



# Update on Belgian eID 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

# For Your Information ☺

- The copyright holder of this information is Danny De Cock (email: [godot@godot.be](mailto:godot@godot.be)), further referenced as the author
- The information expressed in this document reflects the author's personal opinions and do not represent his employer's view in any way
- All information is provided as is, without any warranty of any kind
- Use or re-use of any part of this information is only authorized for personal or not-for-profit use, and requires prior permission by the author



# Overview of eID Card Types

## 1. Belgian Kids:

- Kids card with two revoked certificates, age  $< 6$
- Kids card with valid authentication & revoked non-repudiation certificate,  $6 \leq \text{age} < 12$

## 2. Belgian youngster:

- eID card with valid authentication & revoked non-repudiation certificate,  $12 \leq \text{age} < 18$

## 3. Belgian adults:

- eID card with two valid certificates,  $18 \leq \text{age}$

## 4. Foreign kids:

- Kids card with two revoked certificates, age  $< 6$
- Kids card with valid authentication & revoked non-repudiation certificate,  $6 \leq \text{age} < 12$

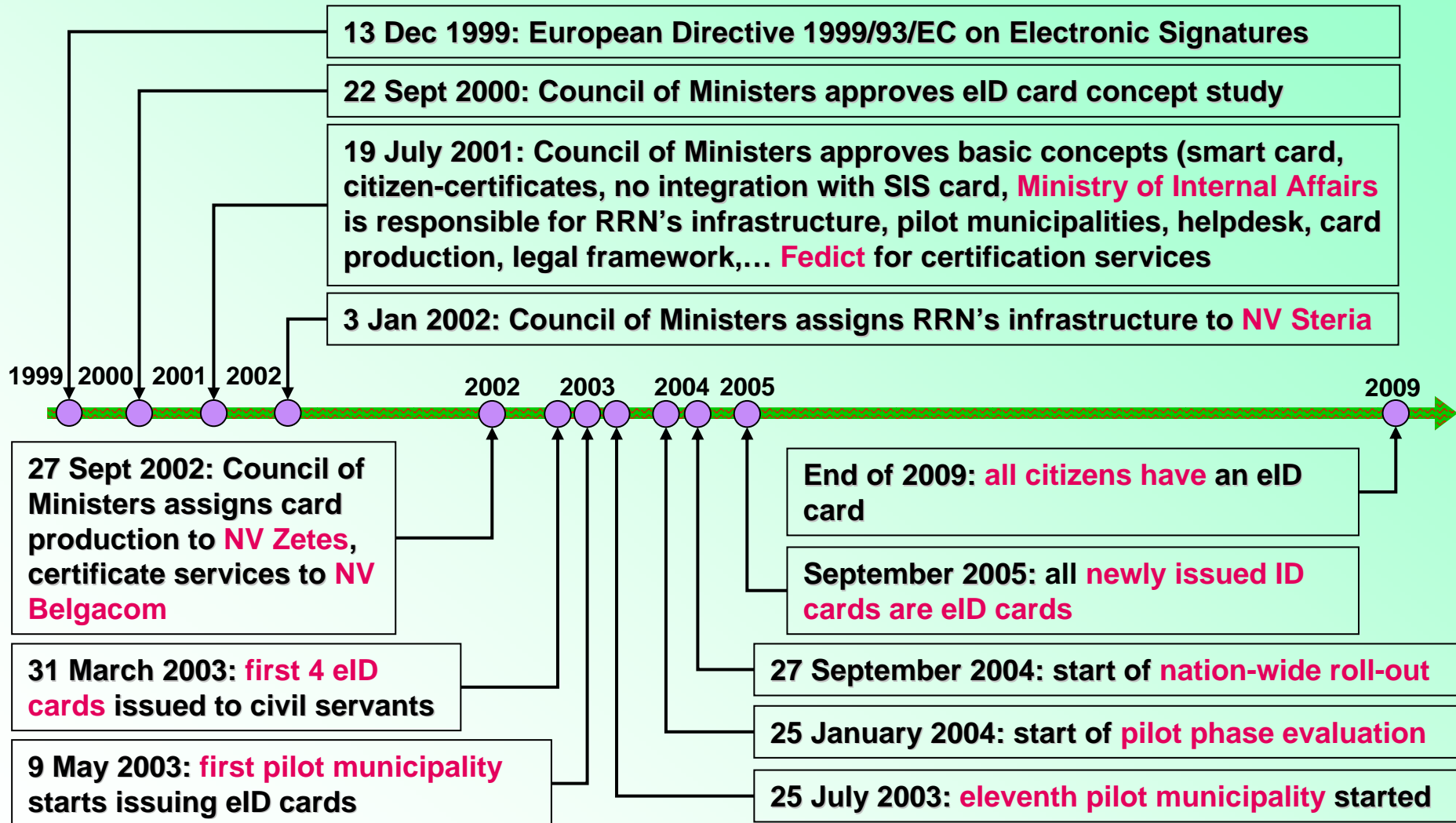
## 5. Foreign youngster:

- eID card with valid authentication & revoked non-repudiation certificate,  $12 \leq \text{age} < 18$

## 6. Foreign adults:

- eID card with two valid certificates,  $18 \leq \text{age}$

# Belgian eID Project Time line



# eID Card = 4 Functions

- Non-electronic

1. Visible Identification of a person

- Electronic

2. Digital identification

- Data capture

3. Prove your identity

- Authentication signature

4. Digitally sign information

- Non-repudiation signature

**Enabler of eServices**

eFunctionality

# Visual Aspects of a Belgian eID card

## Front:

- Name
- First two names
- First letter of 3rd name
- Title
- Nationality
- Birth place and date
- Gender
- Card number
- Photo of the holder
- Begin and end validity dates of the card
- Hand written signature of the holder



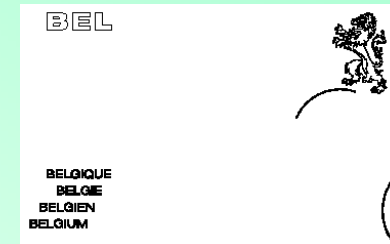
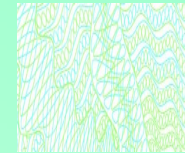
## Back side:

- Place of delivery of the card
- National Register identification number
- Hand written signature of the civil servant
- Main residence of the holder (cards produced before 1/1/2004)
- International Civil Aviation Organization (ICAO)-specified zone (cards produced since 1/1/2005)



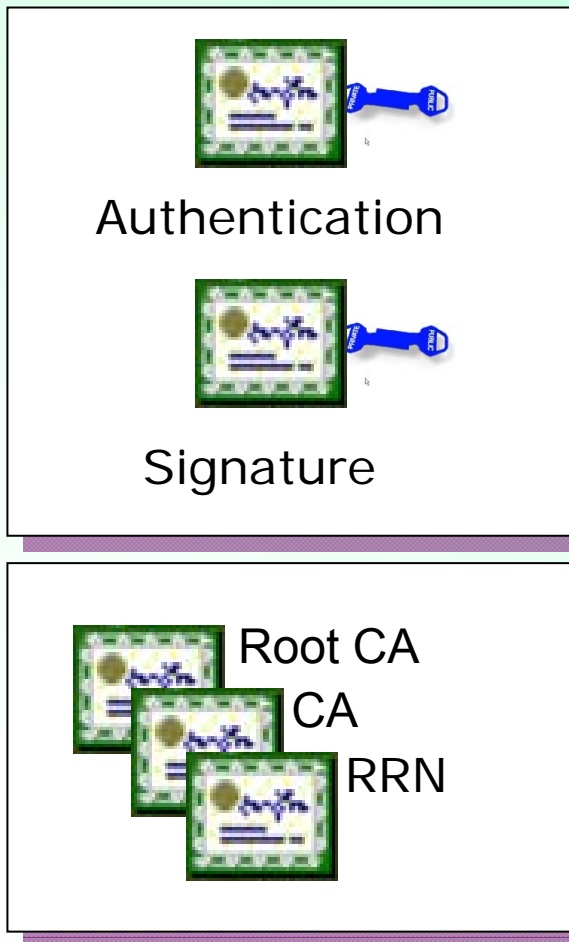
# Visual Security Mechanisms

- Rainbow and guilloche printing
- Changeable Laser Image (CLI)
- Optical Variable Ink (OVI)
- Alpha gram
- Relief and UV print
- Laser engraving

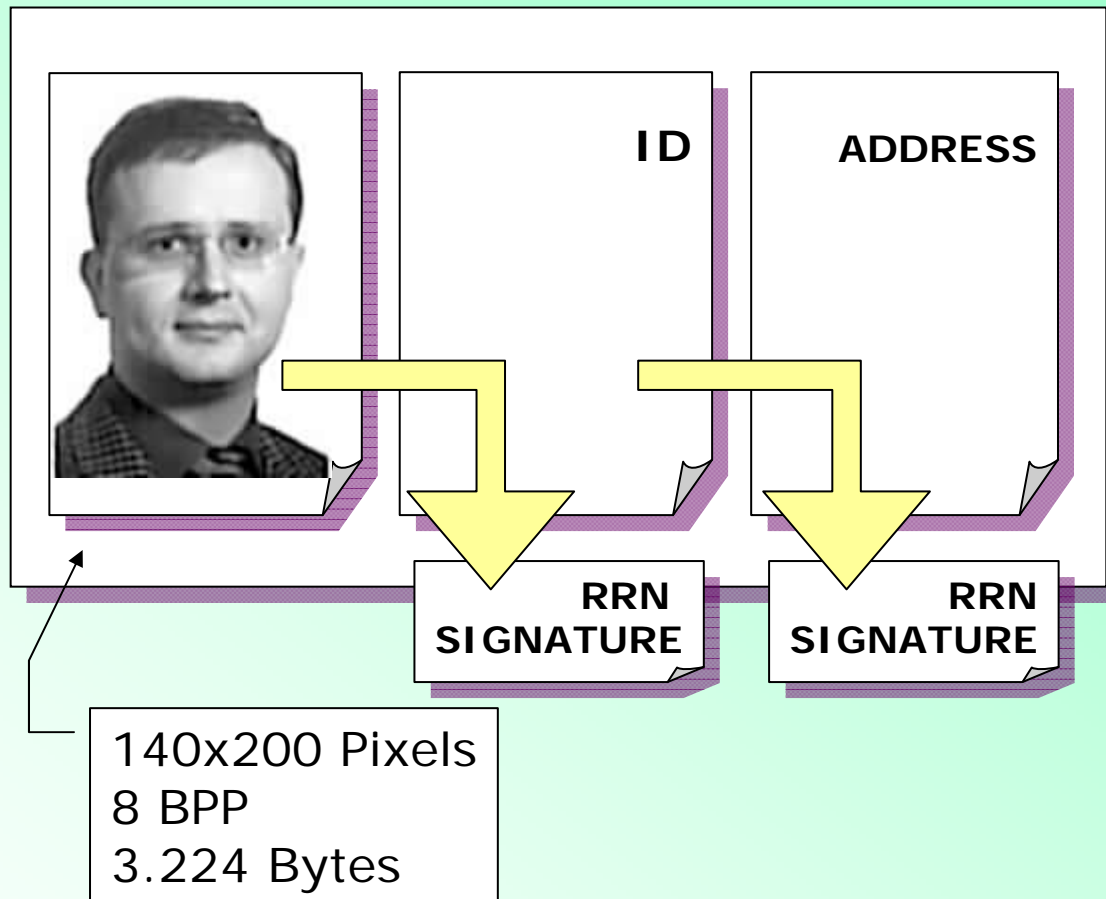


# eID Card Content

## PKI



## Citizen Identity Data



RRN = National Register

# Digital Identification – Identity Files

## ■ Identity file (~160 bytes)

- Chip-specific:
  - Chip number
- Citizen-specific:
  - Name
  - First 2 names
  - First letter of 3<sup>rd</sup> first name
  - RRN identification number
  - Nationality
  - Birth location and date
  - Gender
  - Noble condition
  - Special status
  - SHA-1 hash of citizen photo
- Card-specific:
  - Card number
  - Validity's begin and end date
  - Card delivery municipality
  - Document type

