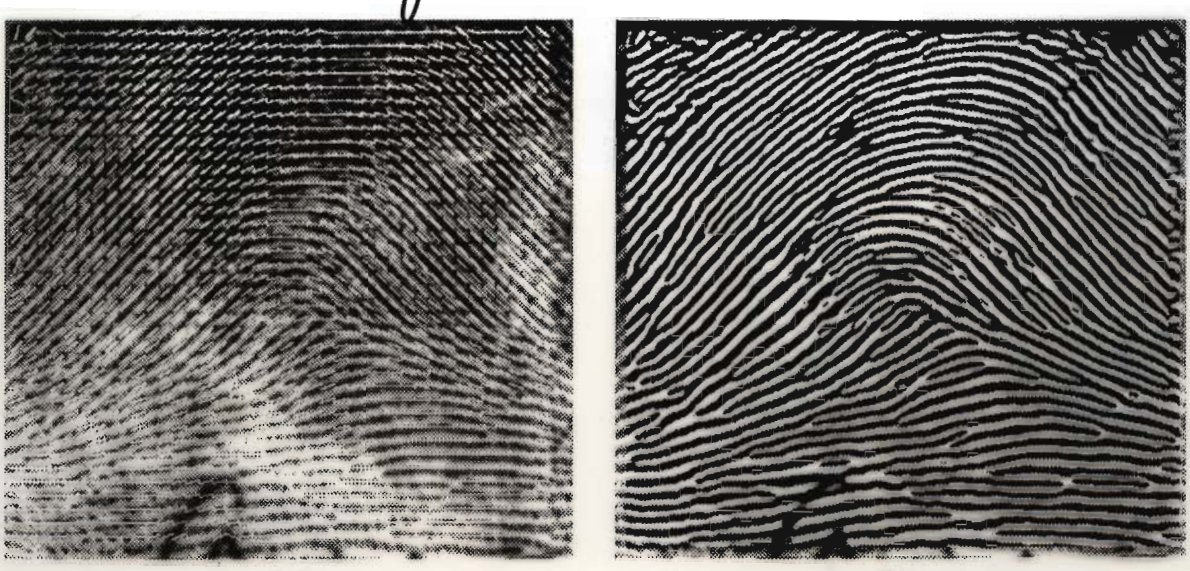
cpb
15.7.09
(4)

$$\text{New Pixel} = \sum_{a,b} g(a,b) \cdot \text{Pixel}(x+a, y+b)$$

$g(a,b)$: Filter function (maybe rotated!)

This does what we want:

- if the ridge frequency coincides with the filter frequency then the ridges emphasized. (Otherwise they are damped.)
- If the ridge direction is parallel to the filter direction then it will be emphasized.
- Being in a valley or on a ridge will be amplified.



(a) (b)

Figure 2.7 The orientation field is superimposed on the fingerprint image in (a). In (b) of the Gabor filters-based contextual filtering of the fingerprint image in Figure 2.3(a)

This nicely smoothes out the picture!
Cool!

However, we need ridge frequency
and ridge direction
for every pixel!

What we have is a function

$$(x,y) \longmapsto \text{gray level of pixel } x,y$$

on some integer raster.

First idea: use derivatives

$$\begin{bmatrix} \partial/\partial x\, I \\ \partial/\partial y\, I \end{bmatrix} = \text{gradient of the intensity}$$

↖ "shortest way to the maximum"

Refinements: $\text{angle} \approx \arctan\left(\frac{\partial I}{\partial y} \Big/ \frac{\partial I}{\partial x}\right)$

Problems: not stable enough.

