Privacy-ABCs
Features and Architecture

Ahmad Sabouri

ahmad.sabouri@m-chair.de
Deutsche Telekom Chair of Mobile Business & Multilateral Security
Goethe University Frankfurt, Germany
www.m-chair.de

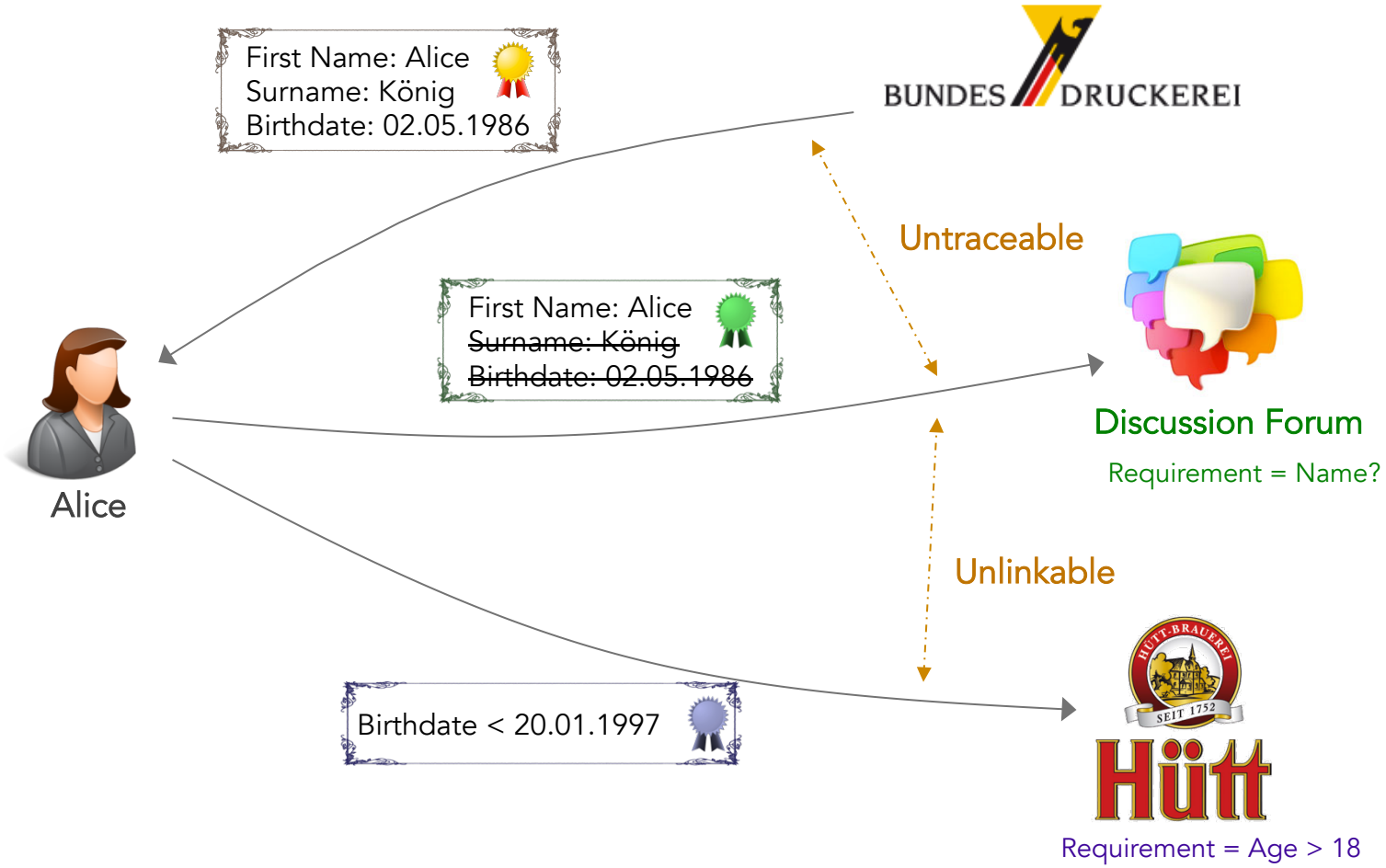
ABC4Trust Summit Event
January 20th, 2015
Brussels, Belgium

A research project funded by the European Commission's 7th Framework Programme.