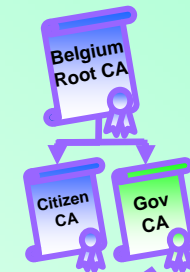
## ■ Citizen's main address file (~120 bytes)

- Street + number
- Zip code
- Municipality
- Digital signature on main address and the identity file issued by the RRN
- Citizen's JPEG photo ~3 Kbytes

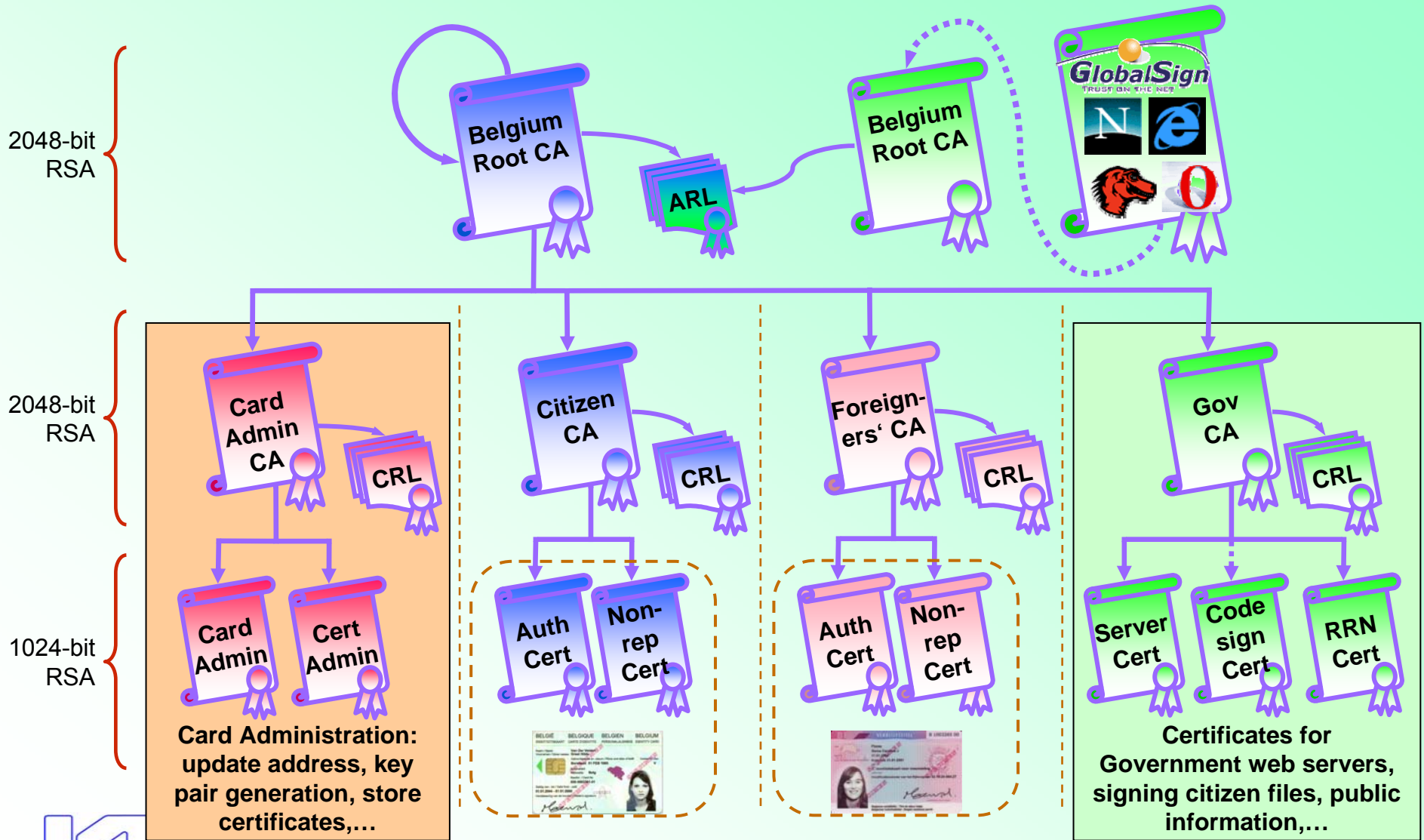
King, Prince, Count, Earl, Baron,...

No status, white cane (blind people), yellow cane (partially sighted people), extended minority, any combination

Belgian citizen/kid, European community citizen/kid, non-European community citizen/kid, bootstrap card, habilitation/machtigings card

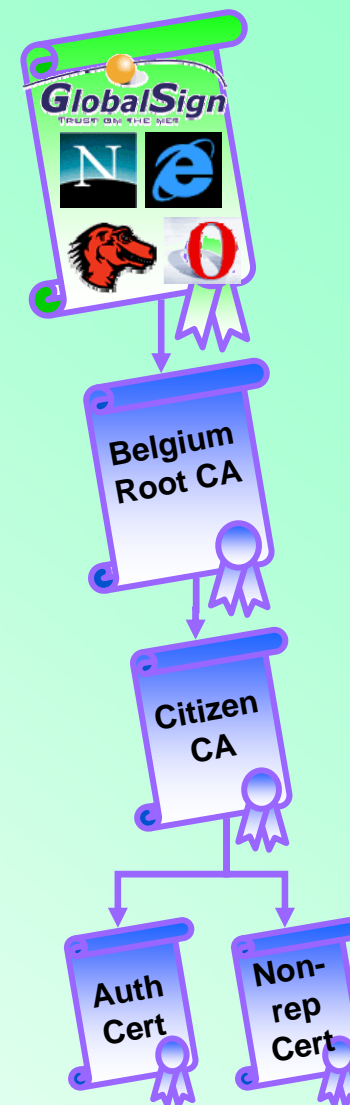


# eID Certificates Hierarchy



# Certificates – Linking public keys to entities

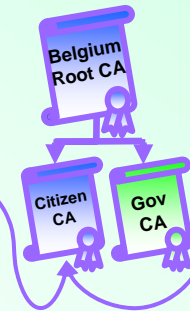
- How does Bob know that a *public* key belongs to Alice?
- Belgian government issues a statement “this *public* key belongs to Alice”
  - Statement is called a “certificate”
  - One certificate per key pair
  - *Private* key only known to certified entity



# Citizen Certificate Details

## Citizen Qualified certificate (~1000 bytes)

Version: 3 (0x2)  
Serial Number:  
10:00:00:00:00:00:8d:8a:fa:33:d3:08:f1:7a:35:b2  
Signature Algorithm: sha1WithRSAEncryption (1024 bit)  
Issuer: C=BE, CN=Citizen CA, SN=200501  
Not valid before: Apr 2 22:41:00 2005 GMT  
Not valid after: Apr 2 22:41:00 2010 GMT  
Subject: C=BE, CN=Sophie Dupont (Signature),  
SN=Dupont, GN=Sophie  
Nicole/serialNumber=60050100093  
Subject Public Key Info:  
RSA Public Key: [Modulus (1024 bit): 4b:e5:7e:6e: ... :86:17,  
Exponent: 65537 (0x10001)]  
X509v3 extensions:  
Certificate Policies:  
Policy: 2.16.56.1.1.1.2.1  
CPS: <http://repository.eid.belgium.be>  
Key Usage: critical, Non Repudiation  
Authority Key Identifier: [D1:13: ... :7F:AF:10]  
CRL Distribution Points:  
URI:<http://crl.eid.belgium.be/eidc0002.crl>  
Netscape Cert Type: S/MIME  
Authority Information Access:  
CA Issuers - URI:<http://certs.eid.belgium.be/belgiumrs.crt>  
OCSP - URI:<http://ocsp.eid.belgium.be>  
Qualified certificate statements: [00.....F..]  
Signature: [74:ae:10: ... :e0:91]



## Citizen Authentication certificate (~980 bytes)

Version: 3 (0x2)  
Serial Number:  
10:00:00:00:00:00:0a:5d:9a:91:b1:21:dd:00:a2:7a  
Signature Algorithm: sha1WithRSAEncryption (1024 bit)  
Issuer: C=BE, CN=Citizen CA, SN=200501  
Not valid before: Apr 2 22:40:52 2005 GMT  
Not valid after: Apr 2 22:40:52 2010 GMT  
Subject: C=BE, CN=Sophie Dupont (Authentication),  
SN=Dupont, GN=Sophie  
Nicole/serialNumber=60050100093  
Subject Public Key Info:  
RSA Public Key: [Modulus (1024 bit): cf:ca:7a:77: ... :5c:c5,  
Exponent: 65537 (0x10001)]  
X509v3 extensions:  
Certificate Policies:  
Policy: 2.16.56.1.1.1.2.2  
CPS: <http://repository.eid.belgium.be>  
Key Usage: critical, Digital Signature  
Authority Key Identifier: [D1:13: ... 7F:AF:10]  
CRL Distribution Points:  
URI:<http://crl.eid.belgium.be/eidc0002.crl>  
Netscape Cert Type: SSL Client, S/MIME  
Authority Information Access:  
CA Issuers - URI:<http://certs.eid.belgium.be/belgiumrs.crt>  
OCSP - URI:<http://ocsp.eid.belgium.be>  
Signature: [10:ac:04: ... :e9:04]