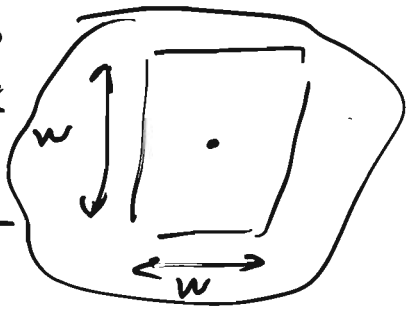
One uses:

$$\theta(x,y) = 90^\circ + \frac{1}{2}\arctan\left(\frac{2\,G_{xy}}{G_{xx}-G_{yy}}\right)$$

$$G_{xy} = \sum_{-\frac{w}{2}\le h,k\le\frac{w}{2}} \frac{\partial I}{\partial x}(x+h,y+k)\,\frac{\partial I}{\partial y}(x+h,y+k)$$

$$G_{xx} = \sum_{h,k}\left(\frac{\partial I}{\partial x}(x+h,y+k)\right)^2$$

$$G_{yy} = \sum_{h,k}\left(\frac{\partial I}{\partial y}(x+h,y+k)\right)^2$$



eg. $w = 15$.

and compute the derivatives via a Sobel filter

-1		1
-2		2
-1		1

This gives us the directions.
Still need the local ridge frequency f.

We still miss the ridge frequency!

cp6
27.7.9
(1)

Take a cross cut orthogonal to the determined ridge direction

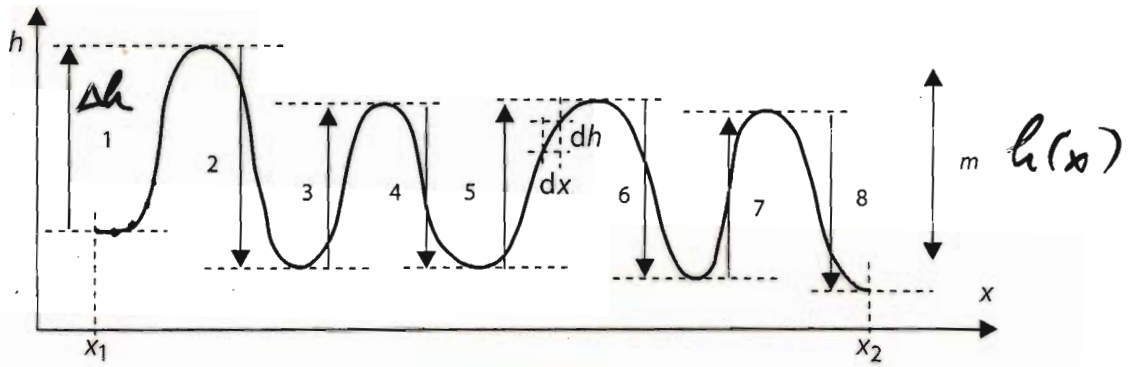


Figure 2.6 The variation of the function h in the interval $[x_1,x_2]$ is the sum of amplitudes $\alpha_1,\alpha_2,\ldots \alpha_8$ [24]. If the function is periodic or the function amplitude does not change significantly within the interval of interest, the average amplitude α_m can be used to approximate the individual α. Then the variation may be expressed as $2\alpha_m$ multiplied by the number of periods of the function over the interval [21].

We consider the total variation:

$$V(h) = \int_{x_0-k}^{x_0+k} |h'(x)|\, dx \qquad \text{(local view)}$$

$$\text{average amplitude} = \frac{\sum |(\Delta h)(x_i)|}{\#\, i}$$

$$V(h) = 2k \cdot 2\,\text{average amplitude} \cdot f \qquad \text{(global view)}$$

$$\Rightarrow f = \frac{\int_{x_0-k}^{x_0+k} |h'(x)|\, dx}{2k \cdot 2\,\text{average amplitude}}$$

In practice: average over a small neighborhood

Now we are ready to forget details. cpb 24.7.9 (2)

The next step is:

- choose a threshold and make the picture b/w (two colors!)
- thin out ridges til they are only one pixel wide.

Finally: determine classification and fine structure.

(a) (b)

ure 2.8 (a) shows the result of binarization (through the ridge location algorithm of [
enhanced fingerprint image in Figure 2.7(b). (b) shows the results of thinning the imag
single pixel width.