Goal of the Presentation

- We aim to:
  - give an impression of the features and concepts of the Privacy-ABCs to all the audiences.
  - introduce the architecture, processes, and the artifacts to application and infrastructure developers.

Example Scenario



Credentials and Issuance

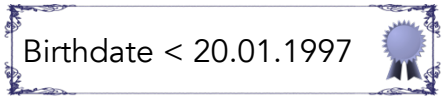
- Credential issuance
  - list of pairs (attribute, value)
  - certified by issuer
  - key-bound to prevent sharing credentials

- Advanced issuance:
  - blindly issued attributes
  - carried-over attributes

Credential Presentation (1)

- Presentation
  - selected attributes from selected credentials

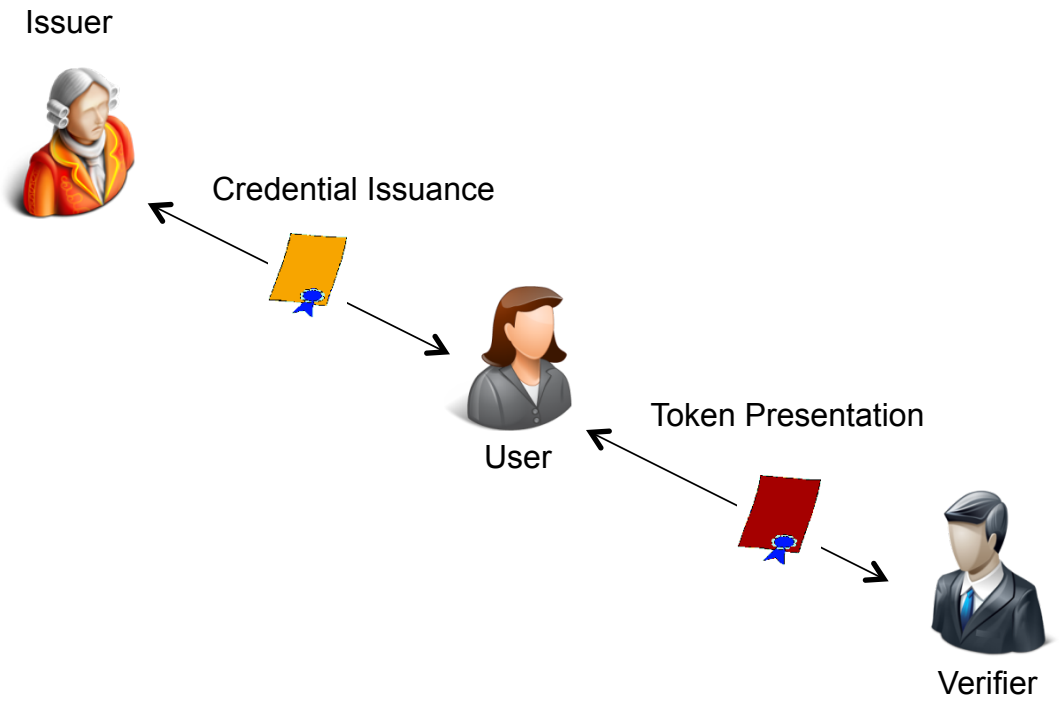
  - predicates over attributes
    - attribute1 =,>,< attribute2 or constant



Credential Presentation (2)

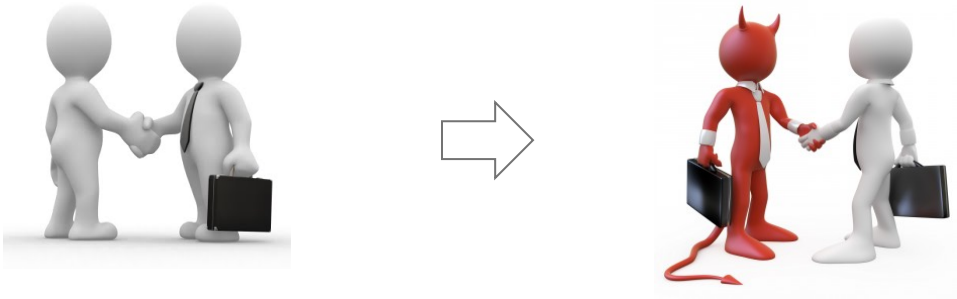
- Pseudonyms
  - equivalent to unlinkable public keys for user's secret key
  - controlled linkability (e.g., account creation)
  - scope-exclusive pseudonym: unique per scope, unlinkable across different scopes