# Signing Keys & Certificates

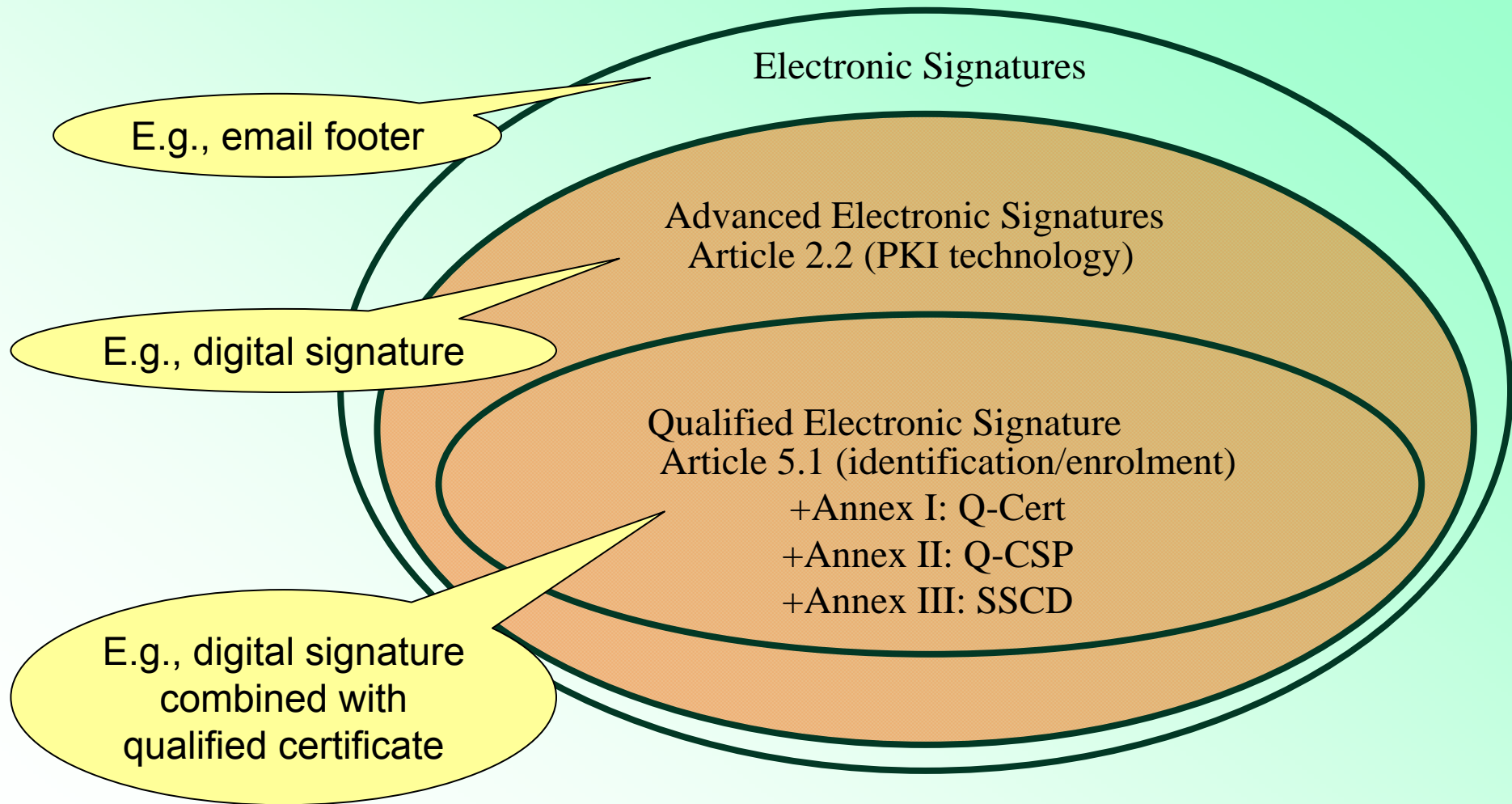
- 2 key pairs for the citizen:
  - Citizen-authentication
    - X.509v3 **authentication certificate**
  - Advanced electronic (non-repudiation) signature
    - X.509v3 **qualified certificate**
    - Can be used to produce digital signatures equivalent to handwritten signatures, cfr. European Directive 1999/93/EC
- 1 key pair for the card:
  - eID card authentication (basic key pair)
    - **No corresponding certificate**: RRN (Rijksregister/Registre National) knows which public key corresponds to which eID card



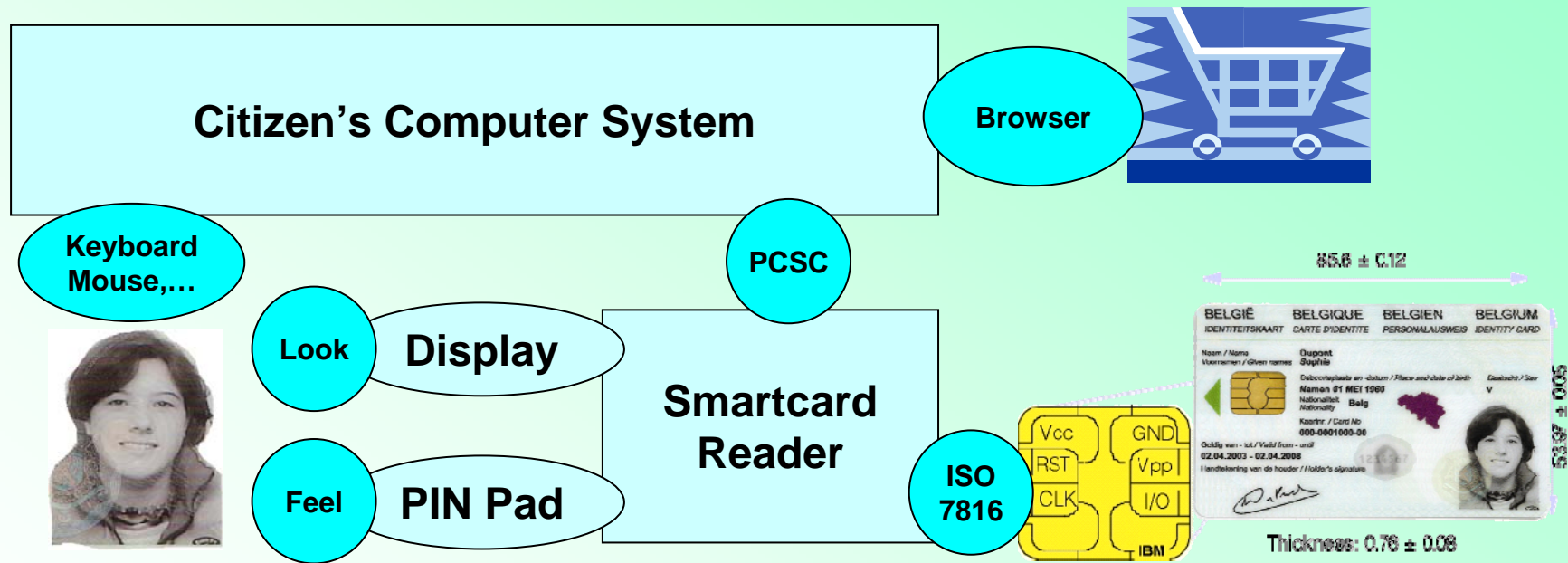
# Future Versions

- eID card's validity will change from 5 to 10 years
  - Citizens will be issued with 2048-bit RSA key pairs
    - Used to be 1024-bit RSA
  - Cards will be used more
    - More prone to physical damage to the card & chip
- Migration with SIS card
  - No migration with SIS card
  - SIS card data fetched online
  - eID card = identification token

# Signature Types – EU Directive 1999/93/EC

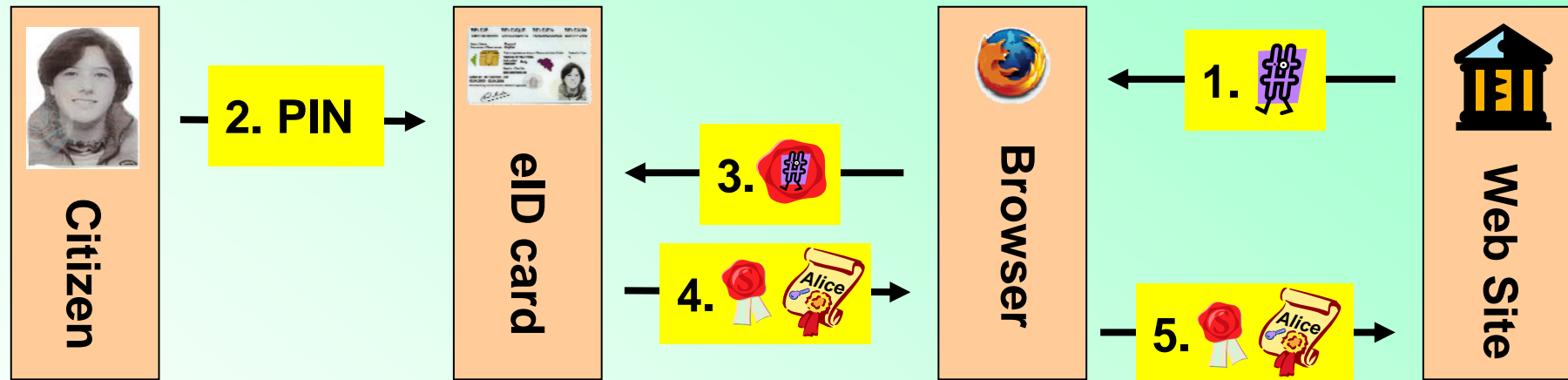


# Typical Smartcard Architecture



# Using an Authentication Certificate

Case study: Alice visits a website which uses client authentication

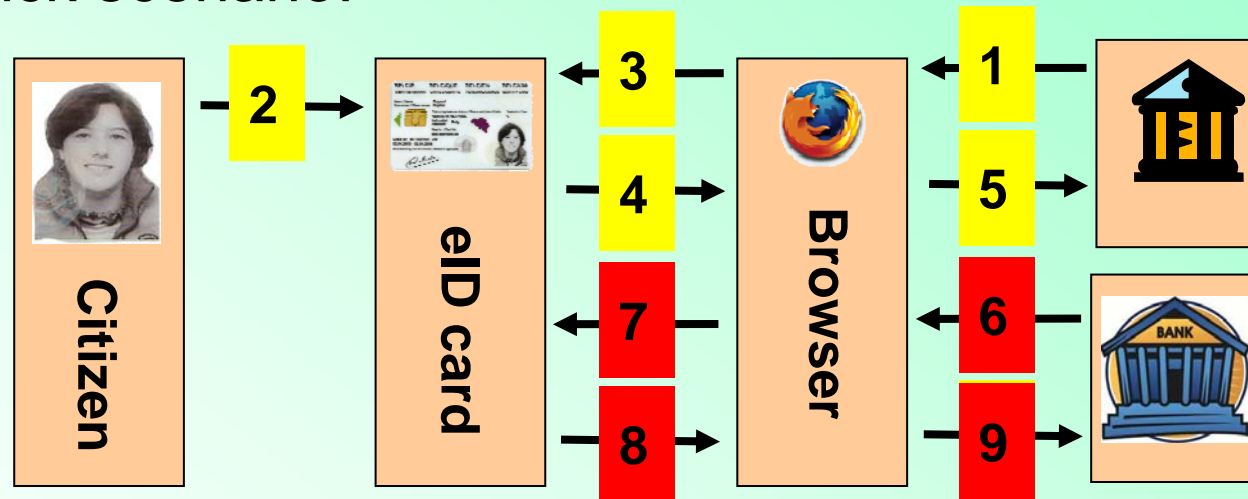


1. The web server Alice visits sends a random challenge to her browser
2. Alice confirms she wants to log in on the web site by presenting her PIN to her eID card and authorizes the signature generation

