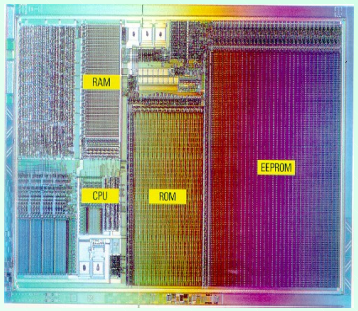
Smart Cards

Danny De Cock
Danny.DeCock@esat.kuleuven.be
Katholieke Universiteit Leuven/Dept. Elektrotechniek (ESAT)
Computer Security and Industrial Cryptography (COSIC)
Kasteelpark Arenberg 10
B-3001 Heverlee
Belgium

These slides can be downloaded at http://godot.be → recently presented slides

What is a Smart Card?

A piece of silicone on and / or in a plastic body



Chip

What is a Smart Card?

- Secure portable device
- Protected piece of hardware
- Contains and handles sensitive data
  - transactions
  - electronic cash
  - identity / healthcare profile
- Contains secret codes and keys
- Performs cryptographic computations for
  - authentication / digital signatures
  - confidentiality by decryption
  - key management protocols
  - data protection

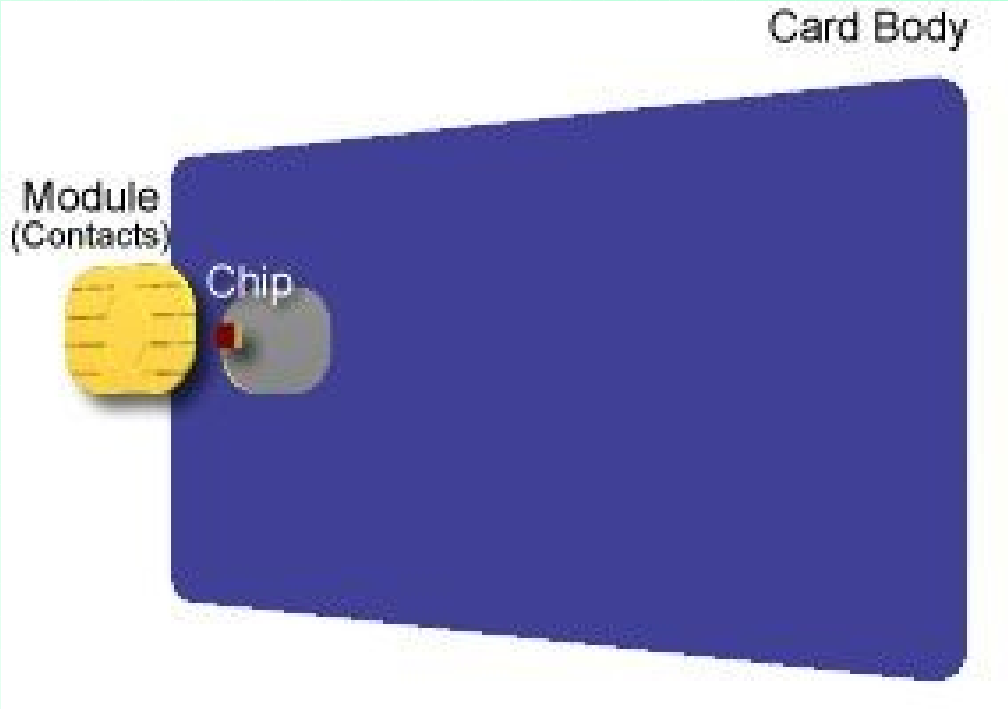
The Smart Card...

- The smart card stores electronic data and programs in a protected file system
  - Protection by advanced security features
  - Tamper evidence
- Several types of smart cards
  - Contact
    - Memory cards
    - Microprocessor cards
  - Contact less
    - RFID chips

Some benefits...

- Security
- Multi-application
- Off-line transactions
- Portable data store
- Personal private keys and codes

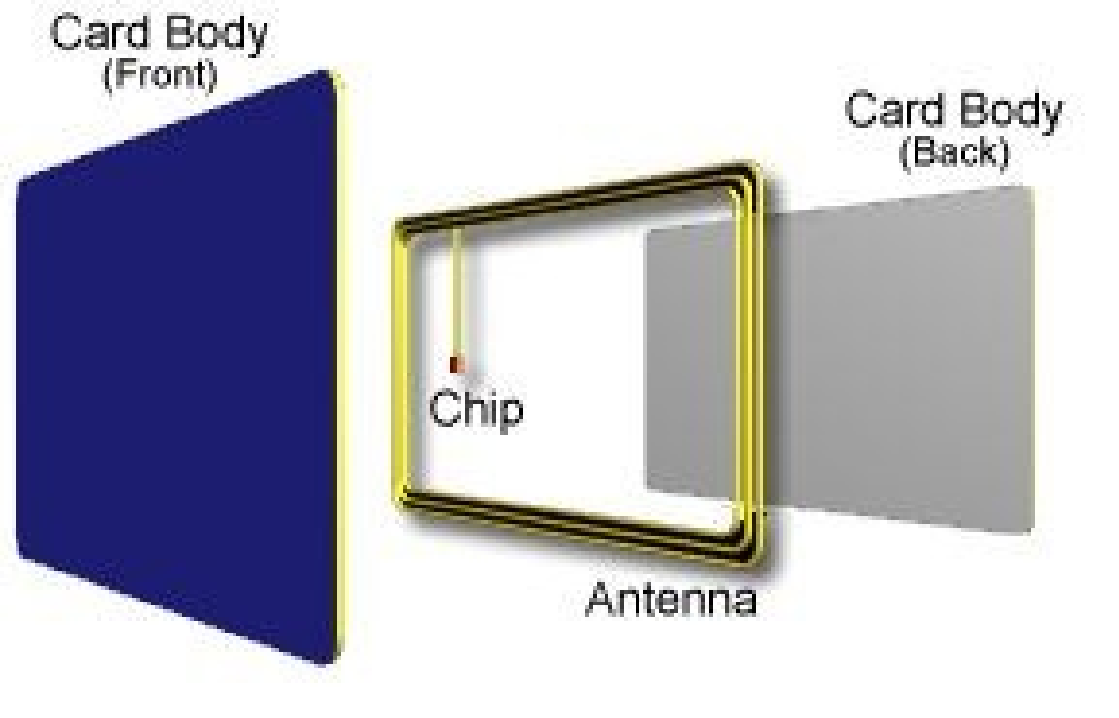
Contact Smart Cards



Communication through electrical contacts

Credits: Gemplus (FR)