Interactions and Entities



Credential Presentation (3)

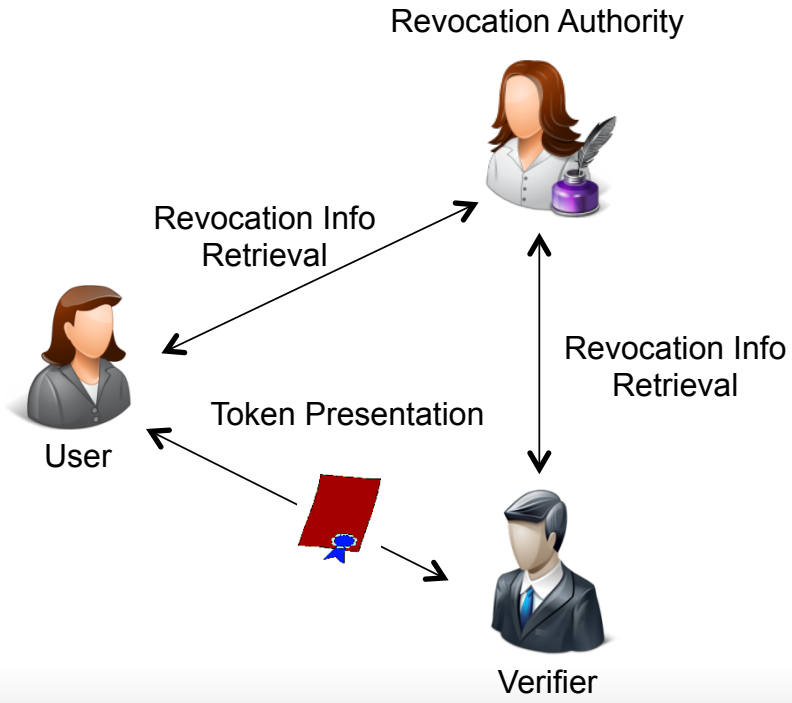
- What happens if the users start misusing the provided anonymity?



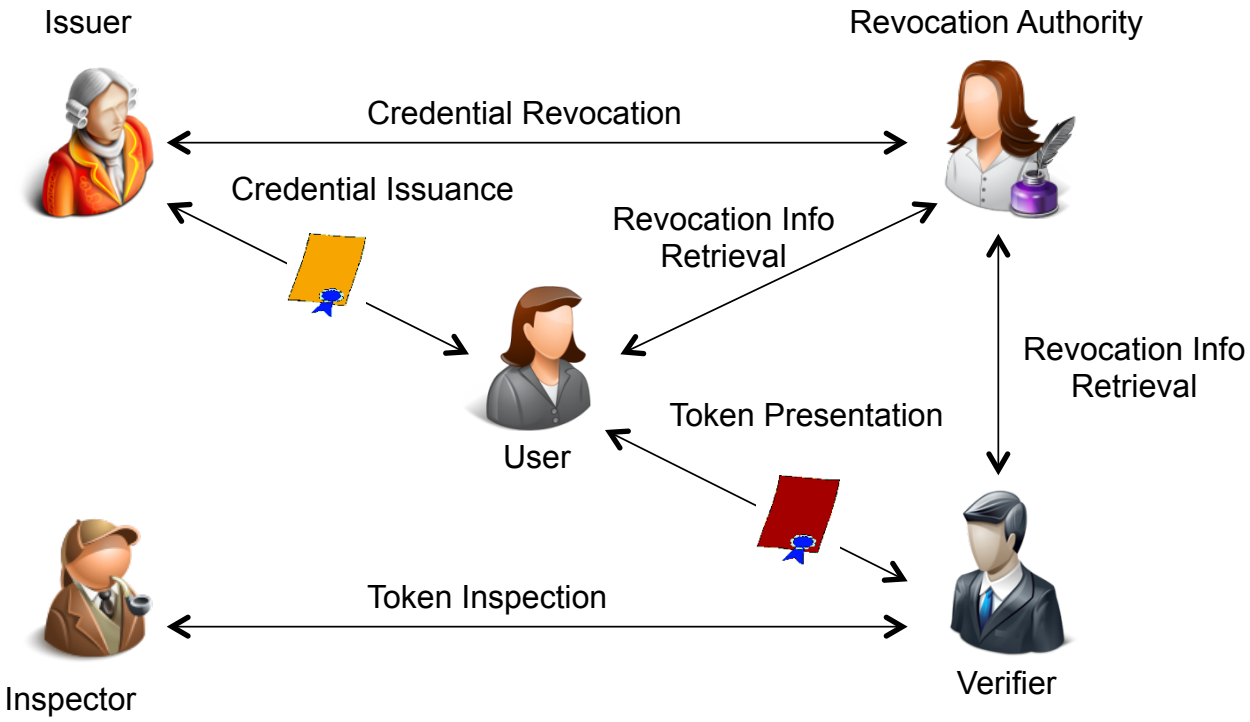
- Inspection
  - The Service Provider makes an agreement with the user at the beginning.
  - The user deliver an identifier encrypted under the public key of the trusted Inspector.
  - The Inspector can investigate the case and reveal the identity of the user if the agreement is violated.