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

# Preventing Hijacking eID Authentication Session

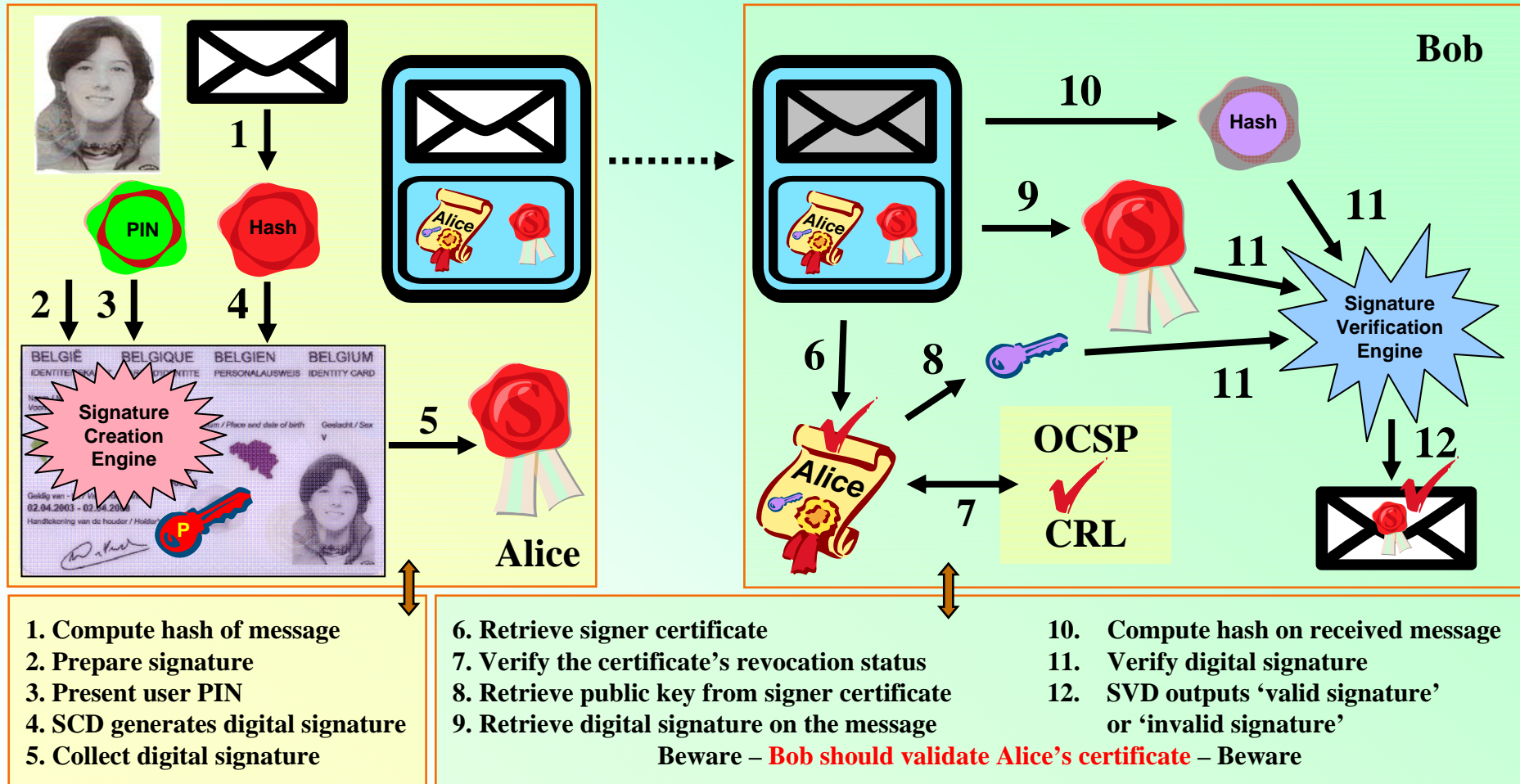
- Attack scenario:



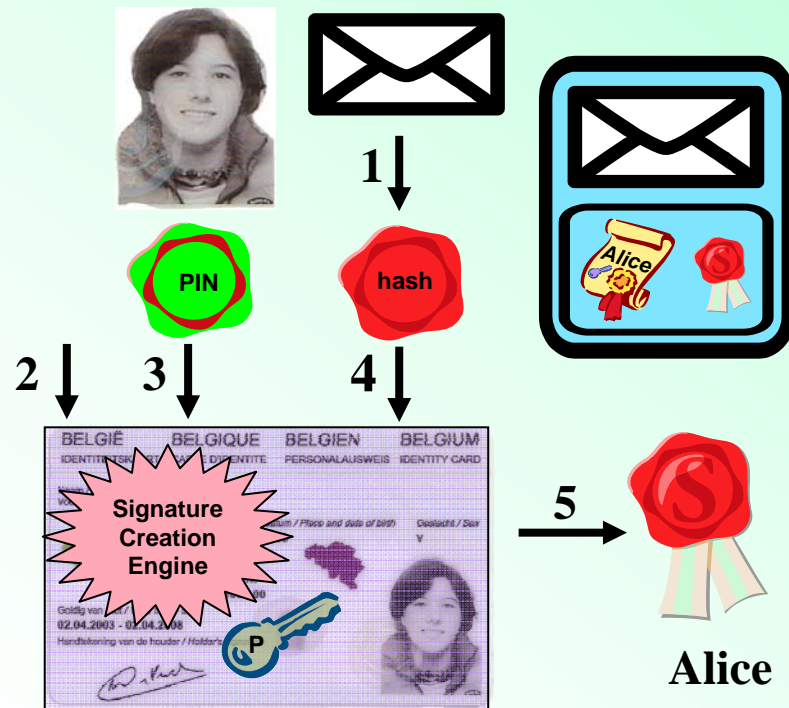
- Use firewall-like smartcard reader:

- Filters potentially malicious smartcard commands
- Provides sensible feedback to citizen
- Citizen provides consent to send command to her smartcard

# Signature Generation/Verification



# Signature Generation Steps



## Alice's application

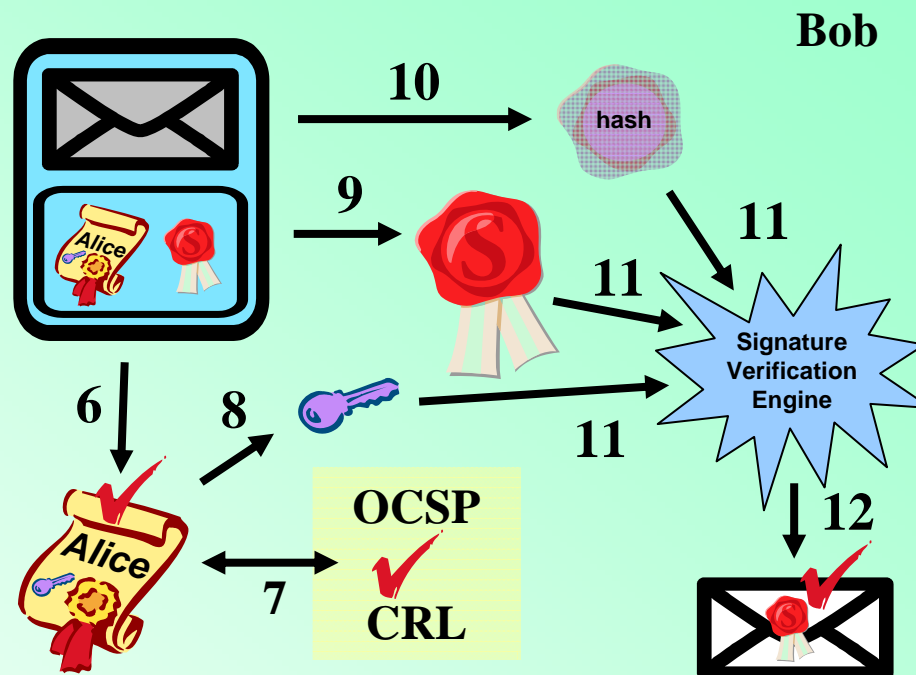
1. Calculates the cryptographic hash on the data to be signed
2. Prepares her eID card to generate an authentication signature or to generate a non-repudiation signature
3. Alice presents her PIN to her eID card
4. Her card generates the digital signature on the cryptographic hash
5. The application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

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

# Signature Verification Steps

Bob

6. Retrieves the potential sender's certificate
7. Verifies the certificate's revocation status
8. Extracts Alice's public key from her certificate
9. Retrieves the signature from the message
10. Calculates the hash on the received message
11. Verifies the digital signature with the public key and the hash
12. 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*



# Certificate & Signature Validity

# Signature Validation

- A digital signature protects the integrity of information
- A digital signature computed on some data is valid if and only if
  - The signature verification engine confirms that the **hash value** computed on the data **matches the digital signature** when applying the signature verification mechanism using the **public key** found in the corresponding certificate
  - The **certificate is valid** (cfr. next slide)
  - All the **key usage and certificate policies** of the certificates in the certificate chain match the context wherein the data is used (e.g., code signing, client authentication, server authentication,...)
- **Caveat:**
  - When was this signature computed?
- **Revoked ≠ Invalid**
  - Keep a log of valid signatures
- **Hash function features:**
  - Given a hash value of a document: hard to find a document with that hash value
  - Given a document and its hash value: hard to find a second document with the same hash value
  - Hard to find two distinct documents that have an identical hash value



Message

Data



Hash value



Digital signature



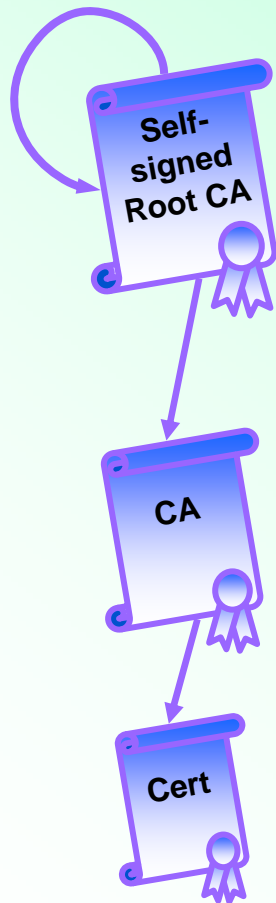
Public key



Signer certificate



# Certificate (Chain) Validation



- A certificate protects the identity of the holder of the corresponding private key
- Given a self-signed certificate Root CA protects the CA certificate which is used to validate a non-CA certificate
- A certificate Cert is valid if and only if
  - The **certificate's digital signature** is (cryptographically) valid given the certificate issuer's certificate (CA certificate)
  - The **certificate issuer's certificate** is valid (using that certificate's issuer certificate. This may be the same certificate if self-signed)
  - The time of certificate validation lies within the **validity period** of all these certificates
  - All certificate extensions must match the respective **profiles and key usages**
  - None of these certificates is known as invalid, i.e.,
    - Their **serial numbers have not been revoked**
- Check the revocation status of a certificate using CRLs or OCSP
  - Depending on the **required security level**, one may decide to rely on the OCSP, or on a local CRL copy, or on a local CRL copy in combination with a recent Delta CRL
  - Offline validation is possible using CRL, preferably combined with Delta CRL
  - OCSP (Online Certificate Status Protocol) requires a live network connection
- Certificate chain is linked with the CRLs through the **Authority Key Identifier**
- Valid ≠ Trustworthy
  - One should check whether the self-signed (Root CA) certificate can be trusted

