Security Mechanism of Electronic Passports

Petr ŠTURC
Coesys Research and Development
Smartcard

CPU 16/32 bit
3.57MHz (20MHz)
1.8 / 3 / 5 V
ROM 16-300 kB
RAM 1-8 kB
EEPROM 8-128kB
Contactless communication

Not RFID!

\[ f = 13.56 \text{ MHz} \]

Near-field \( \rightarrow \) range < 10cm \( \frac{300}{2\pi f} \)

Power via induction

Signal via modulation

ISO 14443
Contactless communication
Threats vs. security mechanisms

- **State’s protection**
  - Modifying data of a given passport
  - Forging a fake passport
  - Cloning a given passport

- **Citizen’s protection**
  - Skimming a passport
  - Eavesdropping the communication

**Passive Authentication [Signature]**
- RSA, DSA, ECDSA
- SHA-1, 224, 256, 384, 512

**Active Authentication [Challenge Response]**
- ISO 9796–2

**Basic Access Control [Reader Authentication]**
- TDES/CBC
- Retail–MAC/DES
- SHA–1 (key der.)

**Secure Messaging [Encryption]**
- TDES/CBC
- Retail–MAC/DES
ICAO Security Mechanisms
RSA vs. ECC

- ECC wins the signature and Key generation match.
- RSA wins the verification match but ECC stays reasonable
- WARNING: Results are chip dependant

ECC: 113ms and 147 ms
Passive Authentication (PA)

- DG1: P<ČEŠTURČ<PETR<...
- DG2: PHOTO
- DG3: FINGER
- DG4: IRIS
- DG7: SIGN
- DG14: EAC PARAMS
- DG15: AA PUB KEY
- SOD:  

Hashes:
- HASH DG1
- HASH DG2
- HASH DG3
- HASH DG4
- ...
- HASH DG14
- HASH DG15
- HASH DG16

RSA, DSA, ECDSA

Document Signer

CMS / PKCS#15 structure

DS private key

Certification Authority (CSA)
Features:

- Keypair generation, CSR generation (ASN.1 templates, cross-signatures), Certificate storage
- SOD generation (from ASN.1 templates)
- Key selection strategies (explicit selection, round-robin, “optimal”, ...)
- Multiple domains
- Connector for Coesys Prod Manager
- Management GUI
- modularity

Supported crypto:

- SW (RSA, RSA-PSS, ECC)
- Luna 3000 HSM (RSA, RSA-PSS, ECC)
- KMS (RSA, RSA-PSS)
‘Fakeproof’ e-passport is cloned in minutes

Steve Boggan

New microchipped passports designed to be foolproof against identity theft can be cloned and manipulated in minutes and accepted as genuine by the computer software recommended for use at international airports.

Tests for The Times exposed security flaws in the microchips introduced to protect against terrorism and organised crime. The flaws also undermine claims that 3,000 blank passports stolen last week were worthless because they could not be forged.

In the tests, a computer researcher cloned the chips on two British passports and implanted digital images of Osama bin Laden and a suicide bomber. The altered chips were then passed as genuine by passport reader software used by the UN agency that sets standards for e-passports.
CSCA Certificates


The Czech Country Signing Certificate Authority (CSCA)
This website contains the information on the Czech CSCA operated by the Ministry of the Interior (MI).

The distinguish name of the CSCA is
C= CZ, O=Czech Republic, OU=Ministry of Interior, CN=CSCA_CZ

CSCA Public Key Certificate
(Serial Number SN=01)

CZE CSCA 20080724.der
Active Authentication (AA)

1. READ AA PUB KEY
2. SEND RND NOUNCE
3. GET AA PRIV KEY
4. ANSWER SIGNED NOUNCE
5. VERIFY

Accessible memory
- DG7: SIGN
- DG14: EAC PARAMS
- DG15: AA PUB KEY
- SOD: ORIGINAL

Protected memory
- OBJECT: AA PRIVATE KEY

e-passport CPU

Inspection System

gemalto
security to be free
Active Authentication - issues

EF.COM not in SOD

Challenge semantic – Active authentication gives non-repudiation (possibility to track the passport holder and have a proof)