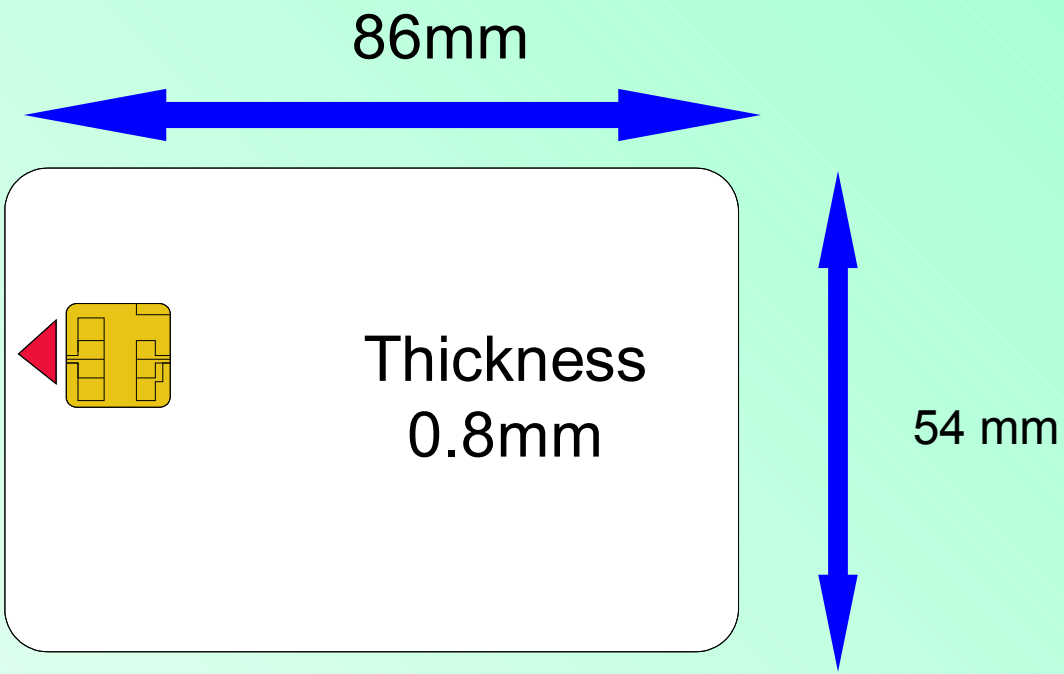
Contactless Smart Cards



Communication over the air

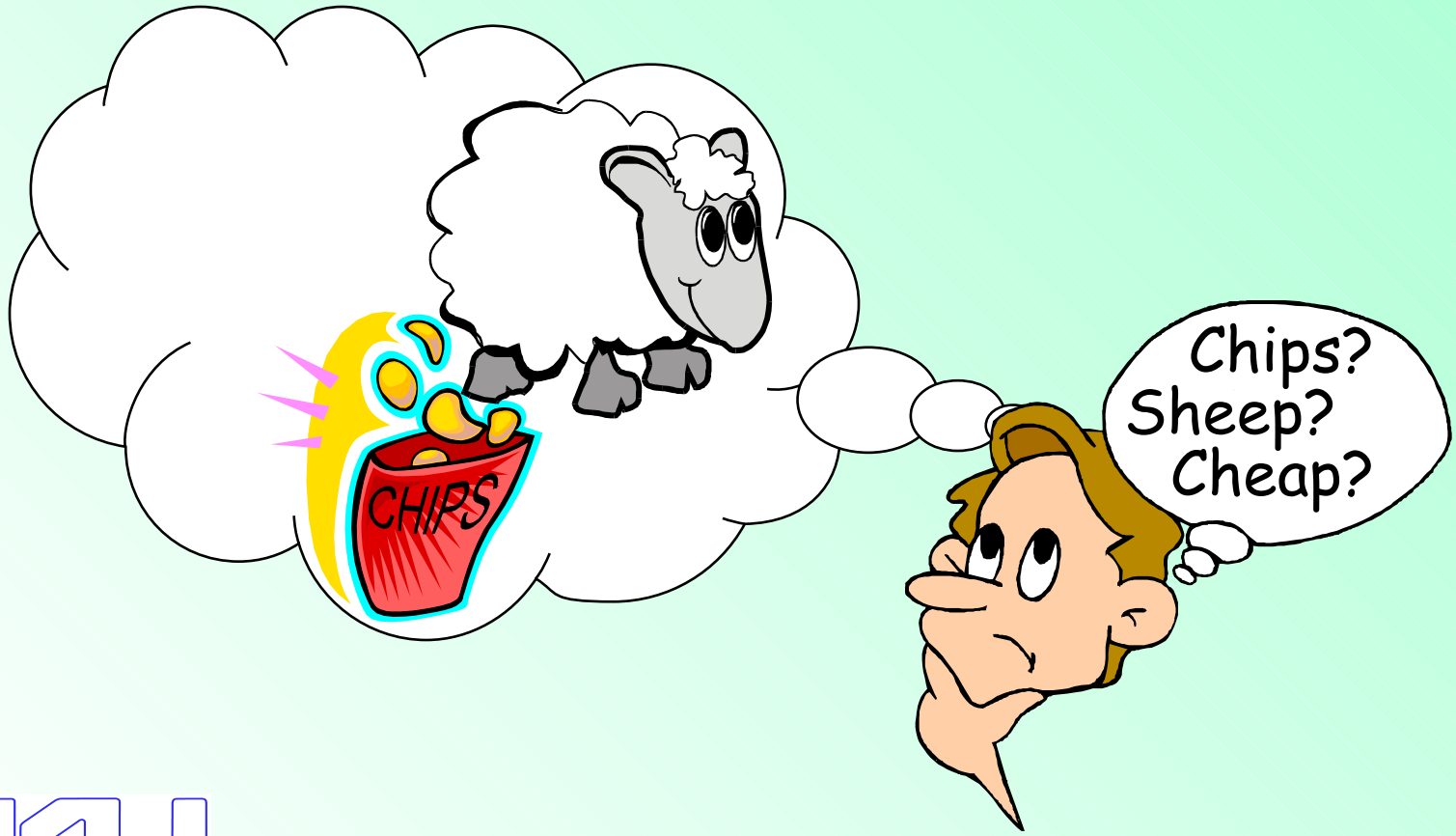
Credits: Gemplus (FR)

The Plastic Card



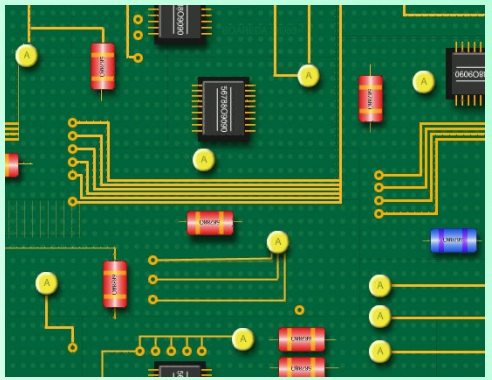
Dimensions and contact positions are standardized in ISO 7816

What is a Chip… ?



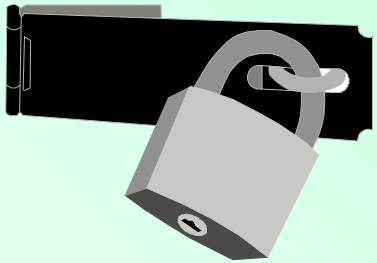
Main Chip Features

- Tiny **microcontroller**
  - Contains crypto coprocessor or not
- Contains **memory**
  - Erasable or not
  - Protected or not



Different Memory Types

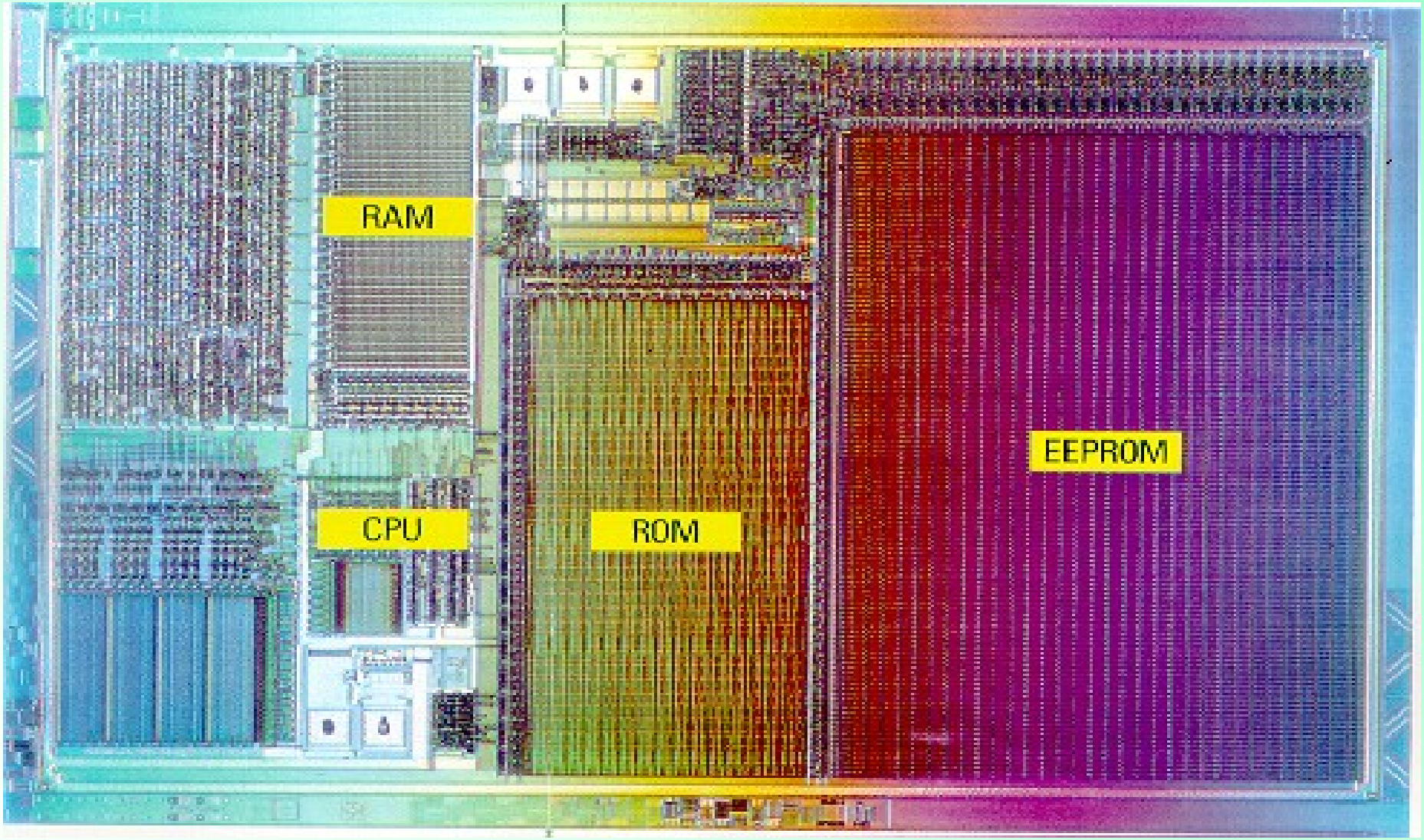
- **ROM** — **CPU only** — **NO ACCESS!**
- **EPROM** — **Write once, read** — **FOREVER!**
- **EEPROM** — **Write, erase, read** — **FLEXIBLE!**
- **RAM** — **Write, erase, read** — **VOLATILE!**



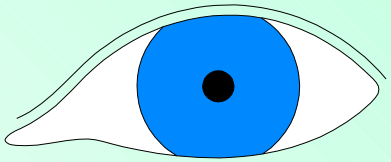
What Does It Stand For?