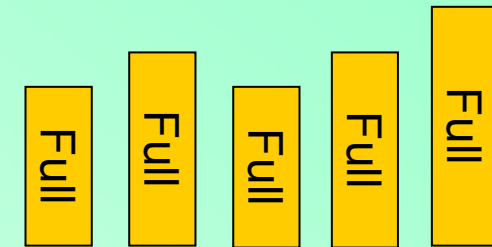
# Certificate Revocation Lists (CRLs)

## ■ Complete CRL

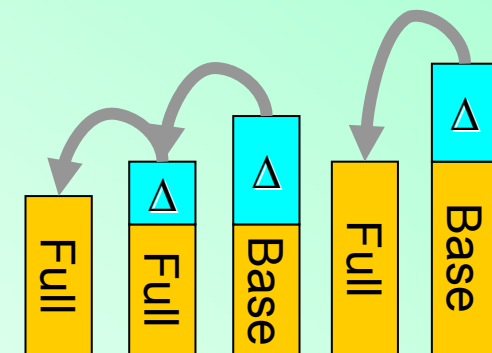
- Enumerates all certificate serial numbers that should not be trusted
- Typically (very) large, e.g., >500 Kbytes
- “NextUpdate” 7 days after creation
- Certificates of new eID cards
  - Appear as on hold
  - Disappear when activated
- Suspended certificates appear as on hold for up to 7 days
- Items without reason code remain revoked forever
- One complete CRL is referred to as the Base CRL

## ■ Delta CRL in theory, Delta-Delta CRLs in practice

- Lists all differences between the current complete CRL and the current Base CRL
- Typically small, e.g., <25 Kbytes
- “NextUpdate” 7 days after creation
- Reason codes:
  - On hold — newly issued eID card certificate is not yet activated, or has been suspended
  - Remove from CRL — eID card certificate has been activated
  - None — eID card certificate has been revoked

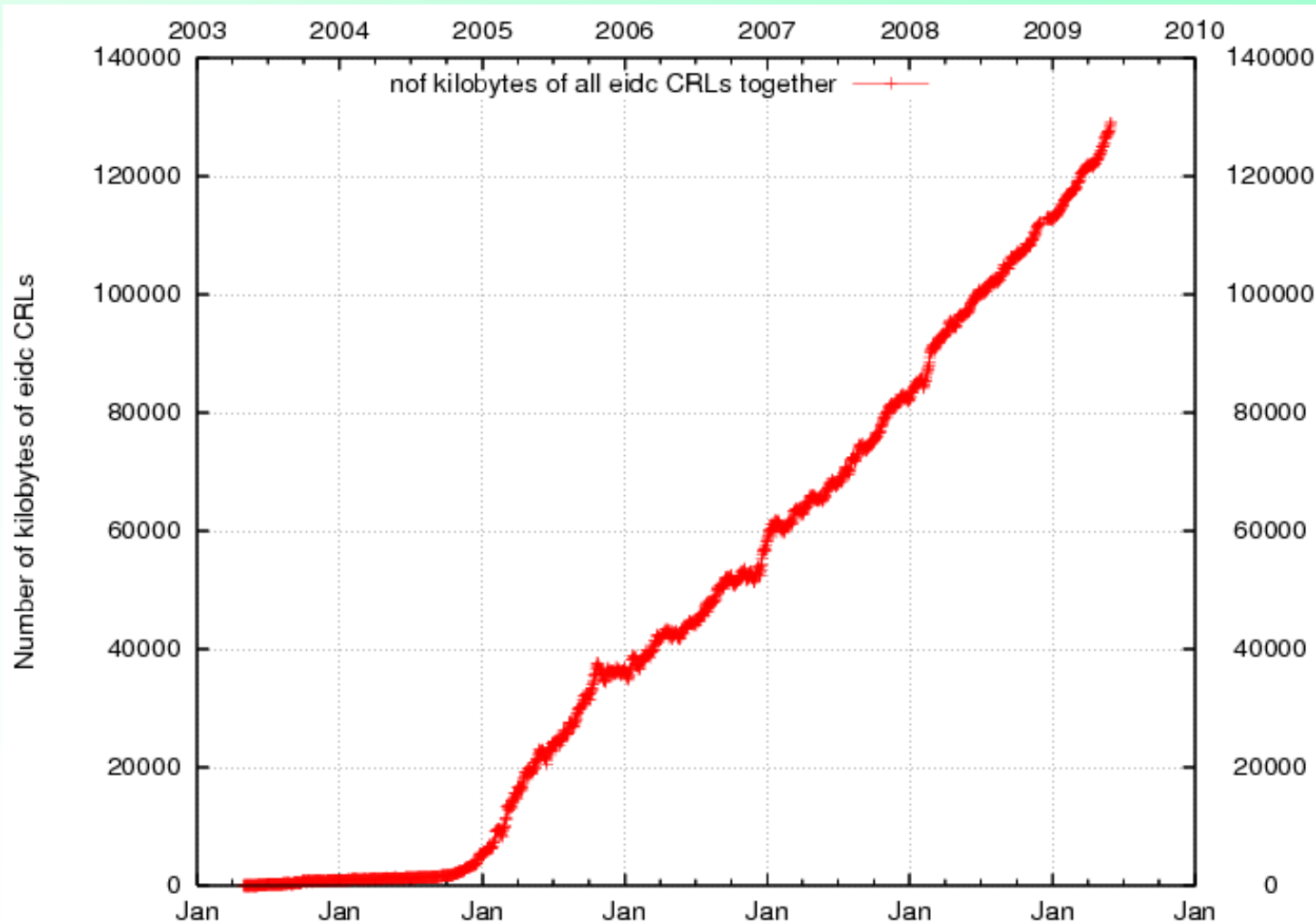


Complete CRLs



Delta CRLs vs. Base CRL

# Current eID full CRL sizes



- A CRL is valid for seven days after it is issued
- A new CRL is issued together with a new Delta CRL
- A Delta CRL refers to a particular Base CRL which is always younger than 7 days
- OCSP queries the database with the most recent certificate status information
- OCSP = Online Certificate Status Protocol

# OCSP vs. CRLs – “Is this certificate valid?”

- Two options to make this **business decision**:
  - Do it yourself and use CRLs and Delta-CRLs
  - Trust a third party and use OCSP
- Use the Online Certificate Status Protocol (OCSP) where a trusted OCSP Responder answers the question with either “yes”, “no”, or “I do not know”
  - Remaining issues:
    - An OCSP Responder **may** use the most recent certificate status information (CSI)
      - An OCSP Responder does not have to use the most recent CSI!
      - The Responder typically uses CRLs to produce its answers
    - How to trust the OCSP Response?
  - Ideal for a few situations:
    - If only a few certificates per time unit must be validated
      - E.g., for citizens who wish to validate a certificate “from time to time”
    - To authenticate high-impact transactions
      - E.g., cash withdrawal, account closure, physical or electronic access control
- Certificate Revocation Lists (CRLs)
  - The digital signature verifier collects the (most recent) CRLs for the certificates in the certificate chain
    - These CRLs may become extremely large (e.g., several megabytes) ⇒ Delta-CRLs
    - Delta-CRLs may be very large (e.g., half a megabyte) ⇒ Delta-Delta CRLs
      - Note: Delta-Delta-CRLs are typically a few kilobytes each, but there is no standard...

# Summary on Validity Statuses

## ■ Digital Signature

- Valid
- Invalid



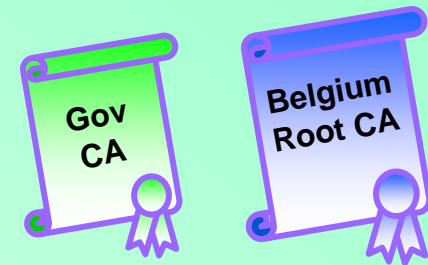
## ■ eID Card (Signature Creation Device)

- Valid
- Invalid
  - Suspended
  - Revoked
  - Expired



## ■ CRL, OCSP Response

- Valid
- Invalid
- Expired

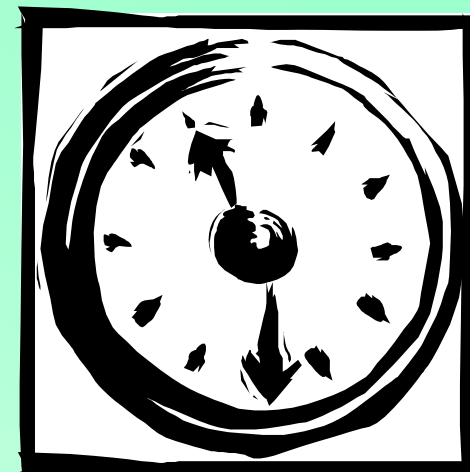


## ■ Certificate

- Valid
- Invalid
  - Suspended
  - Revoked
  - Expired
- Unknown



# Questions?



Belgian eID card information on the Internet

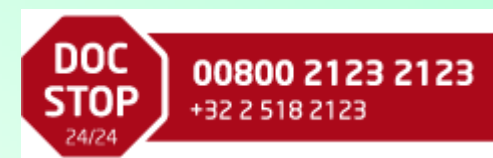
<http://eid.belgium.be>  
<http://www.ibz.rrn.fgov.be/>  
<http://www.fedict.be>  
<http://www.belgium.be>  
<http://www.cardreaders.be>

Middleware available from

<http://code.google.com/p/eid-mw/>

Myself Danny.DeCock@esat.kuleuven.be  
<http://godot.be>

Yourself <https://www.mijndossier.rrn.fgov.be>  
<https://www.mondossier.rrn.fgov.be>  
<https://www.meindossier.rrn.fgov.be>





