W3C Workshop – Ispra, Italy

# A General Certification Framework with Applications to Privacy-Enhancing Certificate Infrastructures

Jan Camenisch
Thomas Gross
Dieter Sommer

October 2006

# Outline

- **Scenario**
- **Protocols**
- **Attribute Assertion Language**

# Outline

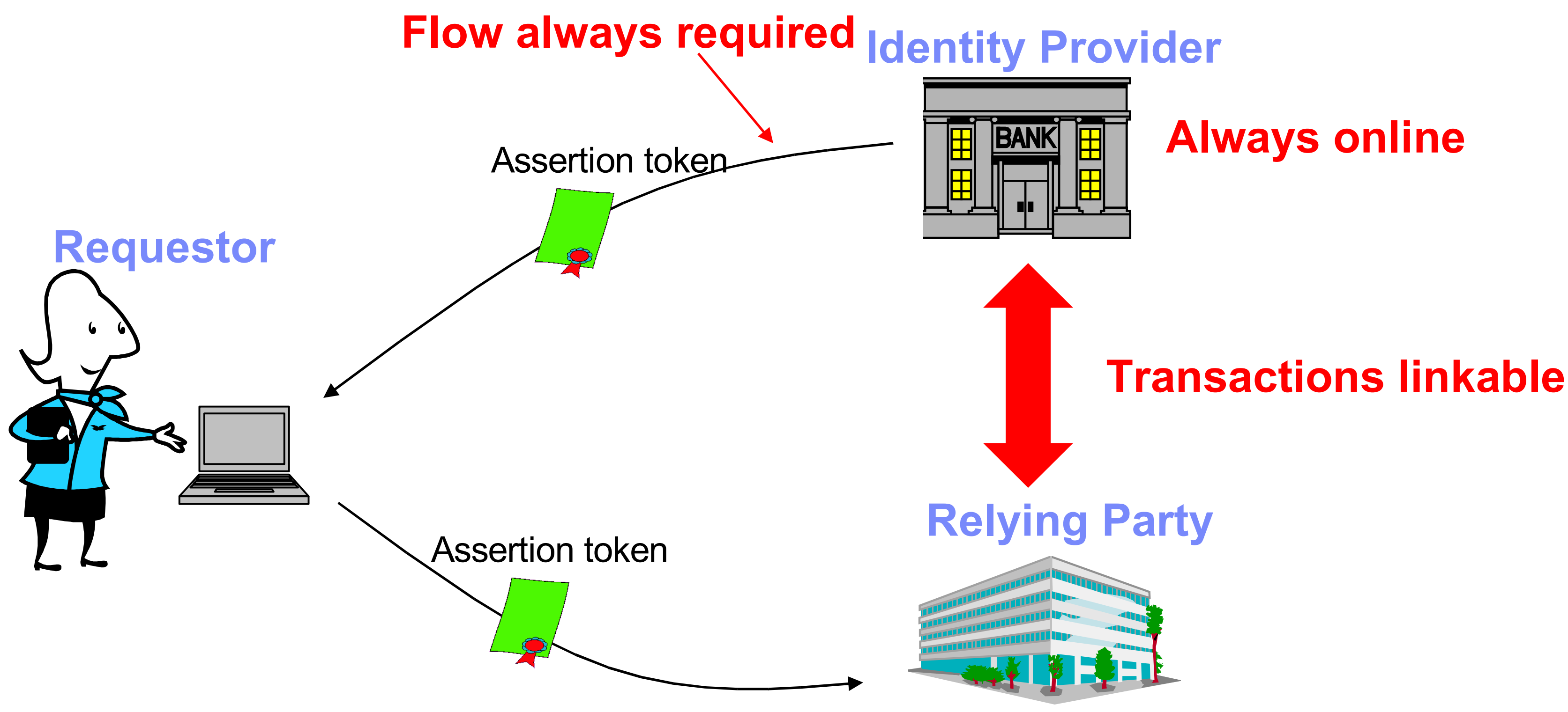
- **Scenario**
- **Protocols**
- **Attribute Assertion Language**