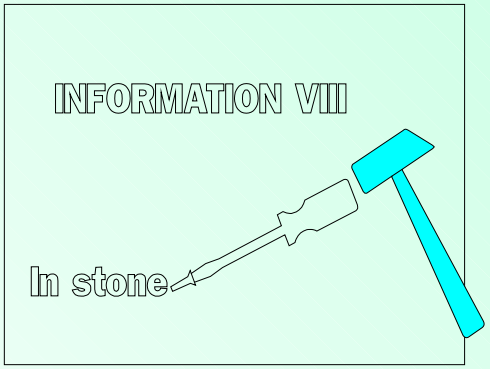
- **ROM**
  - **R**ead **O**nly **M**emory
- **EPROM**
  - **E**lectrically **P**rogrammable **R O M**
- **EEPROM**
  - **E**lectrically **E**rasable **P**rogrammable **R O M**
- **RAM**
  - **R**andom **A**ccess **M**emory

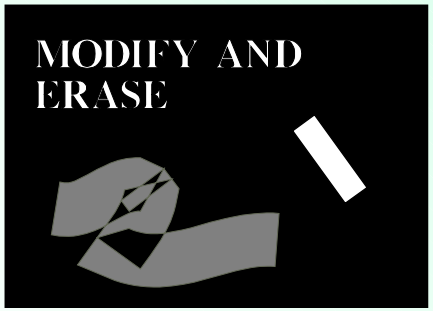
Close-up View...



The Chip Operating System







- File and directory management:
  - Create
  - Read Only
  - Add Information Only
  - Erase and Update
- Access protected by secret codes:
  - Data files
  - Secret Code files
  - Cryptographic key files

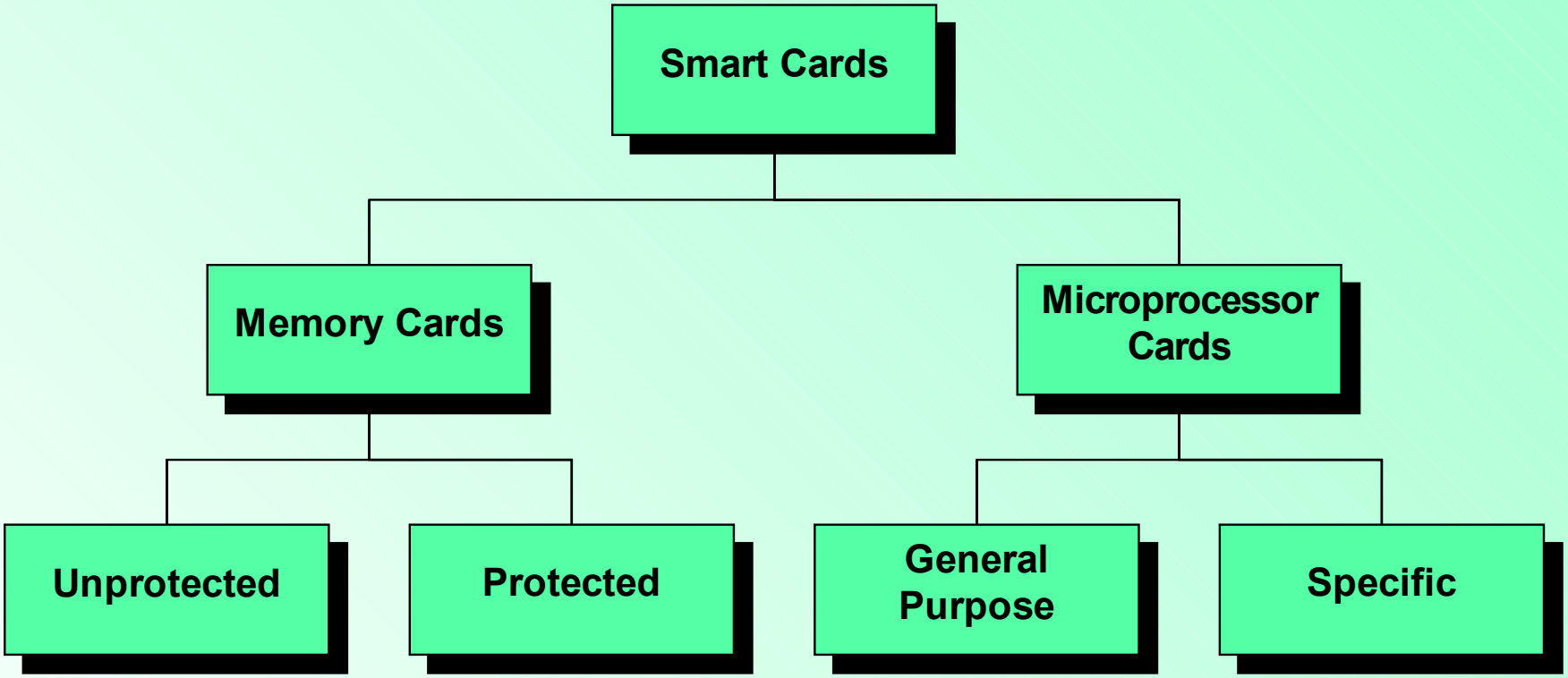
Secure Data Container – Access Control

- None:
  - Some information cannot be accessed by anyone
  - E.g., internal secrets stored in the card
- Public:
  - All information can be freely accessed
  - E.g., certificates, purse balance
- PIN-protected:
  - Sensitive information
  - E.g., use of internal secrets available in the card

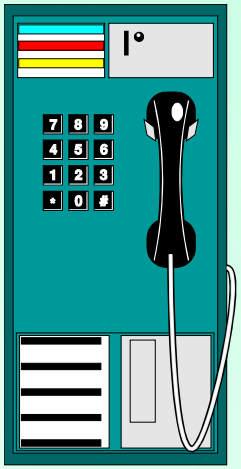
Smart Cards and Cryptographic Speed

- Technical Characteristics of today's cards:
  - 16-bit microprocessor
  - 136 Kbytes ROM
  - 5052 bytes RAM
  - 32 – 64 Kbytes EEPROM
  - Crypto accelerators for RSA, ECC, DES
- Typical figures:
  - DES: 23 µs @ 5 MHz, 8 µs @ 15 MHz
  - 3DES: 35 µs @ 5 MHz, 12 µs @ 15 MHz
  - RSA 1024 sign: 820 ms @ 5 MHz, 273 ms @ 15 MHz
  - RSA 1024 verify: 20 ms @ 5 MHz, 7 ms @ 15 MHz
  - ECDSA 160 sign: 438 ms @ 5MHz, 146 ms @ 15 MHz
  - ECDSA 160 verify: 711 ms @ 5MHz, 237 ms @ 15 MHz

A Wide Range of Capabilities

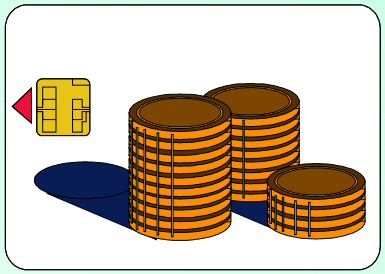


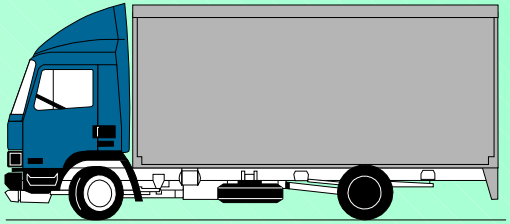
Where Are Smart Cards Already Successful?