# Backup Slides

# Signing Key Pair Properties

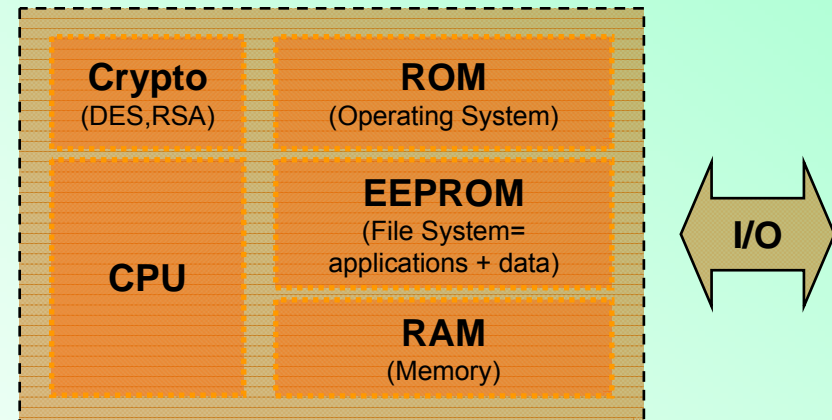
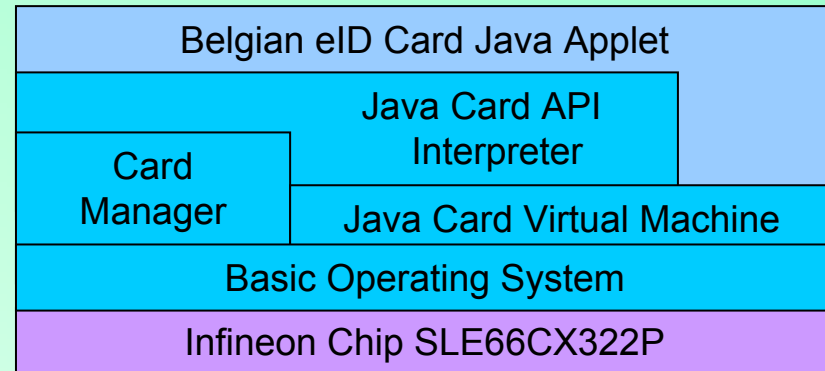
- Private signing key only available to the signer
  - Signer explicitly authorizes the Signature Creation Engine to generate a digital signature with the signing key, e.g., by presenting a PIN (personal identification number, cfr. Bank cards)
  - Signer protects the hash of his/her message with his/her signing key
  - Verifier recovers this hash correctly only if the right verification key is used
- Private signing key corresponds to the public verification key
  - If the Signature Verification Engine (SVE) outputs 'valid signature', the verification key corresponds to the signing key
  - If the SVE outputs 'invalid signature' the triplet (message, digital signature, verification key) does not match:
    - The message may have been *altered*
    - The *verification key may be wrong*, i.e., does not correspond to the signing key
    - The *certificate* of the signer *may have been revoked* (or *suspended*)
- Private signing key is kept in the smartcard
- Public verification key usually accompanies the digital signature
  - Integrity of the verification key is protected through the signer's certificate

# The Belgian eID card...

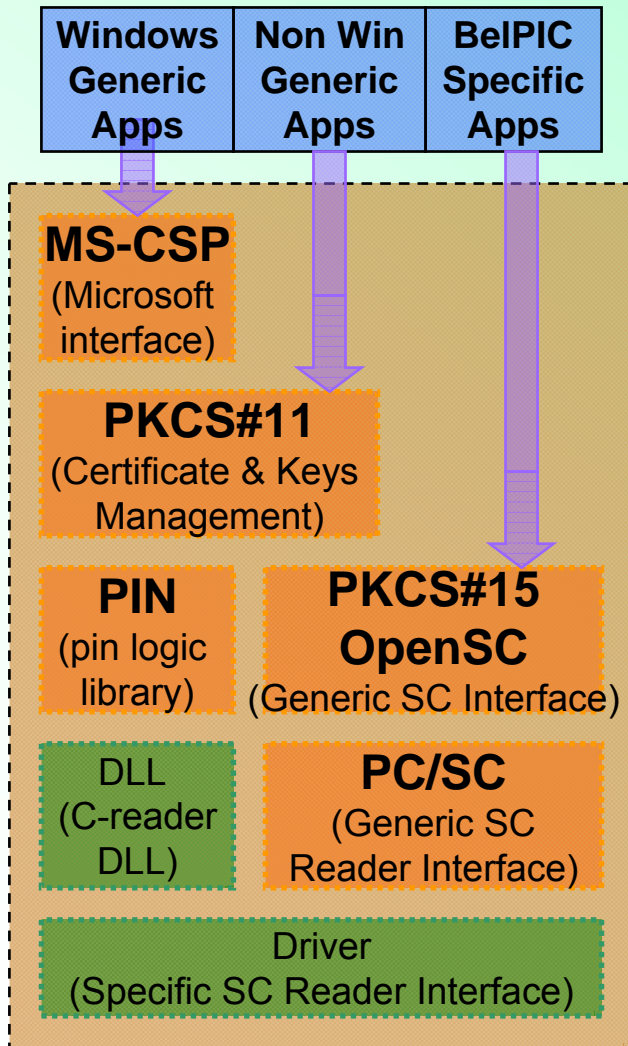
- Uses On-board key pair generation
  - Private keys cannot leave the eID card
  - Key pair generation is activated during the initialization of the eID card
- Uses JavaCard technology
- Can be used using software/middleware – free of charge – provided the Government
- Can only be managed by the Belgian government
  - Citizen identity/address data is read/write for the National Registry
  - eID card refuses update attempts from other parties than the government

# 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)



# eID Card Middleware



- PKCS#15 file system for ID applications
  - All eID-related data (certificates, photo, address, identity files,...)
  - No key management
- PKCS#11 standard interface to crypto tokens
  - Abstraction of signing functions (authentication, digital signatures)
  - Access to certificates
  - Available for Unix, Windows, MacOSX,...
- CSP for Microsoft Platforms
  - Only keys & certificates available via MSCrypto API
  - Allows authentication (& signature)
  - For Microsoft Explorer, Outlook,...



# eID Card Administration

- Mutual authentication of the card and the external party
  - Role-based access control
- Supported roles:
  - 1: delete/create files: data, keys, certificates
  - 2: create files: data, keys, certificates
  - 3: generate new key pairs
  - 4: store new citizen certificates
  - 5: store new Root CA certificate
  - 7: update citizen address or identity file
  - 8: store the Role-CA's new public key

# Comparing eID and Bank Card Functionalities



- Citizen Identification
- Data Capture
- Strong Authentication
  - Authentication
  - Digital Signatures
  - eID Card
- Access Control
  - Container Park, Swimming Pool, Library,...

- Customer Identification
- Data Capture
- Authentication
  - Electronic Transactions
  - ATM Transactions
  - Electronic Purse
- Access Control
  - Self-Bank

# eID & Bank Cards Crypto

- 2 Citizen Key Pairs
  - Citizen-authentication
    - X.509v3 authentication certificate
  - Advanced electronic (non-repudiation) signature
    - X.509v3 qualified certificate
    - Can be used to produce digital signatures equivalent to handwritten signatures, cfr. European Directive 1999/93/EC
- 1 eID Card-specific Key Pair
  - eID card authentication (basic key pair)
    - No corresponding certificate: RRN (Rijksregister/Registre National) knows which public key corresponds to which eID card
- Transactions with vending machines, ATMs, phone booths, parking meters, ...
  - MAC-based use chip card
- Home banking
  - MAC-based
    - Family of secret master keys
    - Uses chip card or Digipass
    - MAC authenticates login, transaction
  - PKI-based
    - Closed user group PKI
    - Key pair stored in key file or smart card
    - Banking organization issues certificate
    - Digital signature authenticates login, transaction

# CA Certificate Details

## Root CA certificate (920 bytes)

Version: 3 (0x2)  
Serial Number:  
58:0b:05:6c:53:24:db:b2:50:57:18:5f:f9:e5:a6:50  
Signature Algorithm: sha1WithRSAEncryption (2048 bit)  
Issuer: C=BE, CN=Belgium Root CA  
Not valid before: Jan 26 23:00:00 2003 GMT  
Not valid after : Jan 26 23:00:00 2014 GMT  
Subject: C=BE, CN=Belgium Root CA

Subject Public Key Info:  
RSA Public Key: [Modulus (2048 bit): 00:c8:a1:71: ... :b0:6f,  
Exponent: 65537 (0x10001)]

X509v3 extensions:  
Certificate Policies:  
Policy: 2.16.56.1.1.1  
CPS: <http://repository.eid.belgium.be>  
Key Usage: critical, Certificate Sign, CRL Sign  
Subject Key Identifier: [10:F0: ... :8E:DB:E6]  
Authority Key Identifier: [10:F0: ... :8E:DB:E6]

Netscape Cert Type: SSL CA, S/MIME CA, Object Signing CA  
Basic Constraints: critical, CA:TRUE

Signature: [c8:6d:22: ... :43:2a]

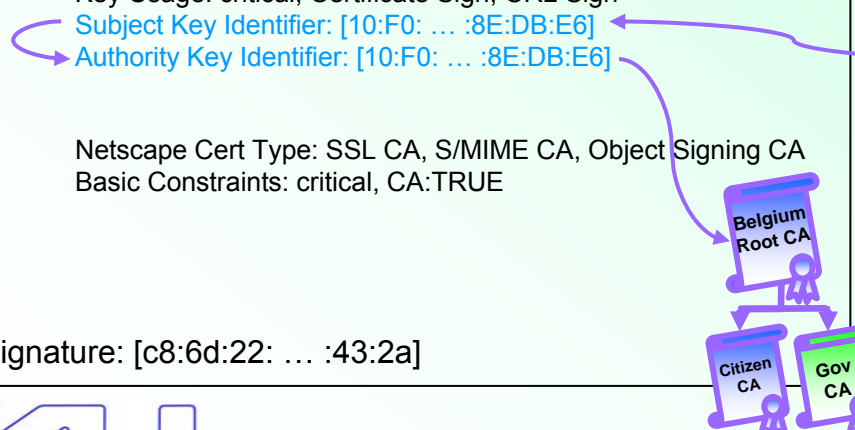
## CA certificate (975 bytes)

Version: 3 (0x2)  
Serial Number:  
6f:77:79:33:30:25:e3:cf:92:55:b9:7a:8a:0b:30:e5  
Signature Algorithm: sha1WithRSAEncryption (2048 bit)  
Issuer: C=BE, CN=Belgium Root CA  
Not valid before: Apr 10 12:00:00 2003 GMT  
Not valid after : Jun 26 23:00:00 2009 GMT  
Subject: C=BE, CN=Citizen CA

Subject Public Key Info:  
RSA Public Key: [Modulus (2048 bit): 00:c9:ae:05: ... :cb:71,  
Exponent: 65537 (0x10001)]

X509v3 extensions:  
Certificate Policies:  
Policy: 2.16.56.1.1.1.2  
CPS: <http://repository.eid.belgium.be>  
Key Usage: critical, Certificate Sign, CRL Sign  
Subject Key Identifier: [D1:13: ... :7F:AF:10]  
Authority Key Identifier: [10:F0: ... :8E:DB:E6]  
CRL Distribution Points:  
URI:<http://crl.eid.belgium.be/belgium.crl>  
Netscape Cert Type: SSL CA, S/MIME CA, Object Signing CA  
Basic Constraints: critical, CA:TRUE, pathlen:0

Signature: [b2:0c:30: ... :18:6e]



