Privacy-ABC technology on Mobile Phones

by Gert Læssøe Mikkelsen,
Alexandra Institute
Challenges and possibilities.

- Pilots and Reference implementation
  - Focus on Client(PC)-Server and smartcards
- Users are using mobile devices
- Smart Phone feasibility study

- Users bring their smart phones everywhere
- New Use cases – e.g., in the physical world.
  - Now even iPhones come with NFC – currently very restricted!
Challenges on mobile devices

• Platform?:
  ▪ Native – very diverse
    • Android, iOS, Windows Phone etc.
  ▪ Common language: JavaScript?
  ▪ Cloud IdMaaS?

• Computational power?
• Storage of keys and credentials.
• Usability
p-ABC on mobile devices

- Relevant roles
  - User
  - Part of User’s SW (Smart Card emulation)
  - Verifier
  - Inspection

- Not so relevant roles
  - Issuer
  - Revocation authority
Smart Card emulation

- Proof of concept
- Still Client(PC)-server setup
  - + Development time
  - + Performance
  - + Convenience for the user
  - + User interface
- - Security
- - Devices
Native App

- Implemented the user service from the ABC4Trust reference implementation as a mobile service-app
- Android!
  - ABC4Trust Reference implementation in Java
- Security
  - Keys/Credential stored in ABC4Trust App’s internal memory
- Usability?
Native App

Smartphone

Application

1: Request

2: Policy/Verifier URL

p-ABC App

3: Policy/Verifier URL

4: Presentation Token

5: Session Key and revealed Attributes

6: Session Key

7: Session Key

Service Provider

Verifier
Native App

Danish Literature

The-very-cool-course

Privacy and Security 101

The following request was sent from Course evaluation of The-very-cool-course:

Show that you own a Course Credential and reveal the value of course id (which is The-very-cool-course).

Show that you own a University Credential

Present a scope-exclusive pseudonym for the scope urn:patras:evaluation.

OK

Cancel
MS U-Prove Native App.

- MS U-Prove C# version can run on Windows Phones
JavaScript?

• JavaScript is highly cross platform
  ▪ Everything with a modern browser
  ▪ Not build for security/Cryptography
  ▪ Where to store keys/credentials securely?
    • Server side?
    • Cookies?
    • Browser Key store?
    • Issue when needed?
JavaScript Prototype

• Prototype implementation of MS U-Prove
  ▪ U-Prove is simpler than the ABC4Trust reference implementation and Identity Mixer.

• Elliptic Curves using “jsbn” (“Stanford”) library.

• Interacts with MS U-Prove C# version
JavaScript

Smartphone

Browser

ABC4Trust.eu only!

1: Request

3: Presentation token

2: Crypto.js

ABC4Trust.eu

Service Provider
JavaScript Performance

- Very dependent on platform, and use of libraries!
- Our implementation:
  - 2.1 sec (Galaxy Nexus, default browser)
  - 30 sec (iPhone 5, Safari)
- Others are getting very different timings, with iPhones nearly as fast as Androids.
JavaScript the new language for Crypto?

• A lot is happening!
  - Since this task of the project was finished:

  • Microsoft U-Prove JavaScript (July 2014)
  • Microsoft Research JavaScript Cryptography Library (August 2014)
  • Google End-to-end Chrome Extension (June 2014)
Conclusion

• Using p-ABC’s on mobile devices is feasible
  ▪ both as native applications and JavaScript.
• New use cases/improved user experience.
• New security issues
  ▪ mobile devices are vulnerable to a number of attacks which should be addressed.
• A lot is happening on JavaScript right now.
• D4.4 Smartphone feasibility analysis (www.abc4trust.eu)