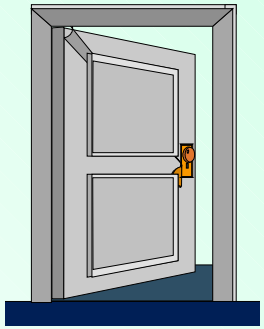
- Payphones
- Mobile Phones
- Bank & Credit Cards
- Social Security
- eID Cards

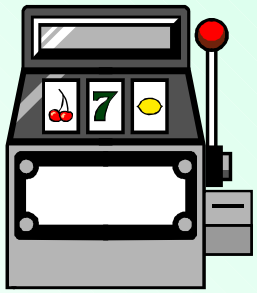
Other Smart Card Applications

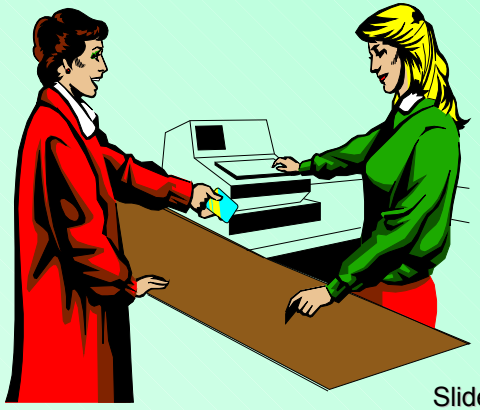




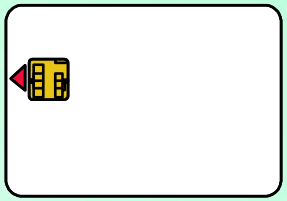
- Electronic Purse
- Transportation
- Physical and Logical Access Control
- **Identity Cards**
- Customer Loyalty
- Gaming
- Pay TV



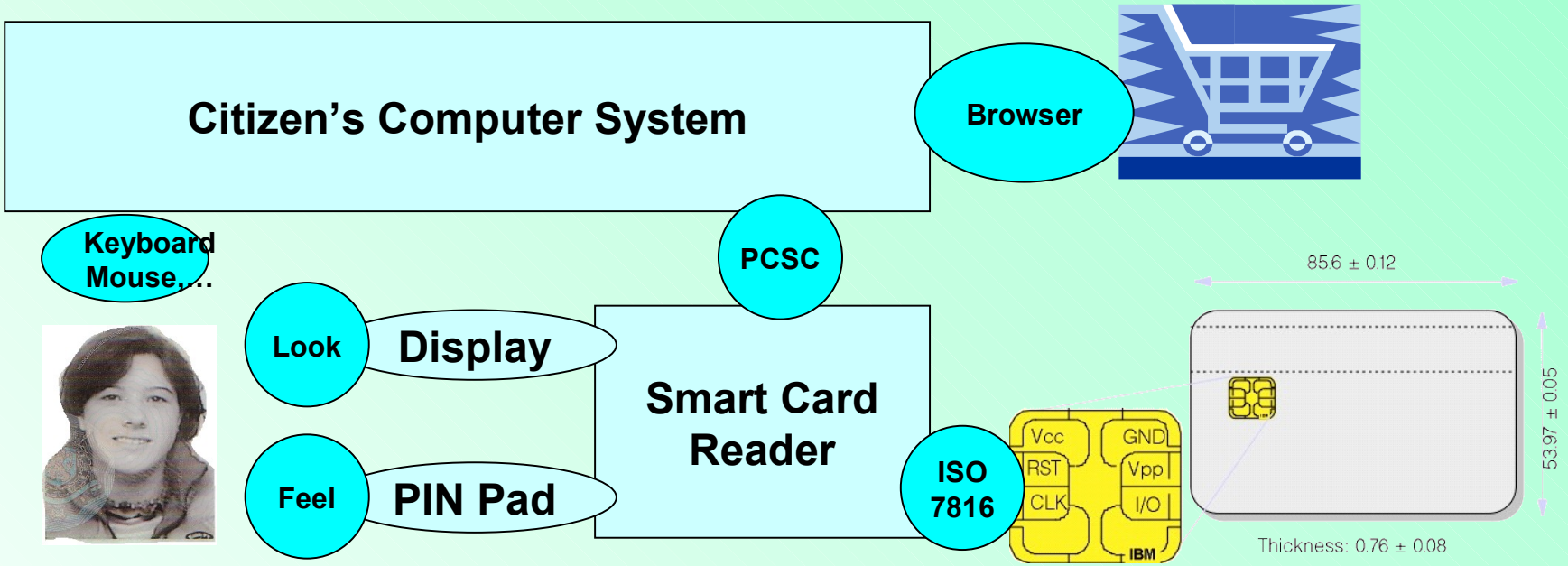




How to communicate with a smart card ?



Typical Smart Card Architecture



Messages

- The card communicates with the reader by exchanging APDU messages (Application Protocol Data Unit)
- A message is either
  - **a Command**: From the reader to the card
  - **a Response**: From the card to the reader

Communication Protocols

- Commands are initiated by the card reader
- T=0 protocol
  - Terminal sends data to the card, or
  - Terminal receives data from the card
- T=1 protocol
  - Terminal sends data to the card, or
  - Terminal receives data from a card, or
  - Terminal sends data to and reads output from the card
- Answer to Reset (ATR)
  - Informs terminal what protocol can be used

APDU command structure

- Command
  - CLA indicates the instruction class
    - Personalization command, normal operation…
  - INS indicates the instruction type
    - Read/write/update file, create/delete file, verify pin, sign…
  - P1/P2 additional instruction parameters
    - File offset, file identifier, key identifier, …
  - Lc length of input data (T=0 or T=1)
  - Data input data (if any) (T=0 or T=1)
  - Le/P3 length of expected output data (T=1)
- Response
  - Output data
    - T=0: In output buffer if Lc ≠ 0
    - T=1: returned to terminal
  - 2 Status bytes
    - 0x9000: Everything OK, 0x6C14: 20 output bytes ready

APDU: Application Protocol Data Unit

APDU Example

- Answer To Reset
  - ATR: 3B9894400AA503010101AD1310
- Get Challenge
  - APDU: 0084000008
  - Output: 9AB2386C06C226B1
- Select File
  - APDU: 00A4080C063F00DF014031
  - Output: 9000

Smart Card Vulnerabilities

- Issuing Process
- Protecting Against Unauthorized Use
- Linking Chip + Plastic

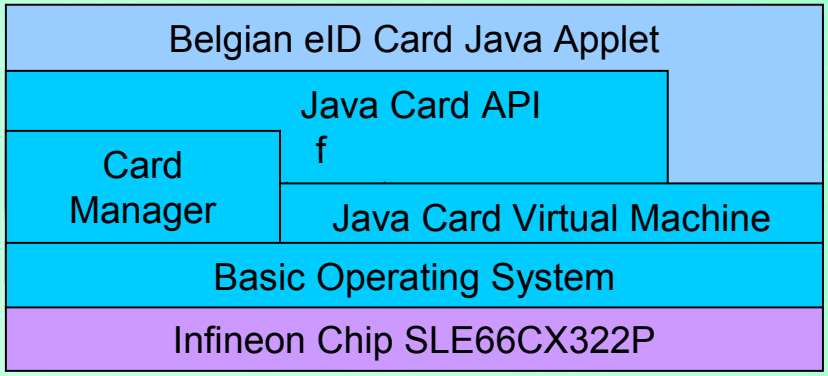
- Solution?
  - RFID Chips

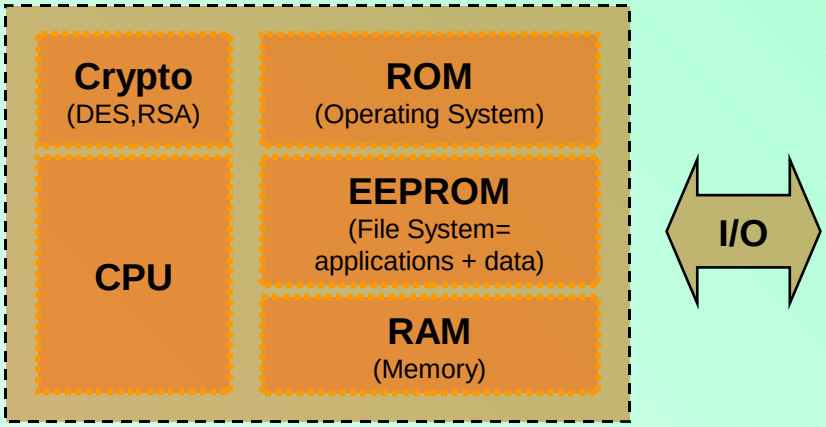
- Use Cases:
  - Belgian eID Cards
  - RFID Passports

