A Secure and Flexible Server-Based Mobile eID and e-Signature Solution

Christof Rath, christof.rath@iaik.tugraz.at,
Institute for Applied Information Processing and Communications
Graz University of Technology, Austria

24 March 2014
Outline

1. Introduction
2. Motivation
3. Implementation
4. Conclusion
Introduction to eID and e-Signatures

Why care about electronic identification?

- Digital Society
- Confidentiality, Integrity, Authenticity
- Identification
- Non-repudiation
- Weak Passwords, Identity Theft
Certificate-based public-key cryptography

Key Generator

Private Key

Public Key

Certification Authority

Signature Creation

Service Provider
Certificate-based public-key cryptography

Key Generator

Private Key

Public Key

Certification Authority

Signature Creation

Service Provider
From PKI to eID

Bind electronic identifier to the user’s key-pair

- Serial number of the certificate
  - Changes with new certificate

- Existing UID as part of certificate
  - Privacy issue?

- External binding: Identity Link
  - Use certificate w/o revealing identity
eID based on smartcards

Benefits:
- Secure storage of private key
- Implicit 2-factor authentication

Drawbacks:
- Hardware requirements
- Driver/middleware installation
- Platform dependency
Motivation

Mobile eID solutions

SIM-based eID
- Private key physically at the SIM card
- Only MNO has access
- Limited usage abroad
- Processing on mobile device

Server-side eID
- Private key stored encrypted at the server
- User and server secret required
- Only SMS functionality
- Not bound to SIM card or MNO
Motivation

Still, is it relevant?

Austria: Active e-cards and mobile eID
Motivation

What is missing?

- Current approaches are eGovernment driven
- Companies can use government eID
- Still bound to government requirements
- How do deal with non-citizens?
- Authorization based on issuing CA?
- Different second factor?
- Company resources/policies?
Implementation
Logical phases of an eID

- **Registration:**
  - Identity verification

- **Activation:**
  - Create key-pairs
  - Link identity
  - Issue certificate

- **Usage:**
  - Sign data
Logical phases of an eID

- Registration:
  - Identity verification

- Activation:
  - Create key-pairs
  - Link identity
  - Issue certificate

- Usage:
  - Sign data
Logical phases of an eID

- **Registration:**
  - Identity verification

- **Activation:**
  - Create key-pairs
  - Link identity
  - Issue certificate

- **Usage:**
  - Sign data
Spring Framework: Flexibility included

Core features:

- Inversion of control (Dependency injection)
- Interface- and annotation-driven design

Additionally:

- Simplified development (Web MVC, SOAP services)
- Data handling (e.g., via Hibernate)
- Messaging (e.g., via ActiveMQ)
- Utilities
Implementation

Overview of core components

- Certification Authority
- SIR Web Service
- Person Register
- Person Records
- Internet
- Shadow Database
- Outer Activation
- Outer Usage
- HSM
- Inner Activation
- Inner Usage
- Private Database
- OTP Gateway
- SIRWeb Service
- PersonRegister
- CertificationAuthority
Registration via identification records

- Personal information
- Provided ID
- Binding (to mobile phone)
- Hash \( H(Binding || ActivationPIN) \)
- Signed by registration officer
- Received via web service
Person register: Source of the UID

- Interface between person database and activation
  - Existing governmental records
  - Existing employee records
  - Growing during registration/activation

- Generates Identity Link
  - Can be used as is, or
  - Extract relevant information
External services

Certification Authority:
- Public key as PKCS#10 request
- Additional data from person register

OTP Gateway:
- Generate TAN and reference value
- Verify received TAN
- Implementation uses online SMS provider
Conclusion

- Scalability from single web container to a national level
- Prepared for qualified electronic signatures
- Core components adapt to the use-case
Questions?

can also be asked offline:
christof.rath@iaik.tugraz.at
Why secure server-based eID?

Key management:

- Usable only in HSM: $K_{\text{sig}}^{\text{priv}}, K_{\text{hsm}}$
- HSM or software: $K_{\text{store}}^{\text{priv}}, K_{\text{pin}}$

1. $EK_{\text{sig}} = \text{encrypt}_{K_{\text{store}}^{\text{pub}}}(\text{wrap}_{K_{\text{hsm}}^{\text{priv}}}(K_{\text{sig}}^{\text{priv}}))$
2. $K_{\text{pw}} = \text{derive}(\text{password})$
3. $EK_{\text{store}} = \text{encrypt}_{K_{\text{pw}}^{\text{priv}}}(K_{\text{store}}^{\text{priv}})$

Christof Rath, IAIK
24 March 2014
Registration

1. Verify the identity
2. Fill out form
3. Create random PIN
4. Calculate hash value
5. Sign SIR
6. Upload SIR
7. Send PIN to user
Registration

1. Verify the identity
2. Fill out form
3. Create random PIN
4. Calculate hash value
5. Sign SIR
6. Upload SIR
7. Send PIN to user
Registration

1. Verify the identity
2. Fill out form
3. Create random PIN
4. Calculate hash value
5. Sign SIR
6. Upload SIR
7. Send PIN to user

Use activation PIN: 104806
Best regards
Activation

1. Load ID record
2. Setup account
3. Verify possession
4. Create key-pairs
5. Access person register
6. Request certificate
7. Inform user
Canned demo

Activation

1. Load ID record
2. Setup account
3. Verify possession
4. Create key-pairs
5. Access person register
6. Request certificate
7. Inform user

Christof Rath, IAIK
24 March 2014
Canned demo

Activation

1. Load ID record
2. Setup account
3. Verify possession
4. Create key-pairs
5. Access person register
6. Request certificate
7. Inform user
Usage

1. U: Send request
2. S: Show login form
3. S: Send TAN
4. U: Verify data
5. U: Send TAN
6. S: Verify TAN
7. S: Sign data
8. S: Return result
Canned demo

Usage

1. U: Send request
2. S: Show login form
3. S: Send TAN
4. U: Verify data
5. U: Send TAN
6. S: Verify TAN
7. S: Sign data
8. S: Return result
Canned demo

Usage

1. U: Send request
2. S: Show login form
3. S: Send TAN
4. U: Verify data
5. U: Send TAN
6. S: Verify TAN
7. S: Sign data
8. S: Return result

Christof Rath, IAIK
24 March 2014