The Belgian eID card

Physical and Optical Security Features

Physical elements

- Card made of durable polycarbonat
- Standard bank card format (ISO)
- Very rich set of physical security elements
Physical and Optical Security Features

• Rainbow Printing
  – Security printing, making it almost impossible to copy using traditional techniques

• Guilloche
  – Printing of thin lines to prevent from copying (like with bank notes)

• Changeable Laser Image
  – Picture and part of national number are engraved through a lenticular window. One or the other image is visible, depending on card orientation.

Physical and Optical Security Features

• Optical Variable Ink
  – Printing with changing colours, depending on card orientation

• Alphagram
  – Transparent holographic element, with light reflection and changing image
Physical and Optical Security Features

- **Laser printing**
  - Personalisation under laminate layer for optimal durability and security

- **Micro-letters**
  - Printing of microscopic characters

- **UV objects**

Physical and Optical Security Features

- **Ultra Violet Objects**
Physical and Optical Security Features

- Relief printing techniques
Belgian eID card

Electronic Functions

Evolution

- magnetic stripe card
- processor card
- processor card + crypto processor + Java
- biometrics?

TODAY

Building applications for the Belgian eID
Comparison SIS and eID

- memory card
- naam + natNR
- verzekeringstatus
- -
- -
- -
- beveiliging door apps
- PVC
- gewone bedrukking
- synchrone kaart
- uitgereikt door imv

- smart card
- naam + natNR
- -
- adres
- foto
- digitale handtekening
- zelf-beveiliging
- polycarbonaat
- speciale bedrukking
- asynchrone kaart
- uitgereikt door RRN

Building applications for the Belgian eID

Multi-application JavaCard

- applet 1
- applet 2
- applet 3

3rd party classes
JavaCard RE and API
JavaCard Virtual Machine
card OS and functions

OS and applications on the card
Building applications for the Belgian eID

OS and applications on the card

Multi-application JavaCard

- ID
- 3rd party classes
  - JavaCard FrameWork
  - JavaCard Virtual Machine
  - card OS and functions

2 Data Sets on the card

PKCS#15 data structure

- authentication key + certificate
- digital signature key + certificate

- signed by RRN
- signed by RRN

ID
address
2 Data Sets on the card

**eID specific data**

- ID
- address

- signed by RRN
- signed by RRN

---

**File Hierarchy on the Card**

Note: This diagram shows the files and directories as they exist on the card.
Building applications for the Belgian eID

PKCS#15 logical data structure

Note: This diagram shows the logical links between the PKCS#15 objects.

Application Areas

1. DATA CAPTURE

2. IDENTIFICATION & AUTHENTICATION

3. ELECTRONIC SIGNATURE
Building Applications for the Belgian eID card

Card Readers and Terminals

PC/SC

- Cards, readers and computers made by different manufactures work together.

- Device independent APIs

- Resource management to allow multiple applications to share multiple smartcard devices with potentially multiple card slots.
Building applications for the Belgian eID

PC/SC

Smart Card Aware Apps
- Common Dialog
- SCSP
- CryptoAPI
- SCCP

PC/SC Resource Manager

Smart Card Reader Driver Library

User Applications
- 3rd party DLLs

System Services
- Drivers for IFD

Hardware

PC/SC OS support

- Windows
  - from Windows 98 and higher
  - W98 and NT4 require installation of the SmartCard Base Components
  - also in Windows CE

  http://www.microsoft.com/downloads
  and search for “smartcard base components”

- Linux and Mac OS X use “PC/SC Lite”

  http://pcsclite.alioth.debian.org

Smart Card Aware Apps
PC/SC and PIN-pad readers

- PC/SC has no provisions for PIN-pad card readers
- Public eID middleware (CSP and PKCS#11) allows plug-in extensions for PIN-pad readers
- Specifications are available on the FedICT web site
- It is up to a vendor or distributor to provide these extensions for their hardware

Device Classification

<table>
<thead>
<tr>
<th>Class 1</th>
<th>Class 2</th>
<th>Class 3</th>
<th>Class 4</th>
<th>Class 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Connection</td>
<td>unconnected</td>
<td>connected PC/SC</td>
<td>Connected (PC/SC)</td>
<td>Connected (PC/SC)</td>
</tr>
<tr>
<td>PIN entry</td>
<td>key pad</td>
<td>-</td>
<td>key pad</td>
<td>key pad</td>
</tr>
<tr>
<td>UI</td>
<td>LCD display (LED)</td>
<td>LED (buzzer)</td>
<td>LCD display buzzer</td>
<td>LCD display buzzer</td>
</tr>
<tr>
<td>Embedded Crypto Device</td>
<td>X</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Embedded software</td>
<td>Firmware</td>
<td>firmware</td>
<td>firmware</td>
<td>firmware progr/downl</td>
</tr>
<tr>
<td>Example</td>
<td>Classic Vasco C/R tokens</td>
<td>&quot;ISABEL&quot; reader</td>
<td>SPR532</td>
<td>Xiring XiPass ACS ACR80</td>
</tr>
</tbody>
</table>
Card Readers for PocketPC

SIS+SAM eID …

Mobile/Standalone Card Reader

- Compact 12,5 x 7,5 x 1,5 cm
- Light 123 gram
- Non-Volatile Memory read/store/synchronize
- Connects to any PC
- 2 AAA batteries
- programmable in C
- SIS approved
Building applications for the Belgian eID