- Passport receives “random” string $r$ from a terminal and respond with signature $S(K_{pr}, r)$ where $K_{pr}$ is passport’s private key. Terminal can hide a meaning into the random $r$ (e.g. $r = date // time // location$)
- Can be solved by Chip Authentication (part of EAC)
Basic Access Control (BAC)

MACHINE READABLE ZONE (MRZ)

Inspection System

PASSPORT NR. | DATE OF BIRTH | DATE OF EXPIRY

SHA-1

$K_{SEED}$ | 00000001 (for $K_{ENC}$) 00000002 (for $K_{MAC}$)

SHA-1

$K_{ENC/MAC} = 3DES$ KEY (16 BYTES) NOT USED (4 BYTES)
Basic Access Control (BAC)

- Accessible memory
  - DG7
  - SIGN
  - DG14
  - EAC PARAMS
  - DG15
  - AA PUB KEY
  - SOD

- Protected memory
  - OBJECT
  - KEY $K_{ENC}$
  - OBJECT
  - KEY $K_{MAC}$

- e-passport CPU

- SECURE CHANNEL
  - 3DES encryption + MAC

- Inspection System

$K_{ENC} + K_{MAC}$
Basic Access Control - Detailed

Inspection System

- Generate random number RND.IFD and keying material K.IFD
- $S = \text{RND.IFD} \| \text{RND.ICC} \| \text{K.IFD}$
- $E_{\text{IFD}} = E[K_{\text{ENC}}](S)$
- $M_{\text{IFD}} = \text{MAC}[K_{\text{MAC}}](E_{\text{IFD}})$

E-Passport

- Get challenge

- Generate random number for challenge RND.ICC

- $\text{RND.ICC}$

- MUTUAL AUTHENTICATE($E_{\text{IFD}} \| M_{\text{IFD}}$)

- $E_{\text{ICC}} \| M_{\text{ICC}}$

- Check $M_{\text{IFD}}$
- Decrypt $E_{\text{IFD}}$
- Check that RND.ICC from S is the correct value
- Generate keying material K.ICC
- $R = \text{RND.ICC} \| \text{RND.IFD} \| \text{K.ICC}$
- $E_{\text{ICC}} = E[K_{\text{ENC}}](R)$
- $M_{\text{ICC}} = \text{MAC}[K_{\text{MAC}}](E_{\text{ICC}})$

- Check $M_{\text{ICC}}$
- Decrypt $E_{\text{ICC}}$
- Check that RND.IFD from R is the correct value
Belgian Biometrics passport proven insecure

A research team in cryptography from the Catholic University of Louvain (Louvain-la-Neuve) disclosed serious weaknesses in the Belgian biometric passport, the only type of passport distributed in Belgium since the end of 2004. The work carried out in Louvain-la-Neuve during the course of May 2007 show that **Belgian passports issued between end 2004 and July 2006 do not include any security mechanism to protect the personal data** embedded in the passport’s microchip. Passports issued after July 2006 do benefit from security means that anyone possessing a little cheap to acquire, can steal the passport victim owners and thus without their knowing at risk. This news is all the more surprising as the Foreign Affairs, declared in the Parliament, passport benefited from the security measures of the International Civil Aviation Organization.
Extended Access Control (EAC)
Chip Authentication (CA)

**Diffie-Hellman key exchange**

(DH or ECDH)

**Ephemeral–Static (EC)-Diffie-Hellman**

**Chip:**
- Chip individual static key pair
- Public Key stored in the DG14 (signed)
- Private Key stored in secure memory

**Terminal:**
- Ephemeral key pair dynamically chosen by the terminal

**ECDH (224Bit) asymmetric key agreement**

**3DES (112Bit) symmetric encryption / integrity protection**

**Accessible memory**
- DG7
- DG14
- EAC PARAMS

**Protected memory**
- OBJECT
- DH PRIVATE KEY

**e-passport CPU**

**Secure channel**

(3DES + MAC)

**Inspection System**

**Gemalto**

security to be free
Chip Authentication - Detailed

Read Chip Authentication public key of the ICC ($PK_{ICC}$) and domain parameters $D_{ICC}$ from the EF.DG14.