# Attribute Exchange Methods

- **Need for attribute exchange**
  - Attributes are key to many (business) scenarios
  - Attribute information allows to distinguish between  entities
- **Web forms**
  - Non-certified attributes (declared)
  - Tedious to use
  - Error-prone (low data quality)
- **FIM (federated identity management)**
  - Certified attributes (endorsed by Identity Provider IP)
  - Weak attacker model
  - Too much trust in IP
  - Privacy problems

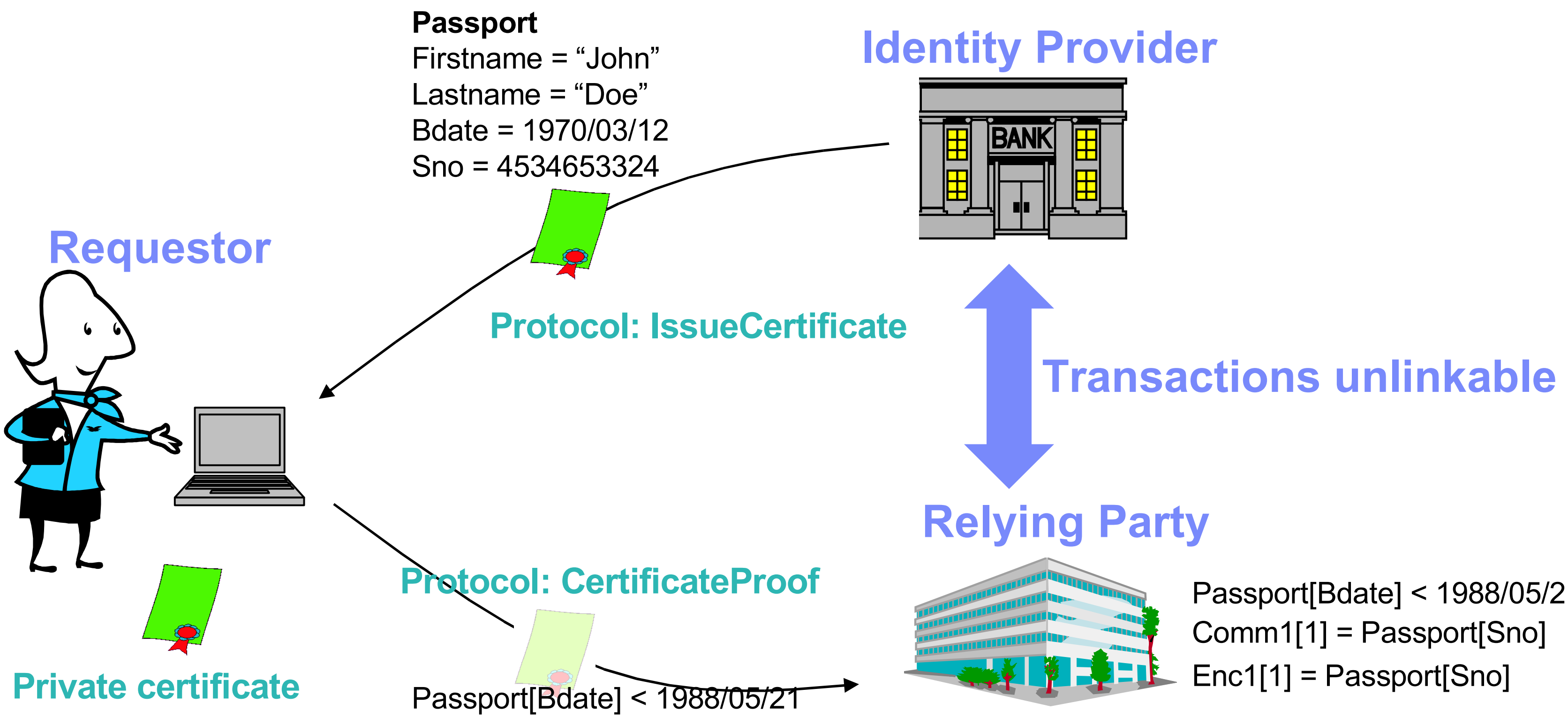
# Attribute Exchange in Traditional FIM Environment



