ABC4Trust
Reference Implementation
Michael Østergaard, Miracle A/S

Dissemination Event
2014-09-29
Athens
Agenda

• Introduction
• Overview
  ▪ Components
  ▪ Interfaces
  ▪ Data Flow
• Demo application
At a Glance

- Implements the ABC4Trust architecture
- Written in Java
  - Most source code is available on GitHub
  - Uses maven as build tool
- Features
  - Designed for a typical desktop/server setup
  - Supports hardware and software smart cards
  - User interaction via a User Service and Browser Plugin
Architecture Overview

User Deployment:

Verifier / Issuer / Inspector / Revocation Authority Deployment:

User Interface

Application

Internet

API

ABC Engine

Storage

Cryptographic Engine

Privacy-ABC Framework

Application

API

ABC Engine

Storage

Cryptographic Engine

Privacy-ABC Framework

2014-09-29
Michael Østergaard, Miracle A/S
Central Components

• core-abce
  • abc4trust-xml
    • Schema definitions for presentation policies, credential specifications, etc.
  • abce-interfaces
    • Interfaces for abce-components
  • abce-components
    • Implementation of core components
  • abce-services
    • REST API for abce-components

• java-ui
  • User interface
Interfaces

• Java API for the ABC Engine
• RESTful web services
  ▪ Mainly for demonstration and integration testing
  ▪ Set up the ABC Engine
    • Import parameters from files
      ▪ E.g. credential specifications and issuance policies
    • Perform simple cross validations
      ▪ E.g. does there exist issuer parameters for the UID in the issuance policy?
    • Export parameters if needed
      ▪ E.g. issuer parameters for other components
Data Flow

• An issuance/presentation between a User and Issuer/Verifier involves communication between the following components:
  ▪ Issuer / Verifier Application
  ▪ Browser Plugin
  ▪ Credential Selector
• And indirectly also
  ▪ ABCE on both Issuer/Verifier and User side
Data Flow

1. The application notifies the Browser Plugin via JavaScript about the action to take, policies and endpoints.

2. The Browser Plugin retrieves the resources and passes them to the User Service.

3. The User Service computes the possible ways to satisfy the policy and returns the list to the Browser Plugin.

4. The Browser Plugin passes this information to the Credential Selector which presents the choices to the user.
5. Once the user has made a selection, the choice is returned to the Browser Plugin.

6. The Browser Plugin passes the choice to the User Service.

7. The User Service computes an ABC protocol message (a presentation token or issuance message) which is sent to the Browser Plugin.

8. The Browser Plugin sends this protocol message to the application, which can then act accordingly.
Building an Application

• Setup
  1. Define credential specification and policies
  2. Setup system parameters
  3. Setup inspector parameters
  4. Setup revocation authority parameters
  5. Setup issuer parameters

• Runtime
  ▪ Call the ABC Engine interfaces (REST or Java)
  ▪ Keep track of sessions
Demo Application Flow

- **Hotel booking**
  - User gets credentials
  - User books a hotel room
  - User does not show up
  - Hotel asks inspector for gets credit card number
  - Inspector reveals credit card number
  - Hotel collects payment

- **Age verification**
  - User proves that he is over a certain age
Demo

HOTEL BOOKING
Hotel Booking

ISSUER
Issuer
Demo Application

Issuer Application

Create Person

Person Id (autogenerated)
Firstname * John
Lastname * Doe
Gender * M
Birthday * 21 July 1976
PIN (autogenerated)
## Issuer Application

### Person List

<table>
<thead>
<tr>
<th>Person Id</th>
<th>Firstname</th>
<th>Lastname</th>
<th>Gender</th>
<th>Birthday</th>
<th>PIN</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.000.000.000</td>
<td>John</td>
<td>Doe</td>
<td>M</td>
<td>1976-07-21</td>
<td>14776</td>
</tr>
<tr>
<td>1.000.000.001</td>
<td>Alice</td>
<td>Von Nextdoor</td>
<td>F</td>
<td>1981-02-03</td>
<td>53281</td>
</tr>
</tbody>
</table>
## Issuer Demo Application

![Issuer Demo Application Screenshot](image-url)

### Person Information

- **Person ID:** 1.000.000.000
- **First Name:** John
- **Last Name:** Doe
- **Gender:** M
- **Birthday:** 1976-07-21
- **PIN:** 14776

### Credit Card Credentials

<table>
<thead>
<tr>
<th>Number</th>
<th>Issuer</th>
<th>Holder</th>
<th>Expire</th>
<th>Issued</th>
<th>Revoked</th>
</tr>
</thead>
<tbody>
<tr>
<td>123.400.000.001</td>
<td>AMEX</td>
<td>John Doe</td>
<td>2016-08-05</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>123.400.000.002</td>
<td>VISA_BEST_BANK</td>
<td>John Doe</td>
<td>2016-08-05</td>
<td>☐</td>
<td>☐</td>
</tr>
</tbody>
</table>