Use Case I: Belgian eID Card

eID Card Chip Specifications

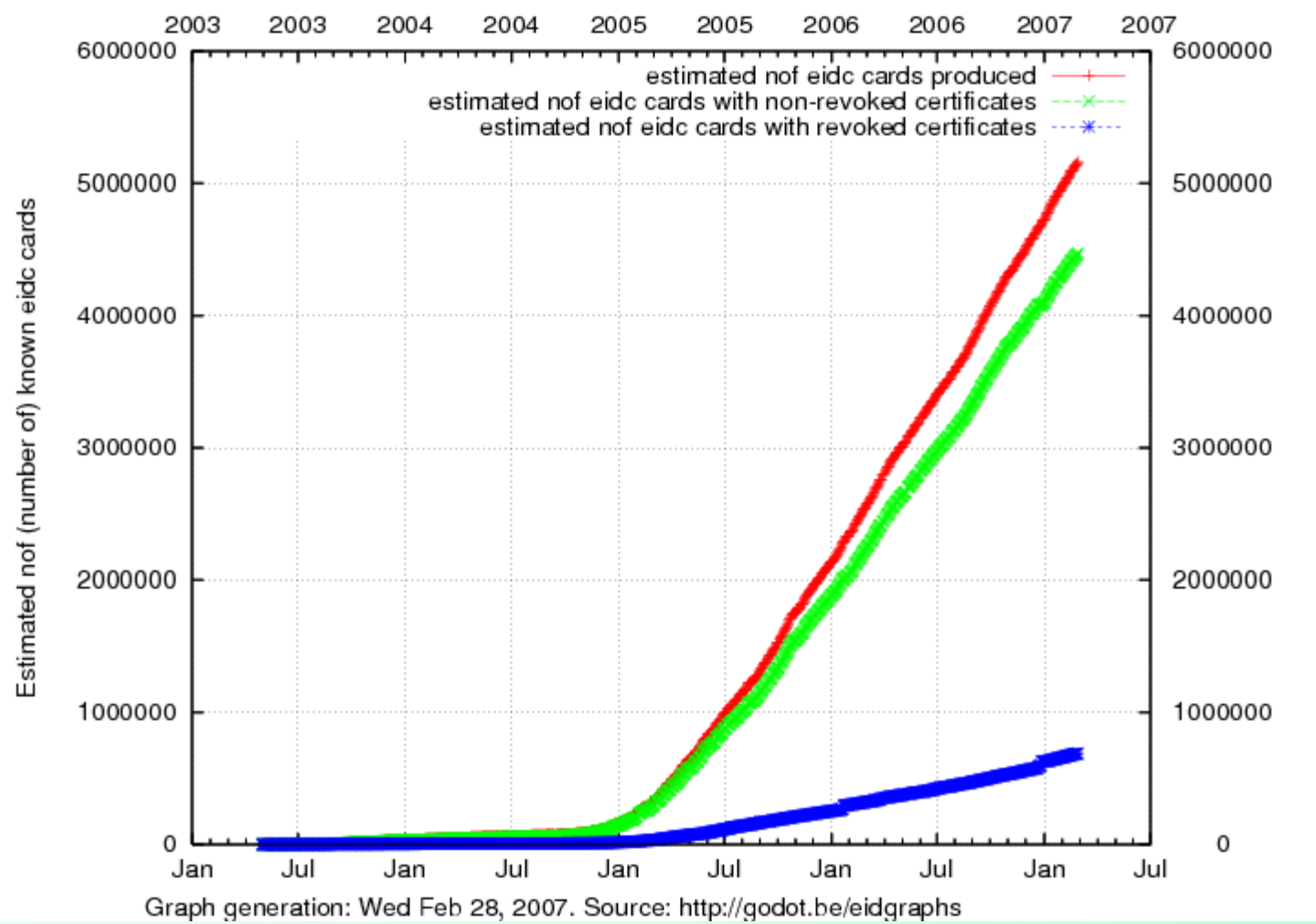
- Cryptoflex JavaCard 32K
  - CPU (processor): 16 bit Microcontroller
  - Crypto-processor:
    - 1100 bit Crypto-Engine (RSA computation)
    - 112 bit Crypto-Accelerator (DES computation)
  - ROM (OS): 136 kB (GEOS Java Virtual Machine)
  - EEPROM (Application + Data): 32 KB (Cristal Applet)
  - RAM (memory): 5 KB
- Standard - ISO/IEC 7816
  - Format & Physical Characteristics ⇔ Bank Card (ID1)
  - Standard Contacts & Signals ⇔ RST, GND, CLK, Vpp, Vcc, I/O
  - Standard Commands & Query Language (APDU)



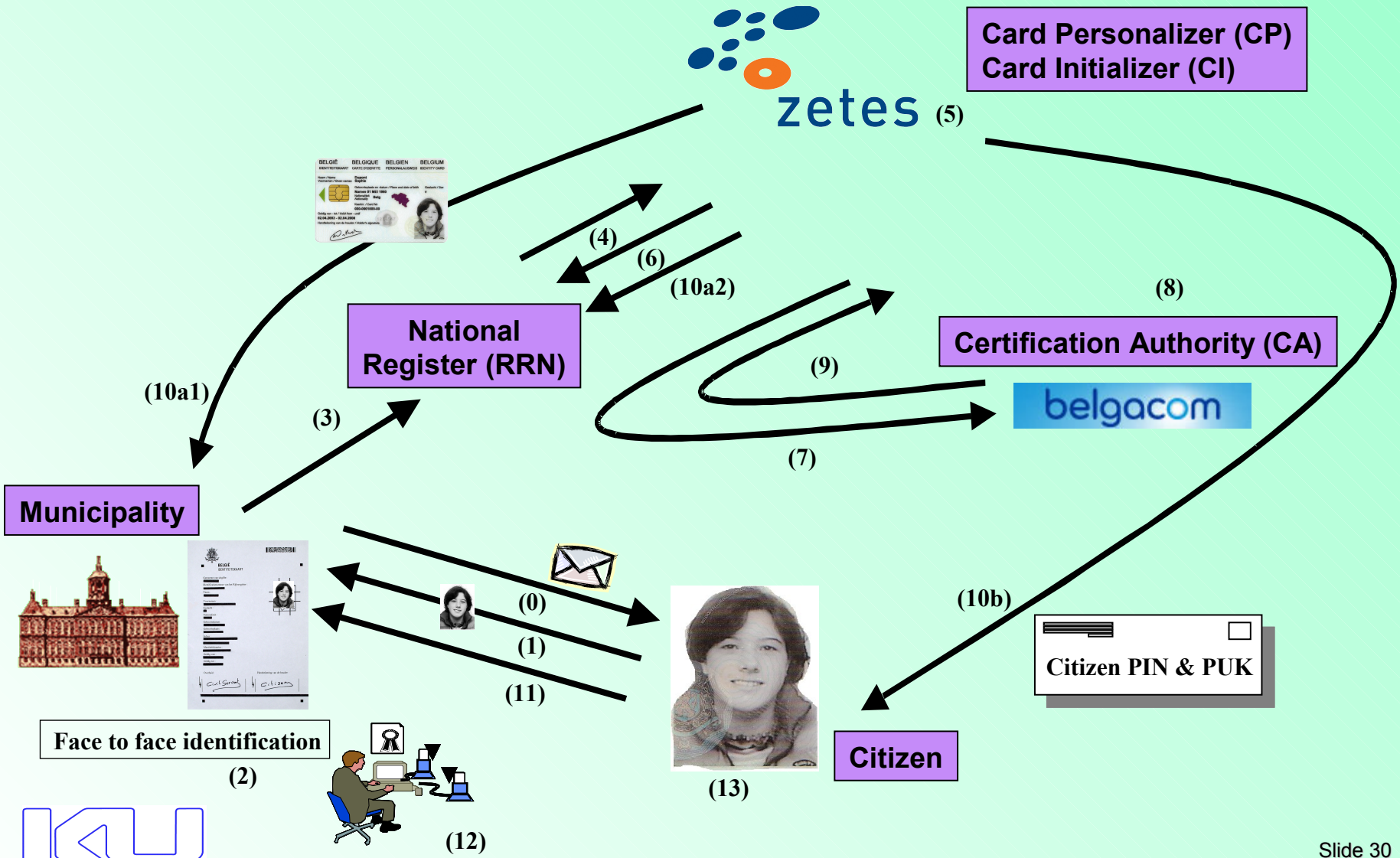


Belgian eID Cards – Quick Summary

- Early 2000, eID concept study
- March 2003, first eID cards issued
- October 2004, massive nation-wide roll out
- September 2005, all 589 municipalities eID ready
- March 2007, 4.5 million cards active
- December 2008, full coverage



eID Card Issuing Process (1/2)



eID Card Issuing Process (2/2)