# Government Certificate Details

## Government CA certificate (~979 bytes)

Version: 3 (0x2)  
Serial Number:  
99:6f:14:78:8e:ea:69:6a:3d:2e:93:42:81:2b:66:f0  
Signature Algorithm: sha1WithRSAEncryption (2048 bit)  
Issuer: C=BE, CN=Belgium Root CA  
Not valid before: Jan 27 00:00:00 2003 GMT  
Not valid after: Jan 27 00:00:00 2009 GMT  
Subject: C=BE, CN=Government CA

### Subject Public Key Info:

RSA Public Key: [Modulus (2048 bit): 00:ac:c9:a0: ... :89:13,  
Exponent: 65537 (0x10001)]

### X509v3 extensions:

#### Certificate Policies:

Policy: 2.16.56.1.1.1.3

CPS: <http://repository.eid.belgium.be>

Key Usage: critical, Certificate Sign, CRL Sign

Subject Key Identifier: [F5:DB: ... :D1:8B:D6]

Authority Key Identifier: [10:F0: ... :8E:DB:E6]

#### CRL Distribution Points:

URI: <http://crl.eid.belgium.be/belgium.crl>

Netscape Cert Type: SSL CA, S/MIME CA, Object Signing CA

Basic Constraints: critical, CA:TRUE, pathlen:0

Signature: [a0:53:21: ... :1d:c9]

## RRN certificate (~808 bytes)

Version: 3 (0x2)  
Serial Number:  
01:00:00:00:00:00:f8:20:18:9e:17  
Signature Algorithm: sha1WithRSAEncryption (1024 bit)  
Issuer: C=BE, CN=Government CA  
Not valid before: Oct 9 09:06:09 2003 GMT  
Not valid after: Jan 26 09:06:09 2009 GMT  
Subject: C=BE, CN=RRN, O=RRN

### Subject Public Key Info:

RSA Public Key: [Modulus (1024 bit): 00:db:72:4d: ... :80:0d,  
Exponent: 65537 (0x10001)]

### X509v3 extensions:

#### Certificate Policies:

Policy: 2.16.56.1.1.1.3.1

CPS: <http://repository.eid.belgium.be>

Key Usage: critical, Digital Signature, Non Repudiation

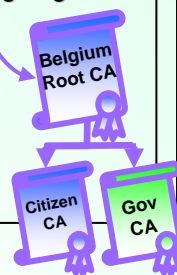
Subject Key Identifier: [09:22: ... :30:01:37]

Authority Key Identifier: [F5:DB: ... :D1:8B:D6]

#### CRL Distribution Points:

URI: <http://crl.eid.belgium.be/government.crl>

Signature: [12:89:cd: ... :ca:2a]



# Certificate Revocation List details

## Citizen CRL (+500 Kbyte)

Version 2 (0x1)  
 Signature Algorithm: sha1WithRSAEncryption (2048 bit)  
 Issuer: C=BE, CN=Citizen CA  
 Creation date: Apr 6 15:19:23 2004 GMT  
 Next update: Apr 13 15:19:23 2004 GMT  
 CRL extensions:

Authority Key Identifier: [D1:13: ... :7F:AF:10]  
 CRL Number: 4294995040

### Revoked Certificates:

Serial Number: 1000000000000004B823FAE7B1BB44B1  
 Revocation Date: Jan 14 12:56:50 2004 GMT  
 CRL Reason Code: Certificate Hold

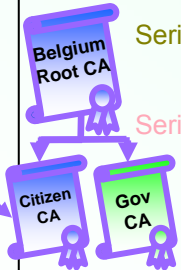
Serial Number: 10000000000000062F6A1BB1431902D4  
 Revocation Date: Oct 23 23:15:11 2003 GMT  
 CRL Reason Code: Certificate Hold

Serial Number: 100000000000001243778BEFF61123DE  
 Revocation Date: Jan 12 10:19:24 2004 GMT

Serial Number: 10000000000000125DC2DF2031534033  
 Revocation Date: Sep 5 09:49:44 2003 GMT

Serial Number: 1000000000000091ACC84FC377F8A6ECE  
 Revocation Date: Dec 16 17:24:15 2003 GMT  
 CRL Reason Code: Certificate Hold

Serial Number: 1000000000000092135CE8FB8F0D66093  
 Revocation Date: Nov 13 17:18:49 2003 GMT



Signature: [95:19:b2: ... :21:31]

## Citizen Delta CRL (~15 Kbyte)

Version 2 (0x1)  
 Signature Algorithm: sha1WithRSAEncryption (2048 bit)  
 Issuer: C=BE, CN=Citizen CA  
 Creation date: Apr 8 17:43:14 2004 GMT  
 Next update: Apr 15 17:43:14 2004 GMT  
 CRL extensions:

Authority Key Identifier: [D1:13: ... :7F:AF:10]  
 CRL Number: 4294995072  
 Delta CRL Indicator: critical, 4294995040

### Revoked Certificates:

Serial Number: 100000000000007E5B11506303959320  
 Revocation Date: Apr 8 16:33:23 2004 GMT  
 CRL Reason Code: Certificate Hold

Serial Number: 1000000000000091ACC84FC377F8A6ECE  
 Revocation Date: Apr 8 16:55:14 2004 GMT  
 CRL Reason Code: Remove From CRL

Serial Number: 100000000000127BE2DA18842E8A7BAC  
 Revocation Date: Apr 8 15:20:13 2004 GMT  
 CRL Reason Code: Remove From CRL

Serial Number: 1000000000001902ECF11657FE2813A5  
 Revocation Date: Apr 8 16:29:54 2004 GMT

Serial Number: 100000000000FDFF72C4E59AD46AFC21  
 Revocation Date: Apr 8 17:33:31 2004 GMT  
 CRL Reason Code: Remove From CRL

Serial Number: 100000000000FE6A4ACD4ECF04233442  
 Revocation Date: Apr 8 15:32:38 2004 GMT

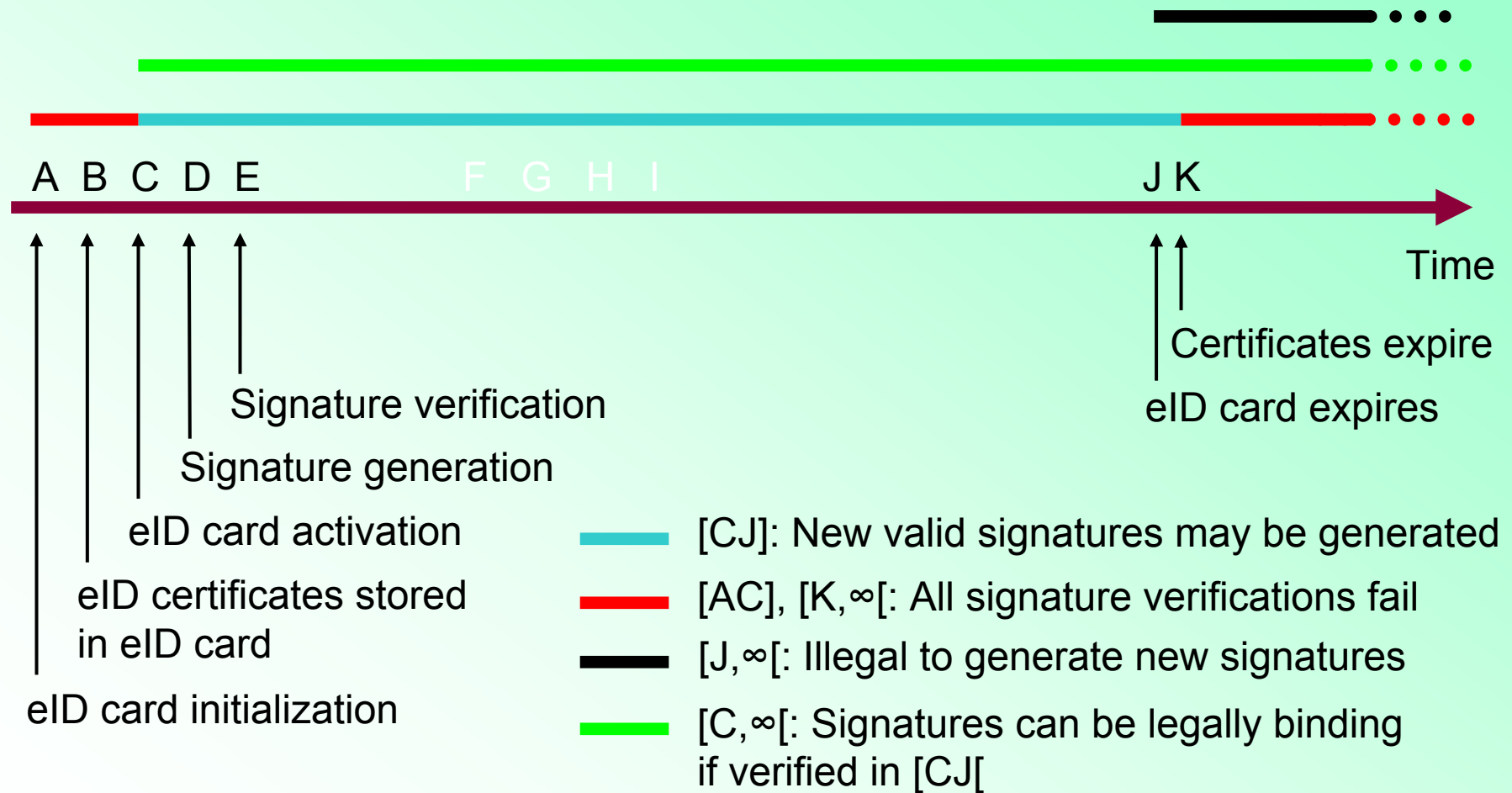
...