PIN-pad readers Class 3

switches to PIN pad directly connected to the reader
PIN-pad readers Class 4

Ruggedized Mobile Terminal

- water/weather proof
- GSM/GPRS/WiFi
- Bluetooth
- barcode & MRZ scanner
- fingerprint sensor
- contact/contactless card reader
- PocketPC/Windows CE
The Belgian eID card

Building Applications
Software Development Kit

FedICT eID software

- Microsoft Windows
  - CryptoAPI CSP for Internet Explorer, Outlook, .NET, …
- OS neutral standards
  - PKCS#11 for Linux, MacOSX, Windows and Sun Solaris
- Java OpenCard Framework
FedICT eID SDK

The main goals of the FedICT eID SDK are:

• To provide an easy way to retrieve the identity information from any version of a Belgian Identity Card

• To automate and hide all validation mechanisms

• To provide an easy to use interface to reduce the integration time in applications

• self-sufficient; as an example, all identity functions will automatically
  – select the right application before reading the identity file
  – ensure they are not interrupted in the middle of a file read
  – interpret the contents of a file based on the card version
Each function returning signed data always checks the signature, together with the integrity of the whole certificate chain.

The function returns
- the status of the signature check (long)
- the global status of the certificate validation (long)
- for each certificate
  - the certificate
  - the certificate’s label
  - the individual checking status
  - the individual validation status
  - the individual policy used: OCSP or CRL
FedICT eID SDK

- BEID_Init() – set OCSP and CRL policy
- BEID_Exit()

- BEID_GetID()
  - BEID_GetAddress()
  - BEID_GetPicture()
  read straight from a card validate the content and return the parsed, interpreted result to the application

- BEID_GetRawData()
  - BEID_SETRawData()
  create or work with a binary copy of the public data

- BEID_BeginTransaction()
- BEID_EndTransaction()

- BEID>SelectApplication()

- BEID_ReadFile()
- BEID_WriteFile()
FedICT eID SDK

- BEID_VerifyPIN()
- BEID_ChangePIN()
- BEID_GetStatusPIN()

- BEID_GetVersionInfo()

- BEID_SendAPDU()

Sample code in Visual Basic

```vbnet
Set RetStatus = EIDlib1.Init("", 0, 0, lHandle)
If (RetStatus.GetGeneral = 0) Then
    Set RetStatus = EIDlib1.GetID(MapColID, CertifCheck)
    strName = MapColID.GetValue("Name")
    Label1.Caption = strName
End If

' Set RetStatus = EIDlib1.GetAddress(MapColAddress, CertifCheck)

'strStreet = MapColAddress.GetValue("Street")
Set RetStatus = EIDlib1.Exit()
```
Microsoft: eID support today

Middleware
- Windows 98, Me, NT 4.0, 2000, XP
- Windows logon
  - Windows’ requirements for certificate based logon are incompatible with standard, generic X.509 certificates
  - workaround possible but this would require a custom developed GINA module (logon plugin)

Office
- Full support in MS Office 2003

Internet Explorer
- Full support SSL in 5.5 and above

Web Sites
- ASP and ASP .NET
- SSO with Federal Portal

Applications
- Can do signing and data capture

Microsoft: eID toolkits

Your client
- Managed C++ class
- .NET class Card
- .NET class Address
- .NET class Identity

Microsoft add-on
- FedICT eidlib
- FedICT CSP

public toolkits
Microsoft: eID toolkits

- .NET wrapper and samples for eID API
- XAdES .NET library and documentation
- .NET cookbook with code for authentication service of Federal Portal
- QUEST documents: legal, technical and practical implementation guidelines for advanced electronic signature with qualified certificates

eID support

Middleware
- Windows 98, Me, NT 4.0, 2000, XP
- Mac OS X, Solaris and Linux

Office
- OpenOffice 2.0
- Adobe Reader 6 and 7

Web Browsers
- Firefox and Mozilla

e-Mail clients
- Thunderbird and Mozilla
eID on the Mac

- smart card support only available on Mac OS X
- no smart card support on MacOS 9
- federal government supplies PKCS#11 for Mac OS X 10.2.8 and higher
- Mac OS X 10.4 is the first OS with built in recognition of the Belgian eID
eID on the Mac

Building applications for the Belgian eID

Apparatenbeheerder

<table>
<thead>
<tr>
<th>Device</th>
<th>Details</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>NSS Internal PKCS #11 Module</td>
<td>Status</td>
<td>Not signed</td>
</tr>
<tr>
<td>Algorithm Cryptodrucker</td>
<td>Description</td>
<td>CCID Smart Card Reader</td>
</tr>
<tr>
<td></td>
<td>Fabric</td>
<td>OpengSC project.fine...</td>
</tr>
<tr>
<td></td>
<td>FW versie</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>FW versie</td>
<td>0.0</td>
</tr>
</tbody>
</table>

Built-in Object Token

eID on the Mac

Voer aflatbaar het hoofdwachtwoord in voor het BELLIC (Basic PIN).

---

Annuleren  OK
Building applications for the Belgian eID on the Mac

Rijksregister

Dossier

Identificatie

Persoon

Andere

Vreemdeling

Historiek Consultatie

Transacties

Gegevens worden eventueel geëxporteerd

Building applications for the Belgian eID on the Mac

eID on the Mac

(zetes)
eID on the Mac