### ID Card Credentials

<table>
<thead>
<tr>
<th>Person Identifier</th>
<th>First Name</th>
<th>Last Name</th>
<th>Birthday</th>
<th>Issued</th>
<th>Revoked</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.000.000.000</td>
<td>John</td>
<td>Doe</td>
<td>1976-07-21</td>
<td>☐</td>
<td>☐</td>
</tr>
</tbody>
</table>

### Passport Credential

<table>
<thead>
<tr>
<th>Number</th>
<th>Country</th>
<th>First Name</th>
<th>Last Name</th>
<th>Birthday</th>
<th>Expire</th>
<th>Gender</th>
<th>Issued</th>
<th>Revoked</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.000.000.000</td>
<td>CH</td>
<td>John</td>
<td>Doe</td>
<td>1976-07-21</td>
<td>2024-08-05</td>
<td>M</td>
<td>☐</td>
<td>☐</td>
</tr>
</tbody>
</table>

### Student Card Credentials

<table>
<thead>
<tr>
<th>Matriculation Number</th>
<th>School</th>
<th>Name</th>
<th>Birthday</th>
<th>Issued</th>
<th>Revoked</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.000.000.000</td>
<td>ETH</td>
<td>John Doe</td>
<td>1976-07-21</td>
<td>☐</td>
<td>☐</td>
</tr>
</tbody>
</table>
Issuer Application

Show CreditCardCredential

Number: 123,400,000,001
Holder: John Doe
Expire: 2016-08-05 00:00:00 CEST
Issuer: AMEX
Issued: False
Revoked: False
Revocation Handle: 

Back
Hotel Booking

USER @ ISSUER
Authenticate with PIN to start Issuance of Credentials

Person Identifier: 1000000000
PIN: 14776

Authenticate
## User @ Issuer Demo Application

![End User Application @ Issuer](image)

### Select Credentials to Issue and press 'Start'

<table>
<thead>
<tr>
<th>Select</th>
<th>Credential Type</th>
<th>UUID</th>
</tr>
</thead>
<tbody>
<tr>
<td>✔</td>
<td>CREDITCARD_Ameth</td>
<td>ca4bf46-b11f-401d-a309-4d51b26ef2f72</td>
</tr>
<tr>
<td>✔</td>
<td>CREDITCARD_VISA BEST BANK</td>
<td>3c5c6473-df1f-43b7-b3d4-51b4c8c9f9f</td>
</tr>
<tr>
<td>✔</td>
<td>ID CARD</td>
<td>4f0d625f-0437-4cc3-8512-77c2108a539</td>
</tr>
<tr>
<td>✔</td>
<td>PASSPORT CH</td>
<td>f9f11fbc-0804-48da-b596-81c5c3e4b0</td>
</tr>
<tr>
<td>✔</td>
<td>STUDENT CARD</td>
<td>94109e05-1501-4bde-8099-2810f302f18</td>
</tr>
</tbody>
</table>

[Link to authentication page](r-application/credentialissuance/authenticate)
Properties of credential to be issued

- The credential is issued by http://www.amex.com/abc/Isskey/Idemix and is of type AMEX Credit Card

Obtain credential

Mode: Issuance. Accepted languages: [en, da, en_US]. SessionID Issuance140725062357721855.
User @ Issuer
Demo Application

ABC4Trust Client Application

- Credential Repository
- AMEX Credit Card
- Visa Credit Card
- ID Card
- Passport
- Student ID

Credential Attributes

Issuer: Swiss Passport (http://admin.ch/passport/issuancekey_v1.0/idemix)
Revoked by Issuer: The credential is not revocable.
Name: John
Lastname: Doe
Birthday: 21-07-1976
Passport Number: 100000000
Expiration Date: 05-08-2024
Issued By: Proper Authority in CH
Hotel Booking

USER @ VERIFIER
User @ Verifier
Demo Application

End User Application @ Verifier

Create Booking

Room Category: NORMAL

Arrival Date: 9 August 2014

Free Mini Bar: ✔

Nights: 3

Single Room: ☐

Start Booking
User @ Verifier
Demo Application
User @ Verifier
Demo Application

End User Application @ Verifier

Result of Hotel Booking

Your Booking has been accepted!

Create new Booking
Hotel Booking

VERIFIER
## Verifier Application

### Booking List

<table>
<thead>
<tr>
<th>Room Category</th>
<th>Arrival Date</th>
<th>Nights</th>
<th>Single Room</th>
<th>Confirmed</th>
<th>Sent To Inspector</th>
<th>Payment Cleared</th>
</tr>
</thead>
<tbody>
<tr>
<td>NORMAL</td>
<td>2014-08-09 00:00:00 CEST</td>
<td>3</td>
<td>False</td>
<td>True</td>
<td>False</td>
<td>False</td>
</tr>
<tr>
<td>LAKEVIEW</td>
<td>2014-08-16 00:00:00 CEST</td>
<td>4</td>
<td>False</td>
<td>True</td>
<td>False</td>
<td>False</td>
</tr>
<tr>
<td>DISCOUNT</td>
<td>2014-09-05 00:00:00 CEST</td>
<td>1</td>
<td>True</td>
<td>True</td>
<td>False</td>
<td>False</td>
</tr>
</tbody>
</table>
Verifier
Demo Application

Verifier Application

Handle Booking
Booking was confirmed by Customer. In case Customer does not show up we can claim our payment from inspector.