Repair and determine minutiae

cpb 27.7.05 (3)

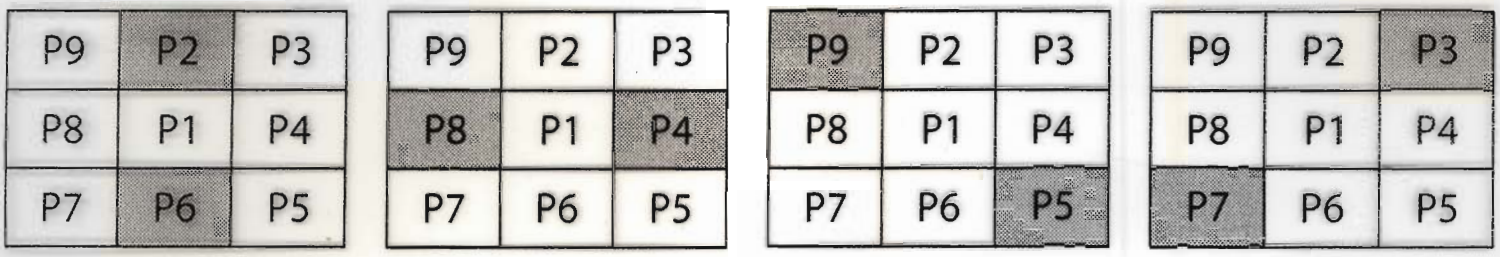
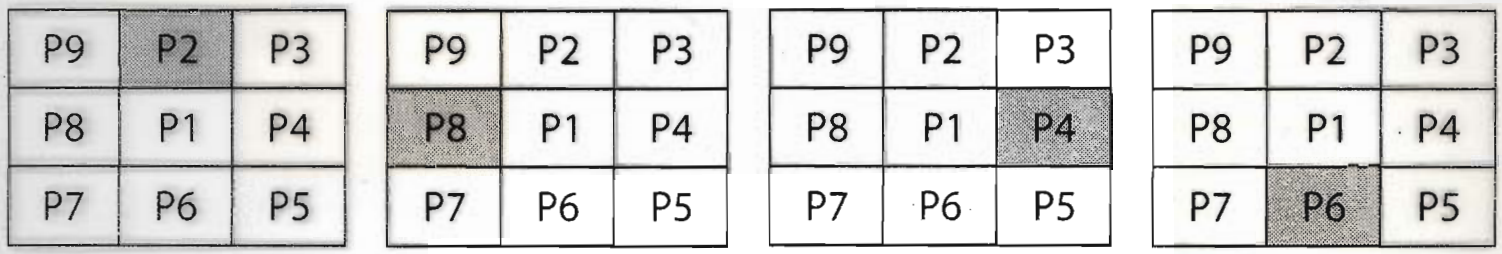


Figure 2.9 Center pixel (P1) is determined to be on a ridge during thinning.

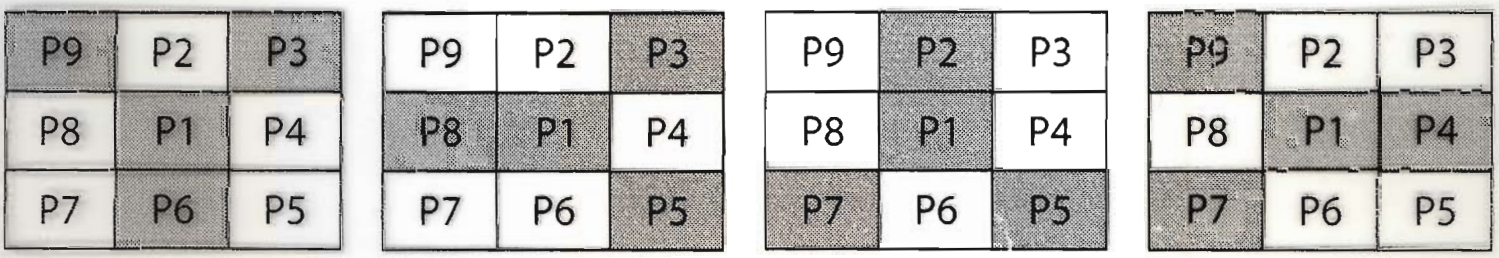
↳ Repair and we color middle pixe in black



South end point | East border point | West border point | North border point

gure 2.10 Center pixel (P1) is determined to be at the end of a ridge during thinning.