Signature: [64:20:22: ... :c3:5e]

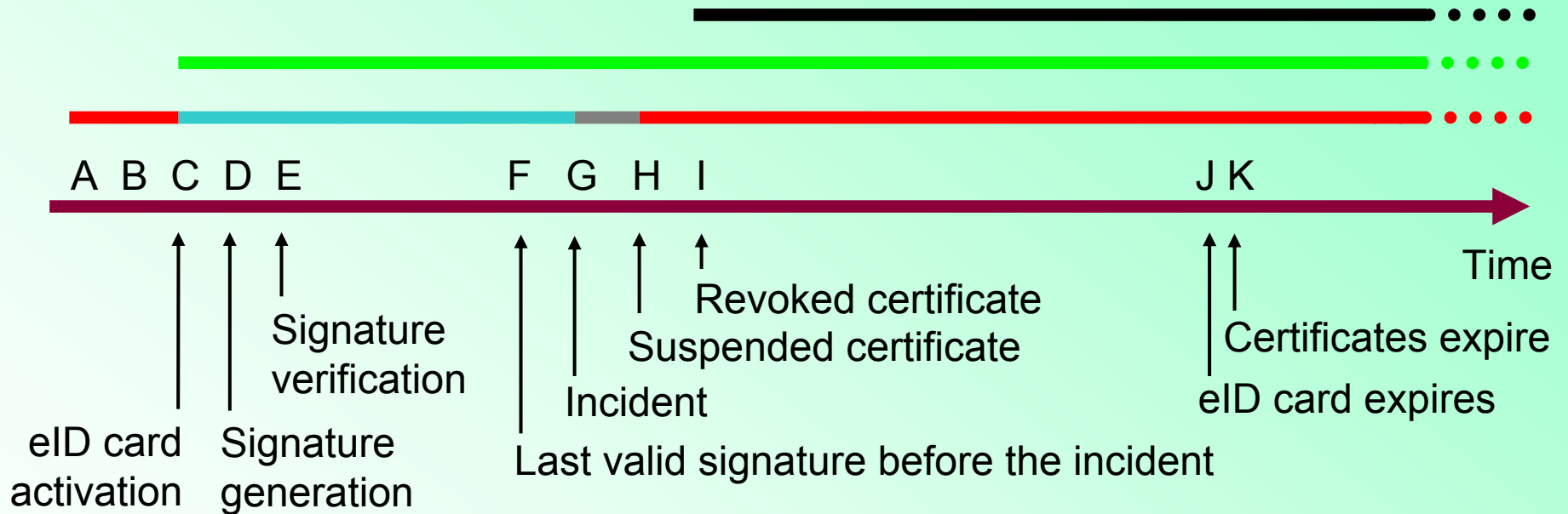


# Signature Validity Over Time

# Signature Validity



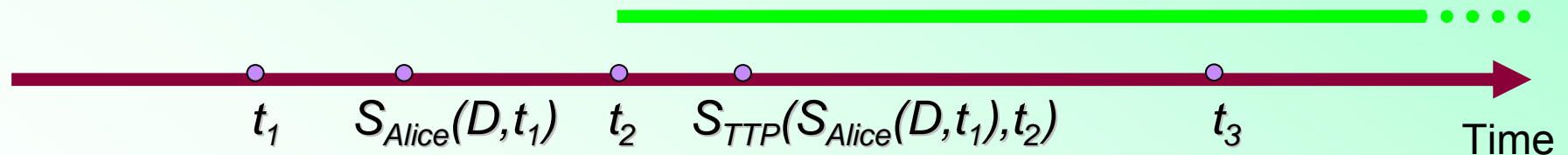
# Signature Validity with Revocation



- [GH]: Signatures created in [GI] should be invalid, H may be equal to I
- [I, ∞[: Illegal to generate new signatures
- [CG[: New valid signatures may be generated
- [AC], [H, ∞[: Signature verification returns invalid
- [CF]: Signatures validated before F may be valid forever

# Long Term Signatures

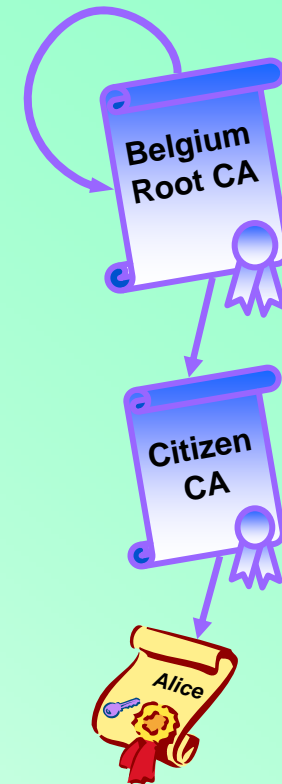
- Alice produces a digital signature on data  $D$  that will resist time:
  - Alice collects a time stamp  $t_1$  from a trusted third party ( $TTP$ )
  - Alice produces a digital signature  $S_{Alice}(D, t_1)$  on the time stamp  $t_1$  and the data  $D$
  - $TTP$  validates a digital signature  $S_{Alice}(D, t_1)$  at time  $t_2$
  - $TTP$  computes a digital signature  $S_{TTP}(S_{Alice}(D, t_1), t_2)$  if and only if the  $TTP$ 
    - Has validated Alice's digital signature, and
    - Confirms that the signature and Alice's full certificate chain was valid at time  $t_2$
  - Alice can now indefinitely rely on  $S_{TTP}(S_{Alice}(D, t_1), t_2)$ , even if her certificate must be revoked, e.g., at time  $t_3$  (after  $t_2$ ), or if her certificate expires



- Note: This procedure assumes that no cryptographic weaknesses are discovered in the signature generation and validation algorithms and procedures

# Archiving Signed Data

- Digital signatures *remain valid forever* if one stores:
  - The digitally **signed data**
  - The **digital signature** on the data
  - The **signer's certificate**
  - A **proof of validity** of the signer's certificate
  - The **verification timestamp** of the signature
- Bottom line:
  - The integrity of this data should be protected!
  - There is no need to retrieve the status of a certificate in the past!
  - Protect your **proofs in a digital vault**





# European Directive 1999/93/EC

# European Directive 1999/93/EC

- Intention
- Definitions
- Requirements
  - Annex I — qualified certificates
  - Annex II — certificate service provider
  - Annex III — secure signature creation device
- Recommendations
  - Annex IV — signature verification

# Directive – Intention

1. An **advanced electronic signature** (i.e., a signature which is linked to (s)he who created it using a signature creation device which only (s)he can control) **satisfies the legal requirements** of a signature in relation to data in electronic form in the **same manner as a handwritten signature** satisfies those requirements in relation to paper-based data; and is admissible as evidence in legal proceedings
  - ⇒ Legislation on handwritten signatures can easily be recycled!!
2. An **electronic signature** is not **denied legal effectiveness** and admissibility as evidence in legal proceedings solely on the grounds that it is:
  1. in electronic form, or
  2. not based upon a qualified certificate, or
  3. not based upon a qualified certificate issued by an accredited certification-service-provider, or
  4. not created by a secure signature-creation device

# Directive – Definitions

- **Electronic signature**: data in electronic form attached to or logically associated with other electronic data and which serve as a method of authentication
- **Advanced electronic signature**: an electronic signature which meets the requirements that
  1. it is uniquely linked to the signatory
  2. it is capable of identifying the signatory
  3. it is created using *means that the signatory can maintain under his sole control*, and
  4. it is linked to the data to which it relates in such a manner that any subsequent change of the data is detectable
- **Signatory**: a person who holds a signature-creation device and acts either on his own behalf or on behalf of the natural or legal person or entity he represents
- **Signature-creation data**: unique data, such as private cryptographic keys, which are used by the signatory to create an electronic signature
- **Signature-creation device**: configured software or hardware to produce the signature-creation data
- **Secure-signature-creation device**: a signature-creation device which meets the requirements specified in Annex III
- **Signature-verification-data**: data, such as public cryptographic keys, which are used for the verification of an electronic signature
- **Certificate**: an electronic attestation which links signature-verification data to a person and confirms the identity of that person
- **Qualified certificate**: a certificate which meets the requirements in Annex I and is provided by a certification-service-provider who fulfils the requirements in Annex II
- **Certification-service-provider**: an entity or a legal or natural person who issues certificates or provides other services related to electronic signatures

# Annex I – Qualified Certificates Conditions

## Requirements for qualified certificates

### ■ Qualified certificates must contain:

1. an **indication** that the certificate is issued as a qualified certificate
2. the identification of the **certification-service-provider and the State** in which it is established
3. the **name** of the signatory **or a pseudonym**, which shall be identified as such
4. provision for a specific attribute of the signatory to be included if relevant, depending on the **purpose for which the certificate is intended**
5. signature-verification data which correspond to signature-creation data under the control of the signatory
6. an indication of the beginning and end of the **period of validity of the certificate**
7. the identity code of the certificate
8. the advanced electronic signature of the certification-service-provider issuing it
9. limitations on the scope of use of the certificate, if applicable; and
10. limits on the value of transactions for which the certificate can be used, if applicable

# Annex II – CA Requirements

## Requirements for certification-service-providers issuing qualified certificates

### ■ Certification-service-providers must:

1. demonstrate the reliability necessary for providing certification services
2. ensure the operation of a **prompt and secure directory** and a **secure and immediate revocation service**
3. ensure that the **date and time when a certificate is issued or revoked** can be determined precisely
4. verify, by appropriate means in accordance with national law, the identity and, if applicable, any specific attributes of the person to which a qualified certificate is issued
5. employ personnel who possess the expert knowledge, experience, and qualifications necessary for the services provided, in particular competence at managerial level, expertise in electronic signature technology and familiarity with proper security procedures; they must also apply administrative and management procedures which are adequate and correspond to recognized standards
6. use trustworthy systems and products which are protected against modification and ensure the technical and cryptographic security of the process supported by them
7. **take measures against forgery of certificates**, and, in cases where the certification-service-provider generates signature-creation data, guarantee confidentiality during the process of generating such data
8. maintain sufficient financial resources to operate in conformity with the requirements in the Directive, in particular to bear the risk of liability for damages, for example, by obtaining appropriate insurance
9. record all relevant information concerning a qualified certificate for an appropriate period of time, in particular for the purpose of providing evidence of certification for the purposes of legal proceedings. Such recording may be done electronically
10. not store or copy signature-creation data of the person to whom the certification-service-provider provided key management services
11. before entering into a contractual relationship with a person seeking a certificate to support his electronic signature inform that person by a durable means of communication of the precise terms and conditions regarding the use of the certificate, including any limitations on its use, the existence of a voluntary accreditation scheme and procedures for complaints and dispute settlement. Such information, which may be transmitted electronically, must be in writing and in readily understandable language. Relevant parts of this information must also be made available on request to third-parties relying on the certificate
12. use trustworthy systems to store certificates in a verifiable form so that:
  - only authorized persons can make entries and changes,
  - information can be checked for authenticity,
  - certificates are publicly available for retrieval in only those cases for which the certificate-holder's consent has been obtained, and
  - any technical changes compromising these security requirements are apparent to the operator

# Annex III – SS CD Requirements

## Requirements for secure signature-creation devices:

1. Secure signature-creation devices (SSCD) must, by appropriate technical and procedural means, ensure at the least that the signature-creation data used for signature generation:
  1. can practically **occur only once**, and that their secrecy is reasonably assured
  2. cannot, with reasonable assurance, be derived and the signature is **protected against forgery** using currently available technology
  3. can be **reliably protected** by the legitimate signatory **against the use of others**
2. Secure signature-creation devices must not alter the data to be signed or prevent such data from being presented to the signatory prior to the signature process

# Annex IV – Verification Recommendations

## Recommendations for secure signature verification:

- During the signature-verification process it should be ensured with reasonable certainty that:
  1. the data used for verifying the **signature correspond to the data displayed to the verifier**
  2. the **signature is reliably verified** and the result of that verification is correctly displayed
  3. the verifier can, as necessary, reliably establish the contents of the signed data
  4. the **authenticity and validity of the certificate** required at the time of signature verification are reliably verified
  5. the result of verification and the signatory's identity are correctly displayed
  6. the use of a pseudonym is clearly indicated; and
  7. any security-relevant changes can be detected