0: Citizen receives a convocation letter or takes the initiative

1: Visit municipality with photo

2: Formal eID request is signed

3,4: CP receives eID request via RRN

5: CP prints new eID card, CI starts on-card key pairs generation

6: RRN receives part of the eID card activation code PUK1

7: CA receives certificate requests

8: CA issues two new certificates and issues new CRLs

9: CI stores these certificates on the eID card

10a: CI writes citizen data (ID, address,…) to the card, deactivates the card

10b: CI sends invitation letter with citizen's PIN and activation code PUK2

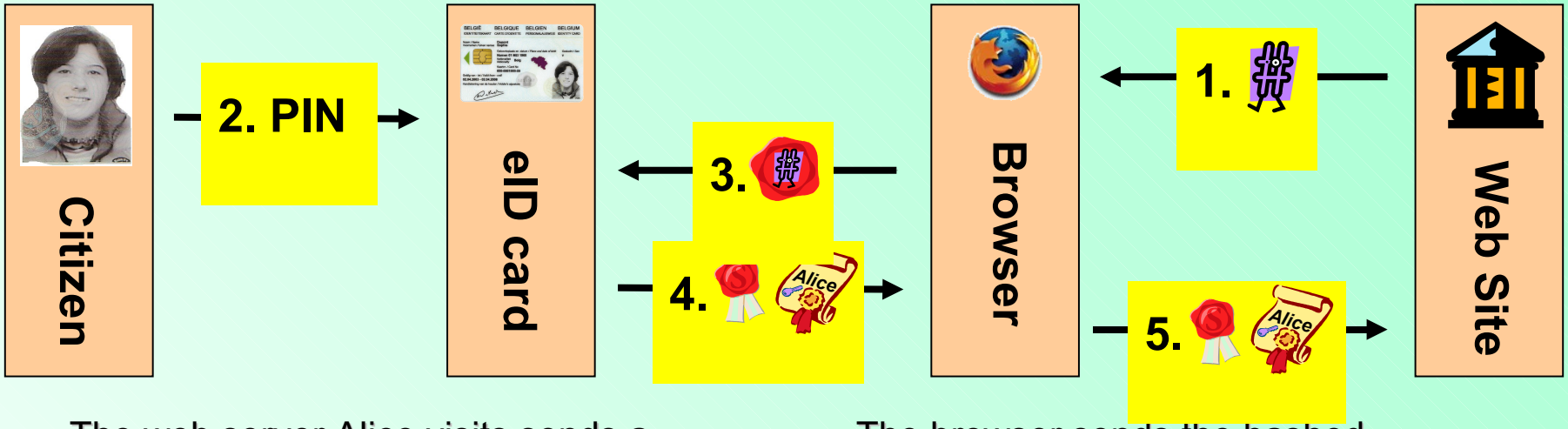
11: Citizen receives invitation letter

12: Civil servant starts eID card activation procedure

13: eID card computes a signature with each private key, CA removes certificates from CRL

Example – Using an Authentication Certificate

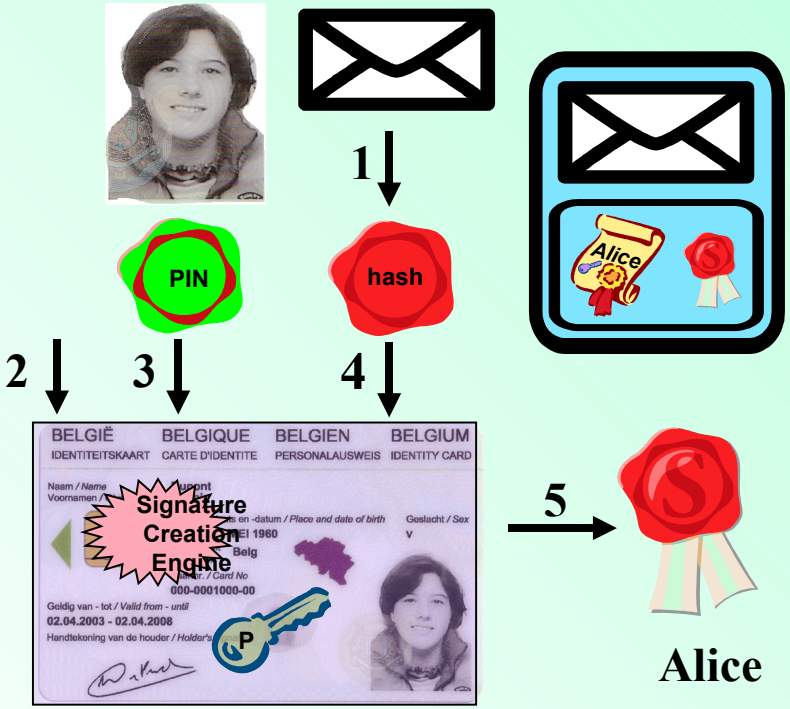
Case study: Alice visits a website which uses client authentication



1. The web server Alice visits sends a random challenge to her browser
2. Alice confirms she wants to log in on the web site by presenting her PIN to her eID card and authorizes the signature generation

1. The browser sends the hashed challenge to Alice's eID card to sign it
2. The browser retrieves the signature and Alice's certificate from her eID card
3. The web server receives Alice's signature and certificate

Signature Generation Steps



Alice's application

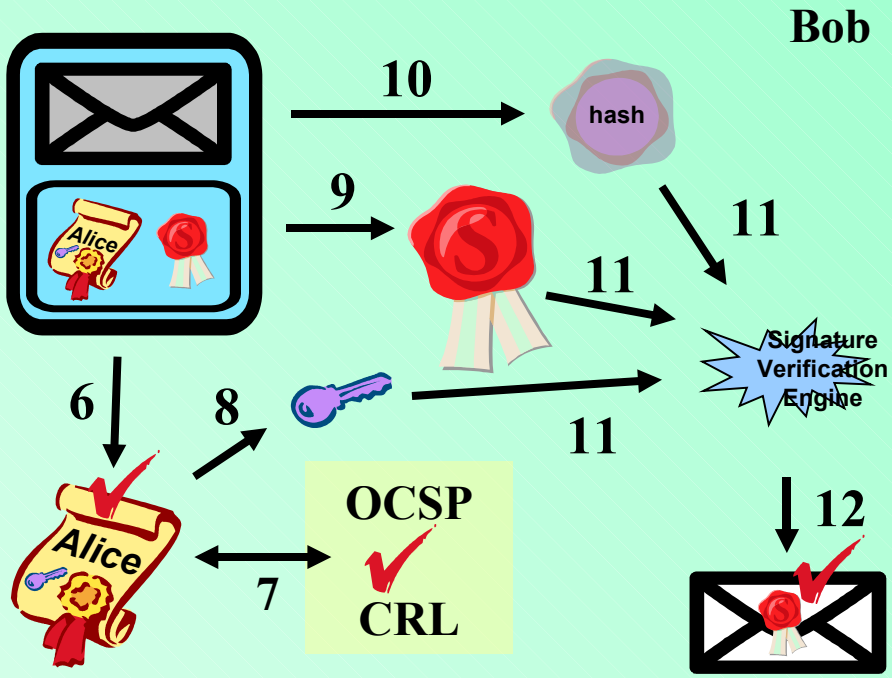
2. Calculates the cryptographic hash on the data to be signed
3. Prepares her eID card to generate an authentication signature or to generate a non-repudiation signature
4. Alice presents her PIN to her eID card
5. Her card generates the digital signature on the cryptographic hash
6. The application collects the digital signature from her eID card

Bob receives an envelope with a digitally signed message and a certificate

Signature Verification Steps

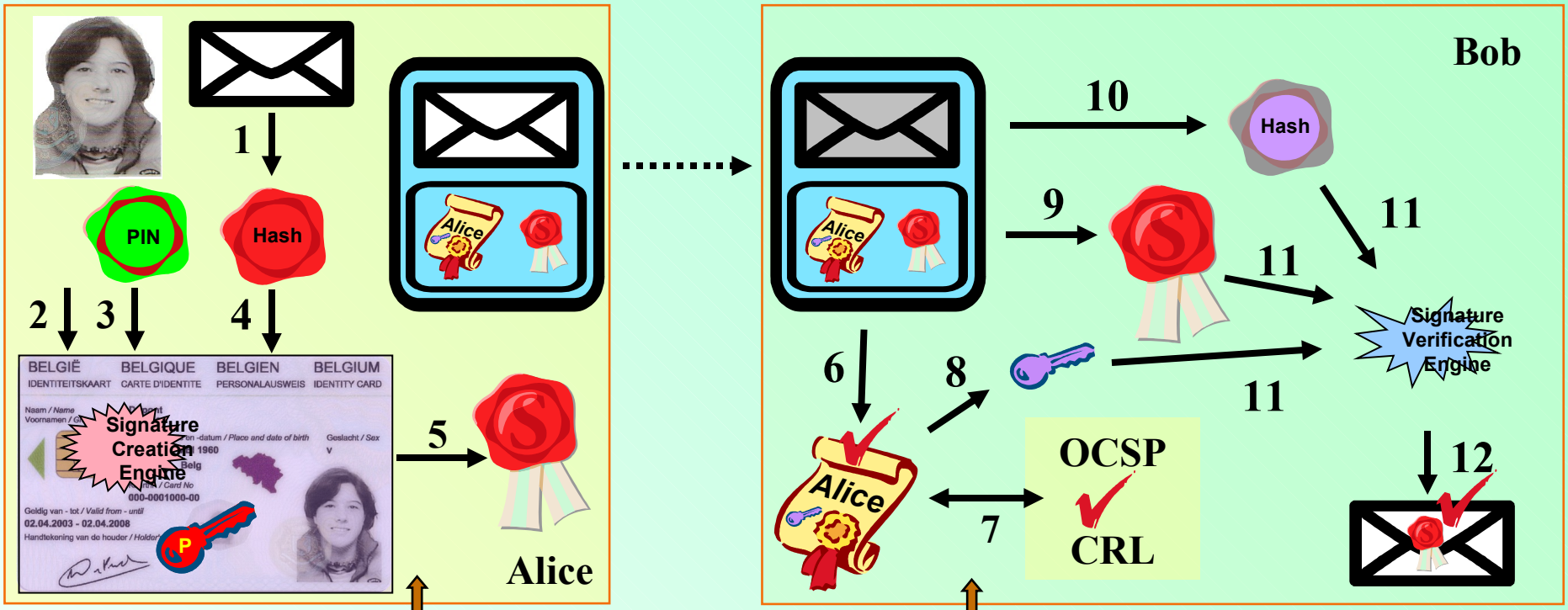
Bob

2. Retrieves the potential sender's certificate
3. Verifies the certificate's revocation status
4. Extracts Alice's public key from her certificate
5. Retrieves the signature from the message
6. Calculates the hash on the received message
7. Verifies the digital signature with the public key and the hash
8. If the verification succeeds, Bob knows that the eID card of Alice was used to produce the digital signature