↳ Detect line ends!



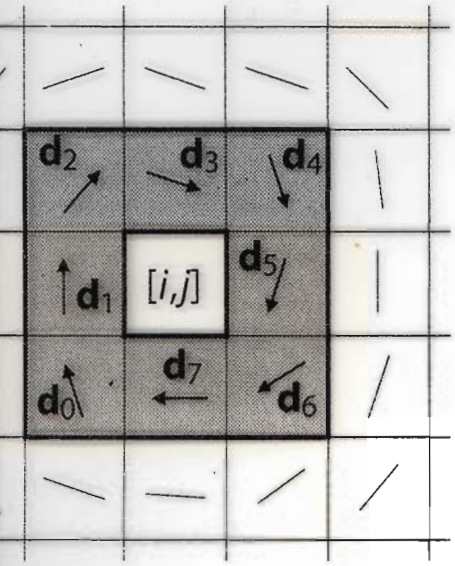
e 2.11 Center pixel (P1) is determined to be a ridge bifurcation minutia during minu tion.

↳ Detect bifurcations!

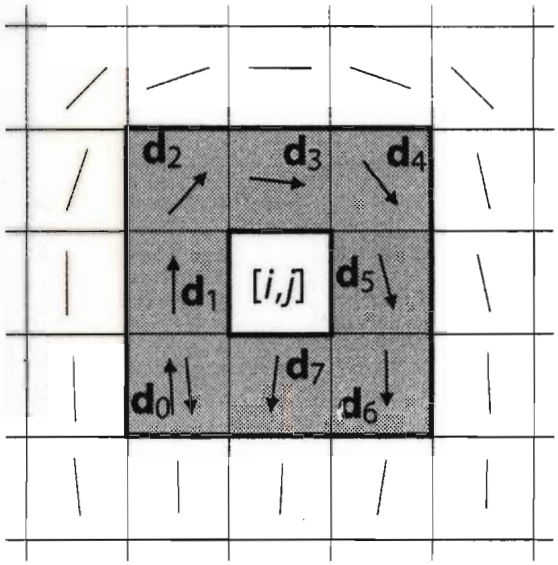
This now very easily gives us the minutiae. If we have retained the angle from earlier analysis we now have all minutiae including the angle info.

Poincaré index

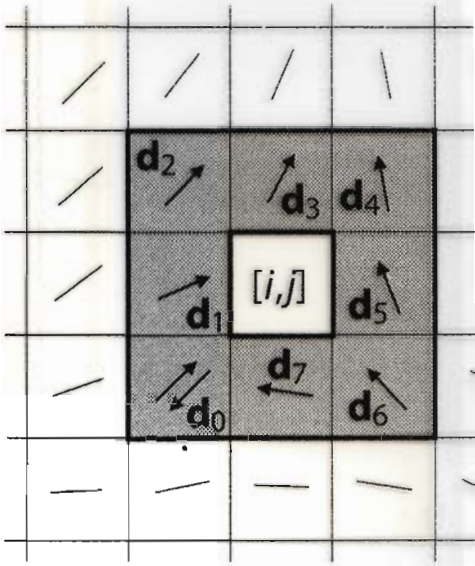
qb
21.7.08
(4)



$P_{\mathbf{G},C}(i,j) = 360°$



$P_{\mathbf{G},C}(i,j) = 180°$



$P_{\mathbf{G},C}(i,j) = -180°$

re 2.13 Example of computation of the Poincaré index in the 8-neighborhood of po
nging (from the left to the right) to a whorl, loop and delta singularity, respectively. Note
ne loop and delta examples (center and right), the direction of $\mathbf{d}_0$ is first chosen upward
pute the angle between $\mathbf{d}_0$ and $\mathbf{d}_1$) and then successively downward (when computing
e between $\mathbf{d}_7$ and $\mathbf{d}_0$) [21].