Credential Presentation (4)

- What happens if one needs to invalidate a credential?
  - Credentials are stolen
  - An attribute has changed



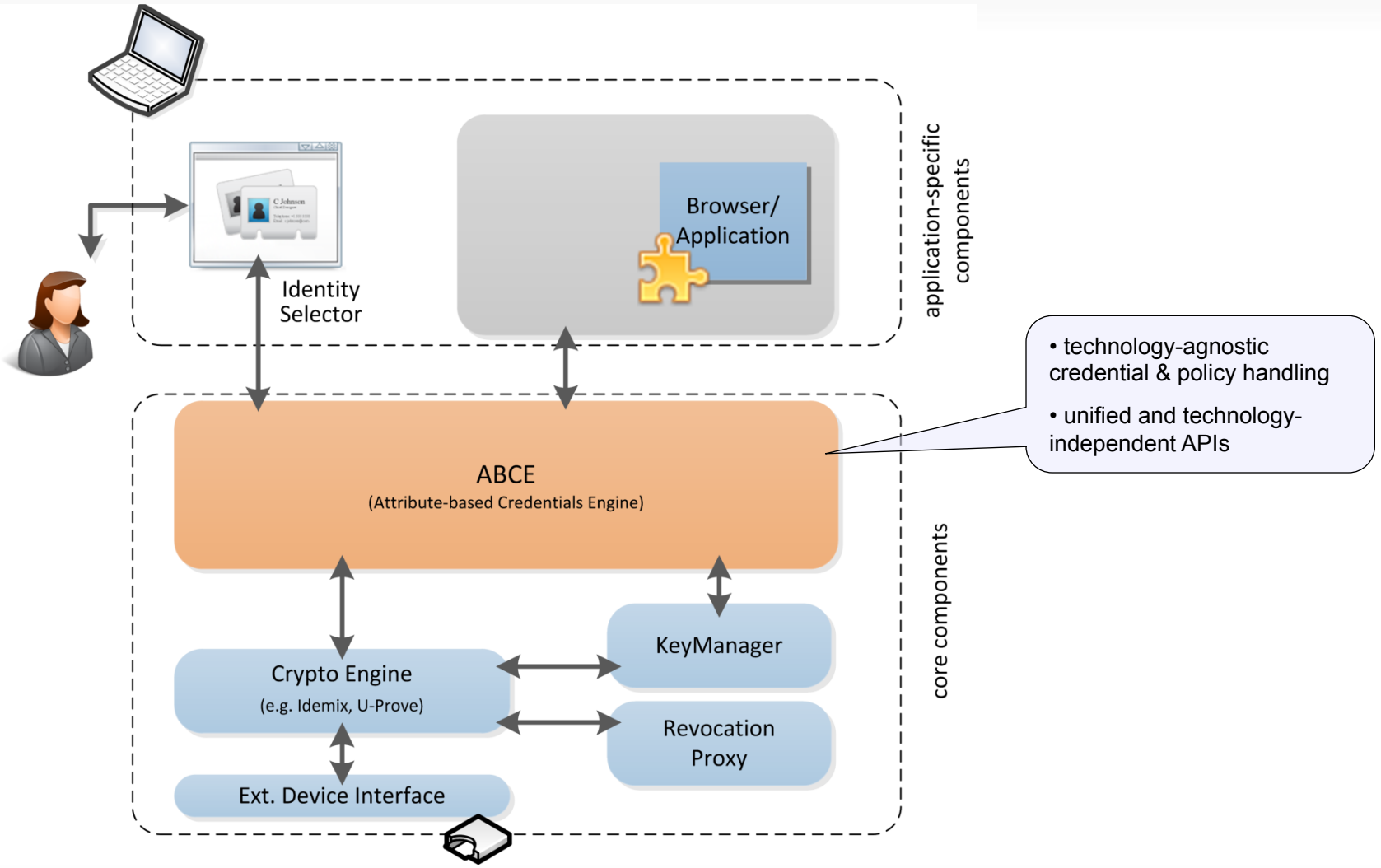
Interactions and Entities



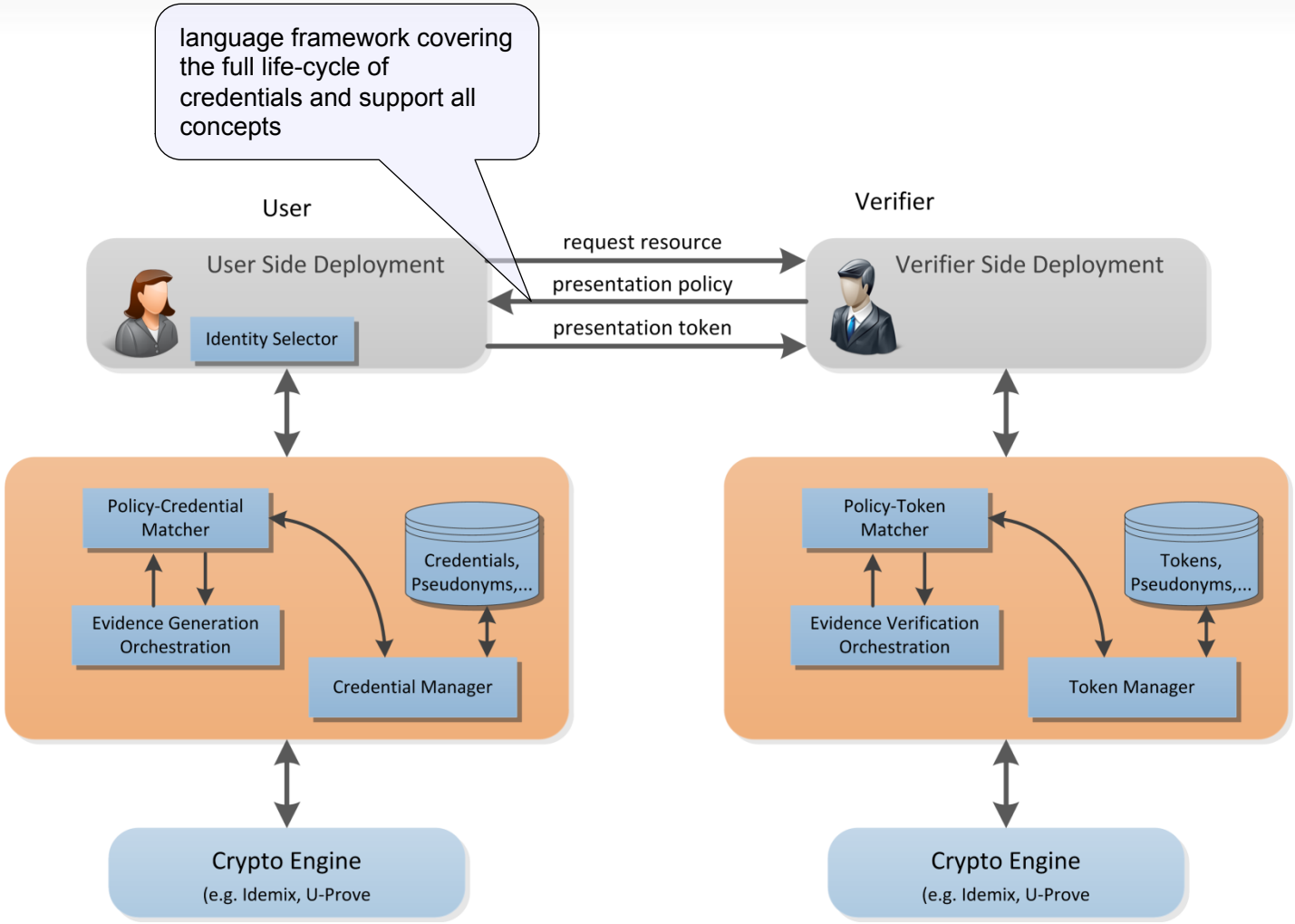
The ABC4Trust Architecture Objectives

- Abstraction of concepts of Privacy-ABCs & unification of features
- A common unified architecture
  - That is independent of the specific technologies
  - Federation of privacy-ABC Systems based on different technologies
  - Interoperability between different privacy-ABC technologies
- Users will be able to
  - obtain credentials for many privacy-ABC technologies and
  - use them on the same hardware and software platforms
  - without having to consider which privacy-ABC technology has been used
- How do we achieve this?
  - System Architecture and components for handling privacy-ABCs
  - Unified and technology agnostic APIs
  - XML specification of all data formats, covering the full life-cycle of credentials

High-level view (user)



High-level view (presentation)