Generate ephemeral key pair $PK_{IFD}$, $SK_{IFD}$

$K = KA(SK_{IFD}, PK_{ICC}, D_{ICC})$
Terminal Authentication (TA)

Accessible memory

Terminal Authentication (TA)

accessible memory

dg7

sign

DG14

EAC params

dg15

AA pub key

SOD

original

original

e-passport CPU

CVCA

certificate

Present certificate(s)

Send RND challenge

Challenge signed by private key

RSA or ECDSA

Verify cert = signature + expiration + revocation

Problem!
Terminal Authentication – Detailed

**Inspection System**

- Read CVCA references
- Set the reference to the public key
- Calculate signature \( S_{IFD} = \text{Sign}(SK_{IFD}, ID_{ICC} || R_{ICC} || H(PK_{IFD})) \)

**E-Passport**

- \( \text{READ\_BINARY()} \)
- \( \text{EF.CVCA} \)
- \( \text{MSE\_SET\_DST} \)
- \( \text{PSO\_VERIFY\_CERTIFICATE} \)
- \( \text{MSE\_SET\_AT} \)
- \( \text{GET\_CHALLENGE} \)
- \( R_{ICC} \)
- \( \text{EXTERNAL\_AUTHENTICATE} \)

Set the reference to the public key
Verify certificate
Verify signature
EAC Cross-certification

COUNTRY A

CVCA

DV

COUNTRY B

CVCA

DV

CVCA signs DV Certificates
Assigns: Access Rights
Validity Period

DV Signs Inspection System Certificates
Restricts: Access Rights
Validity Period

Arrows denote Certification
Certificate renewal

Examples of validity periods:
- CVCA certificate: 2 years
- DV Certificate: 3 months
- IS Certificate: 1 month
Extended Access Control v2
a.k.a “3rd generation e-passport”
# PACE v2

**Password Authenticated Connection Establishment**

<table>
<thead>
<tr>
<th>MRTD Chip (PICC)</th>
<th>Terminal (PCD)</th>
</tr>
</thead>
<tbody>
<tr>
<td>static domain parameters $D_{PICC}$</td>
<td></td>
</tr>
<tr>
<td>choose random nonce $s \in \text{Dom}(E)$</td>
<td></td>
</tr>
<tr>
<td>$z = E(K_\pi, s)$</td>
<td>$\frac{D_{PICC}}{z}$</td>
</tr>
<tr>
<td>additional data required for $\text{Map}()$</td>
<td>$s = D(K_\pi, z)$</td>
</tr>
<tr>
<td>$\widetilde{D} = \text{Map}(D_{PICC}, s)$</td>
<td>addional data required for $\text{Map}()$</td>
</tr>
<tr>
<td>choose random ephemeral key pair $(SK_{PICC}, PK_{PICC}, \widetilde{D})$</td>
<td>$\widetilde{D} = \text{Map}(D_{PICC}, s)$</td>
</tr>
<tr>
<td></td>
<td>choose random ephemeral key pair $(SK_{PCD}, PK_{PCD}, \widetilde{D})$</td>
</tr>
<tr>
<td>$K = KA(SK_{PICC}, PK_{PCD}, \widetilde{D})$</td>
<td>$K = KA(SK_{PCD}, PK_{PICC}, \widetilde{D})$</td>
</tr>
<tr>
<td></td>
<td>$T_{PCD} = \text{MAC}(K_{MAC}, (PK_{PICC}, \widetilde{D}))$</td>
</tr>
<tr>
<td>$T_{PICC} = \text{MAC}(K_{MAC}, (PK_{PCD}, \widetilde{D}))$</td>
<td>$T_{PCD} = \text{MAC}(K_{MAC}, (PK_{PICC}, \widetilde{D}))$</td>
</tr>
</tbody>
</table>

*gemalto*

security to be free
That's all Folks!