Combining the angle information around a point allows us to classify

$$P(i,j) = \begin{cases} 0° & \text{if } [i,j] \text{ not singular} \\ 360° & \text{if } [i,j] \text{ is of whorl type} \\ 180° & \text{if } [i,j] \text{ is of loop type} \\ -180° & \text{if } [i,j] \text{ is a delta.} \end{cases}$$

Now we have:

(a) a picture with one-pixel wide lies

(b) Line endings and bifurcations marked

(c) ridge direction

(d) Poincaré index for each pixel

cpb 27.7.08 (5)

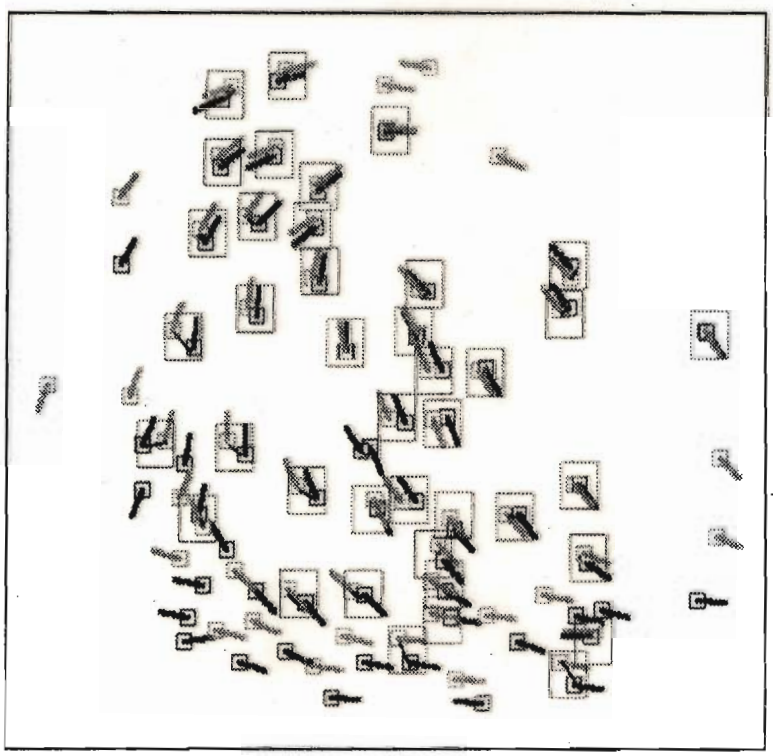
re 2.12 The detected minutiae features are superimposed on the original inked finger
gure 2.3(a) for display.

Matching

cpb
21.7.09
(6)

(a)

(b)



(c)

The computer tries to rotate, shift and rescale, maybe even distort to make as minutiae match as possible.

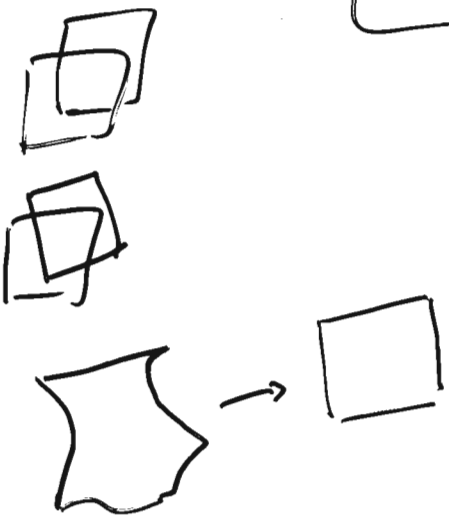
Runs through the database and delivers good matches for further inspection.

cpb
2.7.09
①

Matching

Care of:

- displacement
- rotation
- partial overlap
- nonlinear distortion
- pressure & skin condition
- noise
- feature extraction errors



Summary for fingerprint system

Start: Image

1. LACE
2. Gabor filtering including
   (a) direction computation
   (b) frequency computation
3. convert to 2-color at some threshold
4. thin out lines to 1-pixel-width.
5. determine minutiae incl. features
   - direction
   - Poincaré index

Then: match!

cpb
22.7.08
(2)

Further issues:

- Interoperability
- "Daubert" questions
  Validity in court?
  [1993 US supreme court
  Daubert vs. Merill Dow Pharmac.]