# Outline

- Scenario
- **Protocols**
- Attribute Assertion Language

# Private Certificate Framework – Protocols

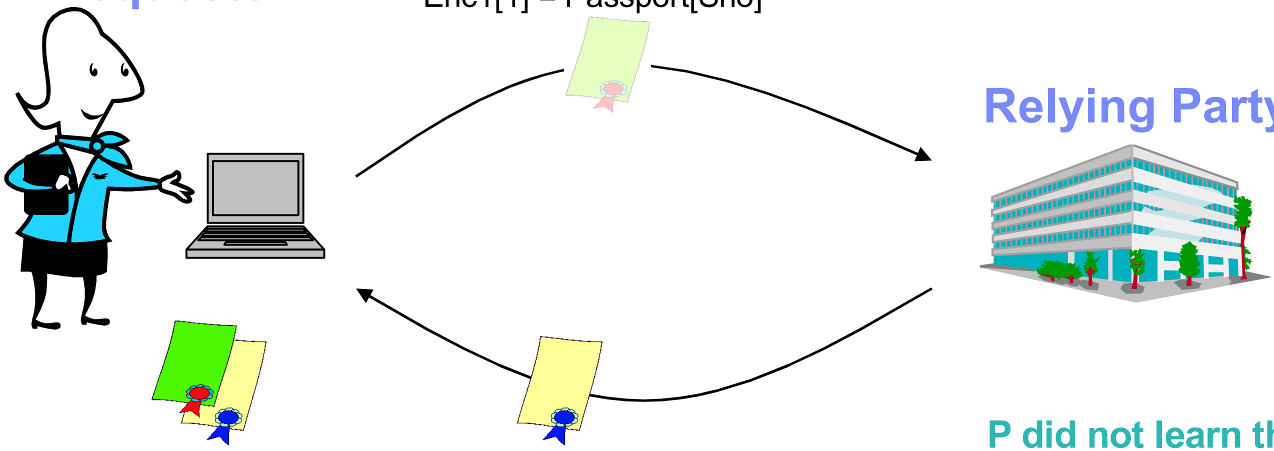



**Enc1[1]:** encryption with a specified PK of party T
Condition bound to it
Proof that the correct attribute is encrypted
Semantics: T may only decrypt if the condition is fulfilled

- Release of subset of attributes of a certificate
- Release of partial information of an attribute
- Commitments to attributes
- Encryptions of attributes

# Private Certificate Framework – Protocols (cont'd)

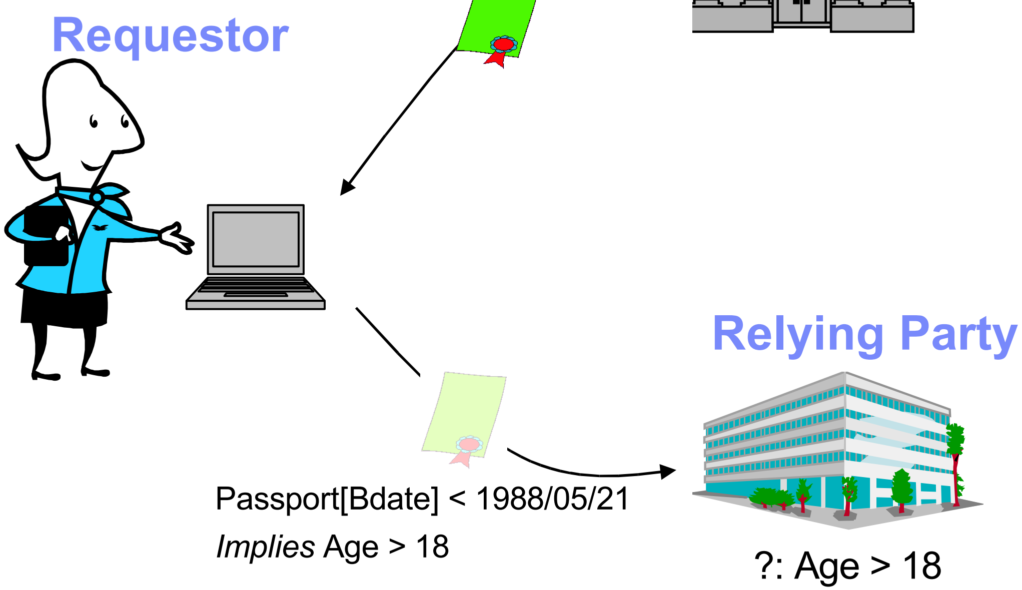



**P did not learn the Lastname attribute**

**Repository of private certificates**

**Open attribute exchange**

# Building Blocks for FIM




- **Security policy language**
- **Specification language**
- **Proof and issuance system**
- **Federation protocols (flows)**
- **Ontologies**
- **Software/hardware components implementing everything**