"*The message comes from Alice*" is a business decision

Signature Generation/Verification

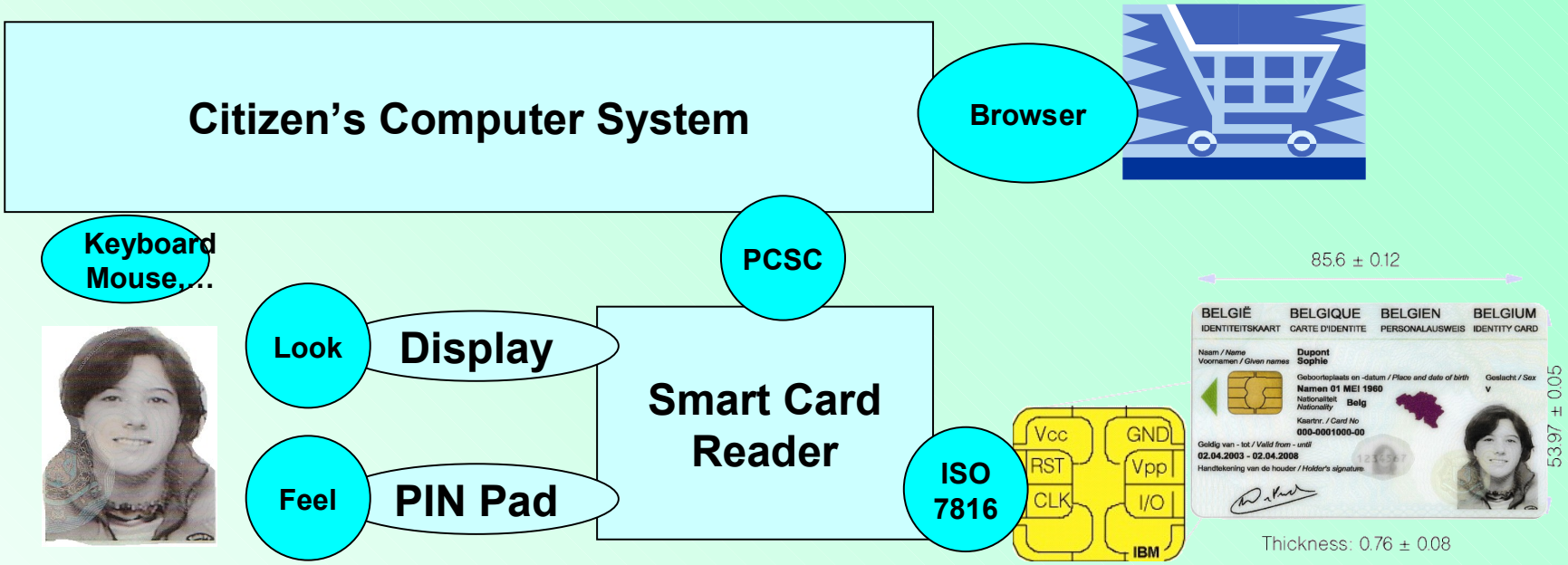


1. Compute hash of message
2. Prepare signature
3. Present user PIN
4. SCD generates digital signature
5. Collect digital signature

6. Retrieve signer certificate
7. Verify the certificate's revocation status
8. Retrieve public key from signer certificate
9. Retrieve digital signature on the message
10. Compute hash on received message
11. Verify digital signature
12. SVD outputs 'valid signature' or 'invalid signature'

Beware – Bob should validate Alice's certificate – Beware

Typical Smart Card Architecture

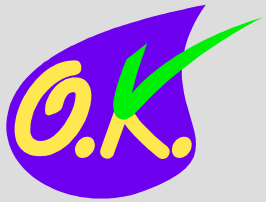


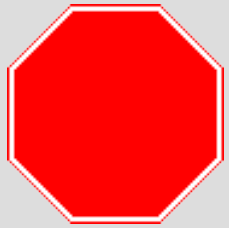
Terrifying Window

PIN entry Window

Your eID card is about to create a qualified signature

Enter your PIN for qualified signatures: ******





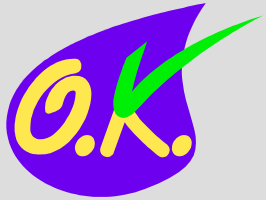
Most Terrifying Window

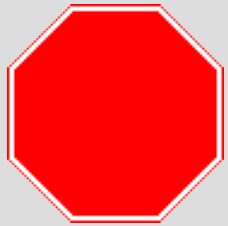
PIN entry Window

Your eID card is about to create a qualified signature

Enter your PIN for qualified signatures: ******

☐ Select the box to make me remember your PIN…





Various Authentication Interfaces

- Authentication of a transaction, client authentication, digital signature,… requires a PIN to be presented to reflect the cardholder's consent

Low *Level of Confidence* *High*

Secure PIN Entry

- Advantages of a secure PIN entry device over a simple smart card reader:
  - Citizen's PIN cannot easily be intercepted by a PC application
- Simply relying on a secure PIN entry device is not enough:
  - The text displayed on the device during a "Verify PIN" command is usually specified by the PC application
  - WYSIWYS: The cardholder does not know which data and commands are sent to the card
- Accepting a cardholder PIN through the PC keyboard should be avoided!!

WYSIWYS: what you see is what you sign

Intermediate Summary

- Smartcards are
  - Very convenient data containers
  - Secure if properly used
- But…
  - Very fragile with respect to access control

- Next step:
  - Back to the future… RFID chips

Use Case II – RFID Passports

- A normal passport + a chip
  - Printed:
    - Citizen name, document number, nationality, gender, citizen photo, birth data, validity period, etc.
    - Machine Readable Zone (MRZ) – passport number, birth date, nationality
  - Electronically:
    - Mandatory:
      - Digital copies of all printed information, including photo
      - Passive data authentication
    - Optional:
      - Fingerprints – short term, privacy-sensitive
      - Iris scan – long term, privacy-sensitive
      - Active (passport) authentication

ePassport – Content Overview

- Physical Document – Booklet
  - Data page
  - Machine-Readable Zone
  - Physical security features
- Digital Document – RF Chip
  - Storage media
  - Cryptographic coprocessor
  - Biometrics
  - Cryptographic security features
- ICAO standardizes (e)Passports

ePassport – ICAO Recommendations

- International Civil Aviation Organization
  - ICAO is a UN organization
  - Specifies technical recommendations for passports
- ePassport Technical Reports
  - Deploy biometrics
  - Logical Data Structure (LDS)
  - Digital signatures
  - PKI & Security
    - Passive authentication (mandatory) — Unforgeability
    - Active authentication (optional) — Copy-Protection
    - Basic/Extended Access Control (optional) — Access Control

ePassport – Security Requirements

- Unforgeability
  - Digital content of the chip
- Copy protection
  - Copies of the digital document must be detectable
- Access control
  - Unauthorized reading of personal data must be prevented

ePassport – Passive Authentication

- PKI for passports
  - Country Signing CA – National (Passport) Root CA
  - Document Signer(s) – Passport manufacturer

Algorithm	Country Signing CA	Document Signer
RSA/DSA	3072 bit modulus	2048 bit modulus
ECDSA	256 bit	224 bit

- Issuing state digitally signs the Document Security Object (SO_D)
  - Contains hashes of the Logical Data Structures (LDSs)
  - Proves that the SO_D and the LDS are genuine
  - Does not protect against copying the chip content or chip substitution

Prevents against forging!

ePassport – Active Authentication

- Every passport has its own key pair
  - Public key stored in digital document DG15
  - Private key is stored in secure memory of the chip
- Challenge-Response Protocol
  - Terminal challenges passport chip
  - Chip digitally signs the challenge
- Possible problems
  - Chip in the middle attacks
  - Replay of Challenges – request & forward signature from genuine passport