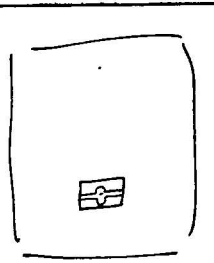
Applications

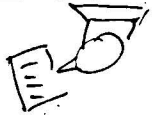
- electronic passport, borders control
- access control → places, buildings, areas, ...
  → notebooks, ...
  → mobile phones, ...
- criminal investigations → cars ...
- welfare fraud reduction
- driver registration
-

cpb
22.7.09
(3)

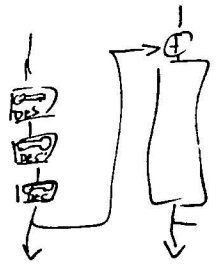
SUMMARY

- Introduction
- Cryptography
  - digital signatures — SECURITY models
  - group theory —
  - key exchange, DH
  - AES, TDES
  - Hash functions
  - PKI, certificate directory structure, CA
- ePassport
  - passive & active authentication
  - Doc 9303
    data structure, communication + protocols, MRZ, Basic Access Control, Security Object
  - Extended Access Control → Chip authentication (repl. Active Authentication and has DH), Terminal authentication
- Biometrics, Fingerprints
  - History, and classification
  - Automation
    - Image quality enhancement, ...

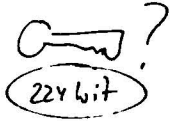


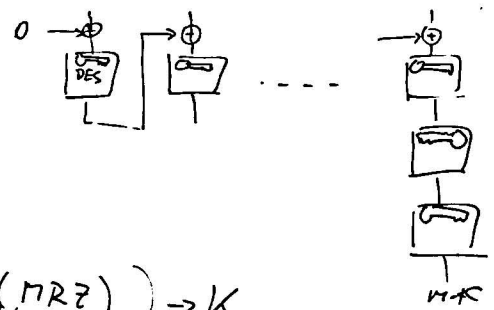


encryption: 2-TDES
(AES?)

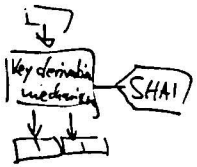


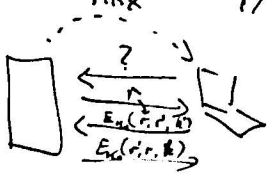
authentication: retail-CBC-MAC





BAC: key derivation(SHA1(MRZ)) → K_0





entropy: ~81 bits
(~56/40 bits)

key derivation($k \| k'$) → K_1

Chip Authentication (EAC)

= DH key exchange

key derivation(xyP) → $K_{2 \ldots}$

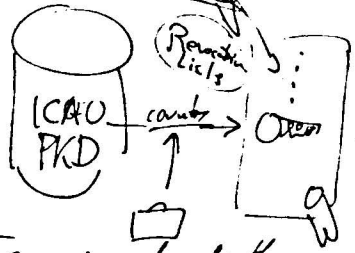
$(P, xP, yP) \xrightarrow{?} xyP$

Diffie-Hellman-Problem

Passive Authentication

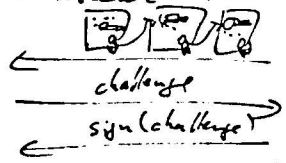
• Chip uses static key pair: (x, xP)

Terminal checks signature ECDSA & that the hash of xP is correct



Terminal Auth.

Check signatures based on a stored Root-CA



Verify.