# Proof Protocol – Summary

- **Proof specification**
  - Statement over one or multiple certificates
  - “Assertion”
- **Cryptographic proof**
  - Cryptographic proof for the correctness of the proof specification
  - Verifies with respect to the issuers' public keys
  - Extension to framework of Bangerter et al. 2004

→ **This separation holds for all deployed approaches**

# Outline

- Scenario
- Protocols
- **Attribute Assertion Language**

# Proof Specification

- **Based on propositional logic**
- **Variables**
  - Attributes of certificates: E.g. SwissPassport[Birthdate]
  - Commitments: Comm4[3]
  - Encryptions: Enc6[1]
- **Predicates**
  - Predicates over variables
- **Connectives: AND, OR**
  - Connects the predicates

    E.g. Passport[Bdate] < 1988/05/21 OR Driverslicense
- **No negation**
  - Negation of specific predicates cannot be proved

    E.g., to NOT have a driver's license; no cryptographic proof tool available
- **Applicable to both interactive and non-interactive proofs**

# Predicates

- **Value domain of variables**
  - Subset of the integers [-2^a;2^a]
  - Strings of arbitrary length
- **Arithmetic comparison operators**
  - $>, \geq, <, \leq, =, \neq$
- **Predicates on n variables**
- **Arithmetic operators**
  - +, *, ^
- **Examples**
  - Bankstmt[Balance] > Comm1
  - Bankstmt1[Balance] + Bankstmt2[Balance] > 4000
  - Bankstmt[Subject] = Enc1[1]

# ⟨ ⟩-Annotated Predicates

- **Required for formulas containing OR connectors**
- **Prover uses ⟨ ⟩-annotation to specify the predicates the prover actually fulfills**
  - ⟨ Passport[Bdate] < 1988/05/21 ⟩ OR Driverslicense
  - Enc1[1] = Passport[Sno] OR ⟨ Enc1[1] = Driverslicense[Sno] ⟩
  - Only applied to prover's specification
    - → OR proofs conceal this information
- **For each ⟨ ⟩-annotated predicate, the prover must be able to fulfill the predicate**
- **There must exist one DNF clause where all predicates are annotated with ⟨ ⟩**

# Uninstantiated Variables

- **Instantiated variables**
  - Attributes of certificates, commitments, encryptions
  - Are instantiated through the attribute values of certificates, commitment openings, and plaintexts to encryptions
- **Uninstantiated variables**
  - Attributes of certificates, encryptions, commitments
- **Instantiation semantics**
  - Instantiation is specified by predicates
  - OR connective leads to interesting instantiation semantics
    - Variables are instantiated through the predicates that are ⟨ ⟩-annotated
    - Variables that appear only in non-⟨ ⟩-annotated predicates are instantiated with a random value
  - E.g.: Enc1[1] = Passport[Sno] OR ⟨ Enc2[1] = Driverslicense[Sno] ⟩

# Comprehensive Example

- **Private certificates**
  - USPP: United States passport
  - EUPP: European Union passport
- **Proof specification**
  - ⟨ Enc1[1] = USPP[Sno] AND Enc1[2] = 1 AND Enc2[2] = 0 ⟩

    OR

    Enc2[1] = EUPP[Sno] AND Enc2[2] = 1 AND Enc1[2] = 0
- **Encryptions**
  - Enc1 = (USPP[Sno],1), encrypted with PK_U
  - Enc2 = (rand, 0), encrypted with PK_E

# Conclusion

- **Defined new building blocks for identity federation**
  - → **Natural model for attribute exchange**
  - → **Better privacy in attribute exchange**
  - → **Weaker trust assumptions**
- **Further work**
  - – Ontologies
  - – Security policy language