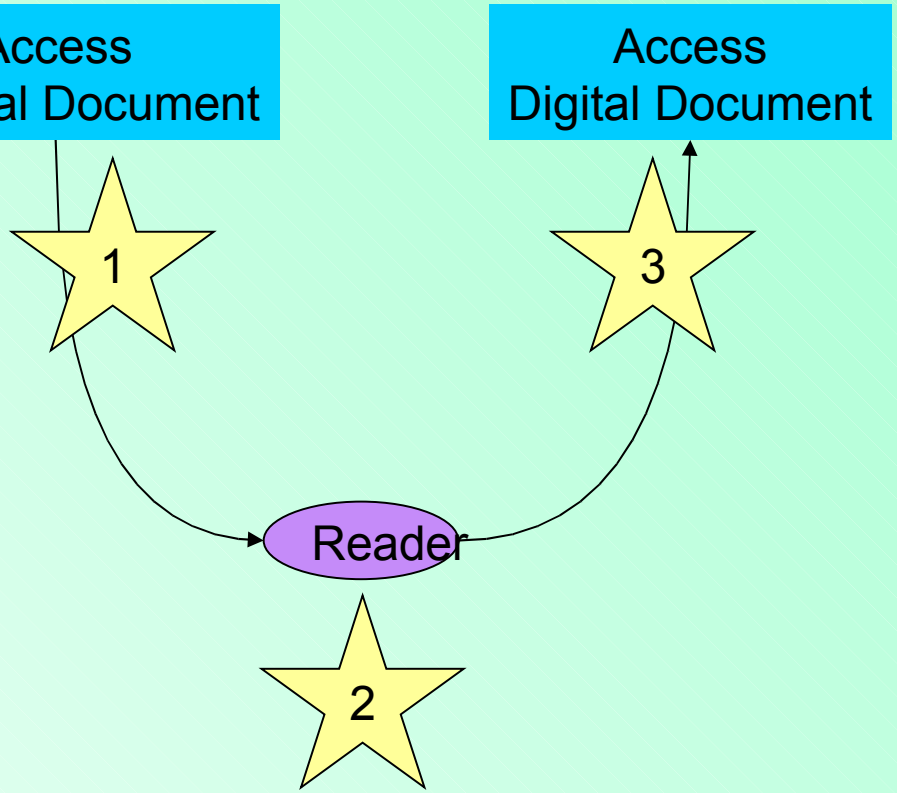
Prevents against cloning!

ePassport – Access Control

- Sensitivity of (biometric) data
  - Face – MRZ (less-sensitive)
    - Can be obtained easily from other sources
    - Required for global border crossing
    - Requires Basic Access Control
  - Fingerprints, Iris (sensitive)
    - Difficult to obtain from other sources (at a large scale)
    - Only used for national/bilateral purposes
    - Requires Extended Access Control (unspecified)

Prevents against evil reader!

ePassport – Basic Access Control – Principle



1. Read MRZ optically
2. Calculate Access Key
3. Authenticate + Read

Prevents against eavesdropper!

ePassport – Basic Access Control – Details

RF Chip in ePassport

3. Generate Challenge C
4. Generate Key C

ePassport Reader

1. Read MRZ
2. Calculate Access Key K

Challenge C →

5. Obtain Challenge C

6. Generate Challenge R
7. Generate Key R
8. Encrypt Challenges C, R + Key R

← Ciphertext

9. Decrypt Ciphertext
10. Verify Challenge C
11. Encrypt Challenge R, C + Key C

Ciphertext →

12. Decrypt Ciphertext
13. Verify Challenge R

After successful authentication:
Secure Messaging + Access to less-sensitive data

ePassport – Basic Access Control – Security

- Length of the access key
  - Approx. 56 bits if passport number is numeric
  - Approx. 73 bits if passport number is alphanumeric
- Goal – anti-eavesdropping protection
  - At least 56 bits are necessary for less-sensitive data
  - At least 112 bits are required for sensitive data

➔ Dangerously short key!

RFID Vulnerabilities

- ISO14443A-compatible RFID chip
  - Range with *standard* reader: 10cm
  - BSI has working setup to eavesdrop on the communication within several meters
- RFID chips typically identify themselves with 4-byte identifier
- Rely on external power source

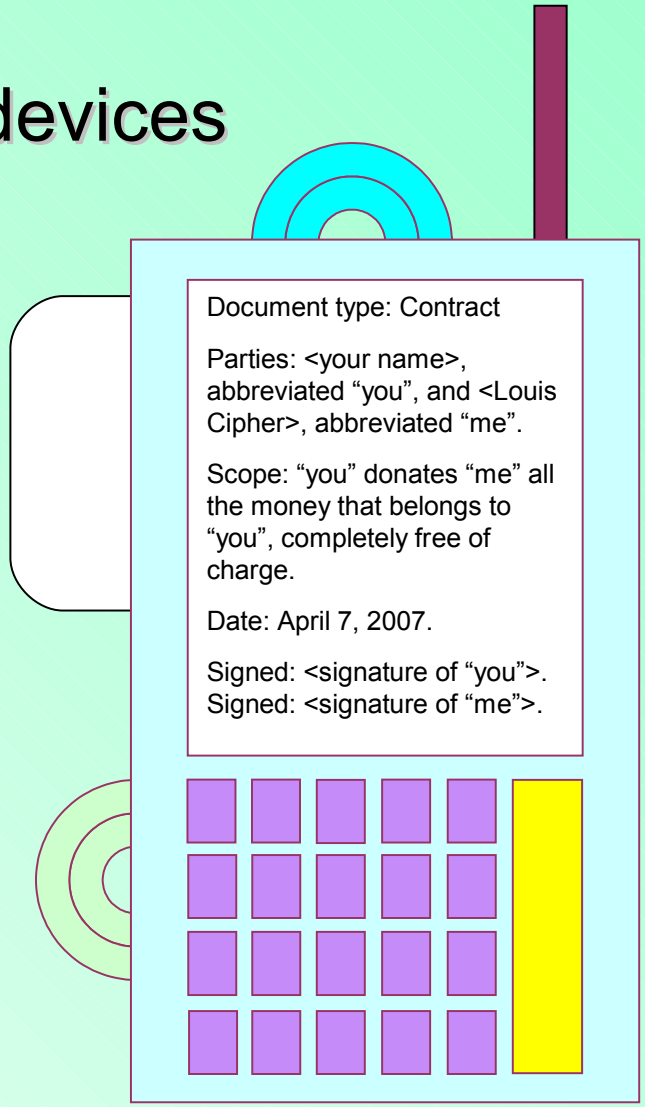
  ➔ Attractive analysis target ☺
- Large advantage:
  - Strong link between chip and booklet

Outlook…

- Smart card → Smart card Device
  - Smart card + reader + user interface
  - ☺ Solves all issues, but…
  - ☹ Not cheap…

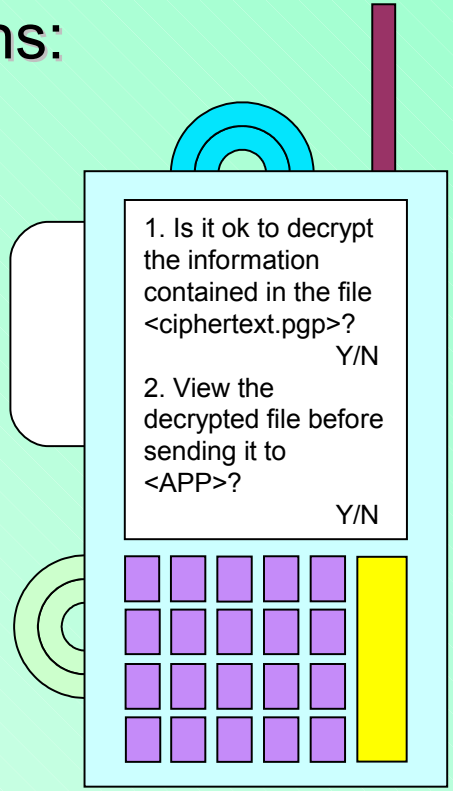
Hardware Requirement

- Smartcards become smartcard devices
  - Tamper-evident crypto chip
  - Reasonable CPU + storage
  - Contact interface
  - Visual feedback
  - Keypad
  - Smartcard slot
  - Biometric sensor
  - Contactless interface
    - RFID, Bluetooth, Infrared
  - Audible feedback
  - Large battery pack



Using Smartcard Devices

- Smartcard Devices need high-level operations:
  - Digital signatures: Sign/Verify this data
    - Input: data to be signed
    - Internal processing:
      - Authentication of the user
      - Hashing, signing, encapsulation
    - Output: envelope with signed data
  - Data encryption: Decrypt this data
    - Input: encrypted data
    - Internal processing:
      - Authentication of the user
      - Decryption
    - Output: cleartext data
- Strong link with DRM technology!



(DRM: Digital rights management)

Questions?

- Email: Danny.DeCock@esat.kuleuven.be
- Web: http://godot.be
- Slides: http://godot.be/slides

That's it…