Presentation Policy

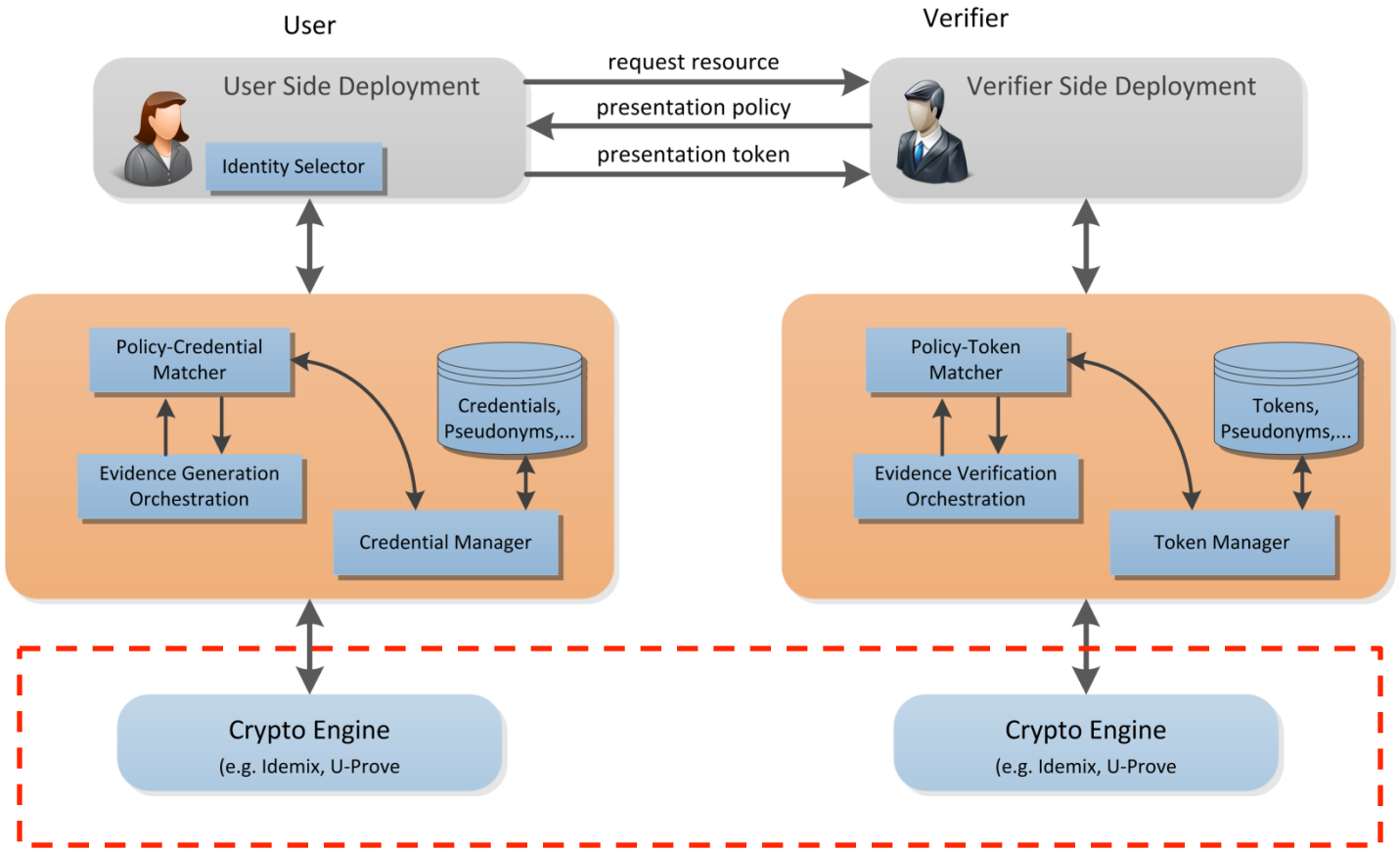
```xml
<?xml version="1.0" encoding="UTF-8"?>

<PresentationPolicyAlternatives xmlns="http://abc4trust.eu/wp2/abcschemav1.0"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xs="http://www.w3.org/2001/XMLSchema"
    xmlns:xenc="http://www.w3.org/2001/04/xmlenc"
    xsi:schemaLocation="http://abc4trust.eu/wp2/abcschemav1.0 schema.xsd"
    Version="1.0">
<PresentationPolicy PolicyUID="policy1" EnforceSameUserBinding="true" EnforceSameDeviceBinding="false">

    <Message>
        <Nonce>aDk3UEMzOTNjOTl1cmZHQ210U0c=</Nonce>
    </Message>
    <Pseudonym Alias="nym" Scope="http://sweden.gov/poll0105" Exclusive="true"/>
    <Credential Alias="id">
        <CredentialSpecAlternatives>
            <CredentialSpecUID>urn:sweden:id</CredentialSpecUID>
        </CredentialSpecAlternatives>
        <IssuerAlternatives>
            <IssuerParametersUID>urn:sweden:id:issuer</IssuerParametersUID>
        </IssuerAlternatives>
        <DisclosedAttribute AttributeType="urn:sweden:id:city"/>
    </Credential>
    <AttributePredicate Function="urn:oasis:names:tc:xacml:1.0:function:date-less-than">
        <Attribute CredentialAlias="id" AttributeType="urn:sweden:id:bdate"/>
        <ConstantValue>1994-01-20</ConstantValue>
    </AttributePredicate>

</PresentationPolicy>
</PresentationPolicyAlternatives>
```

ABC4Trust Crypto Architecture (1)



ABC4Trust Crypto Architecture (2)

- Provide a truly plug-and-play architecture that allows the seamless integration of cryptographic primitives e.g.:
  - Privacy-ABC signatures: Idemix and Uprove
  - Predicate Proofs
- Move away from the "bridging" approach between several incompatible crypto engines
- Encapsulated in components with common interfaces, allowing the rest of the cryptographic layer to be implementation-agnostic

ABC4Trust Crypto Architecture (3)

Summary

- ABC4Trust produced a generic and layered architecture for Privacy-ABCs:
  - Defining features, processes, and artifacts
  - Enabling the Reference Implementation and the Pilots
  - Preventing lock-in situations
- The architecture is more privacy-friendly than the available alternatives, e.g. STORK, which is important for the eIDAS discussion.
- The ABC4Trust Crypto Architecture enables modular instantiation of new Privacy-ABC technologies.

Questions?

Thanks for Your Attention

coord-abc4trust@m-chair.de