Arrival Date: 2014-08-09 00:00:00 CEST
Room Category: NORMAL
Nights: 3
Single Room: False
Free Mini Bar: True
Confirmed: True
Sent To inspector: False
Payment Cleared By inspector: False

Customer did not show up

Send Presentation Token to Inspector
Verifier Application

Handle Booking

PresentationToken was sent to Inspector 200

Payment claim has been sent to Inspector - but is not confirmed yet!

- Arrive Date: 2014-08-09 00:00:00 CEST
- Room Category: NORMAL
- Nights: 3
- Single Room: False
- Free Mini Bar: True
- Confirmed: True
- Sent To Inspector: True
- Payment Cleared By Inspector: False

Get Payment Clearing Status From Inspector
Hotel Booking

INSPECTOR
### Inspection Request List

<table>
<thead>
<tr>
<th>Customer Name</th>
<th>Inspection Token</th>
<th>Inspection Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>SweetDreamSuites</td>
<td>99310c8e-3e40-4b81-b3a7-4c50ef496212</td>
<td></td>
</tr>
<tr>
<td>SweetDreamSuites</td>
<td>c62d24e7-d778-4f46-a1dc-c657f8f7acd6</td>
<td></td>
</tr>
</tbody>
</table>
Inspector
Demo Application

Show InspectionRequest

Customer Name: SweetDreamSuites
Inspection Token: 99310c8e-3e40-4b81-b3a7-4c50ef496212
Inspection Result

Inspect Presentation Token
Inspector
Demo Application

Show InspectionRequest

Presentation token was inspected: CardNumber = 123400000002

Customer Name: SweetDreamSuites
Inspection Token: 99310c8e-3e40-4b81-b3a7-4c50ef496212
Inspection Result: CardNumber = 123400000002

Back
Inspector
Demo Application

InspectionRequest List

Presentation token was inspected: CardNumber = 123400000002

<table>
<thead>
<tr>
<th>Customer Name</th>
<th>Inspection Token</th>
<th>Inspection Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>SweetDreamSuites</td>
<td>99310c8e-3e40-4b81-b3a7-4c50ef496212</td>
<td>CardNumber = 123400000002</td>
</tr>
<tr>
<td>SweetDreamSuites</td>
<td>c62d24e7-d778-4f46-a1dc-c657f8f7acd6</td>
<td></td>
</tr>
</tbody>
</table>
Hotel Booking

VERIFIER
Verifier
Demo Application

Verifier Application

Handle Booking

PresentationToken was accepted by Inspector200

Payment has been cleared by Inspector1

Arrival Date: 2014-08-09 00:00:00 CEST
Room Category: NORMAL
Nights: 3
Single Room: False
Free Mini Bar: True
Confirmed: True
Sent To Inspector: True
Payment Cleared By Inspector: True

Back
## Booking List

<table>
<thead>
<tr>
<th>Room Category</th>
<th>Arrival Date</th>
<th>Nights</th>
<th>Single Room</th>
<th>Confirmed</th>
<th>Sent To Inspector</th>
<th>Payment Cleared</th>
</tr>
</thead>
<tbody>
<tr>
<td>NORMAL</td>
<td>2014-08-09 00:00:00 CEST</td>
<td>3</td>
<td>False</td>
<td>True</td>
<td>True</td>
<td>True</td>
</tr>
<tr>
<td>LAKEVIEW</td>
<td>2014-08-16 00:00:00 CEST</td>
<td>4</td>
<td>False</td>
<td>True</td>
<td>True</td>
<td>False</td>
</tr>
<tr>
<td>DISCOUNT</td>
<td>2014-09-05 00:00:00 CEST</td>
<td>1</td>
<td>True</td>
<td>True</td>
<td>False</td>
<td>False</td>
</tr>
</tbody>
</table>
Demo

AGE VERIFICATION
Age Verification

USER @ VERIFIER
End User Application @ Verifier

Verify Birthday

Enter date. Date will be compared to Birthday attributes within your credentials.

Birthday: 5 August 1986

Verify
End User Application @ Verifier

Show Birthday

Firefox with ABC4Trust Extension: true

The specified date will be compared to Birthday attributes within your credentials.

Birthday 1986-08-05 00:00:00 CEST

Cancel  Start
User @ Verifier
Demo Application
Result of Birthday Presentation

We have verified your PresentationToken with Policy urn:eu:abc4trust:demo:comparebirthday:before:session.

Your birthday is Before 1986-08-05 00:00:00 CEST

Run new Birthday Verification
More Information

• My contact information
  ▪ Michael Østergaard
  ▪ mop@miracle.dk

• ABC4Trust website
  ▪ www.abc4trust.eu