Privacy enhanced authorizations and data handling

Ernesto Damiani, Sabrina De Capitani di Vimercati, Pierangela Samarati
Dipartimento di Tecnologie dell’Informazione
Università degli Studi di Milano
samarati@dti.unimi.it
Categories of policies

- **Access control policies** govern access to service and release to data stored at some service.
- **Release policies** govern release of personal private information (properties/credentials).
- **Data handling policies** define restrictions on secondary use of PII.
- **Sanitization policies** regulate the dialog between parties to protect sensitive policy information.

Existing standards

Access control

- eXtensible Markup Access Control Language (XACML)
- Enterprise Privacy Authorization Language (EPAL)
- WS-Policy
- XrML

Secondary use

- Platform for Privacy Preference (P3P)
Convergence between AC and PP models and languages

• AC departs from traditional authenticate/authorize approach
• Provisions for coordinated evaluation of different policies (client, server, data respondent, ....)
• Support for preferences->conditions mappings
  • PP declaratively expresses privacy preferences in a human-understandable way
  • AC states access conditions on user data in a consistent way w.r.t. privacy preferences
  • May be done explicitly via language mappings (e.g. XACML privacy profile) or implicitly (e.g. as part of an operational semantics of PP)
  • Both techniques have pros and cons
• Support for client-side and server-side access restrictions
• Support for authorizations depending on partial identities
• Support for new categories of conditions
  • Location-based, trust, purpose, obligations, data handling etc.
  • Raise privacy concerns in the evaluation of AC conditions
  • Raise policy disclosure concerns
Language extensions

• Integration with encryption
  • Evaluation of conditions based on ciphertext or signed assertions over ciphertext
• Support for context representation and reasoning
• New context-related predicates (e.g. LBS)
• Delegation
Encryption-aware language: issues

• An encryption-aware language needs
  • Support for conditions/predicates requesting encrypted data
  • Definition of a syntax and semantic for encryption-based statement
  • Definition of an infrastructure for cryptographic credentials evaluation
  • Enhanced representation of parties portfolios
Context representation/reasoning (1)

- Definition of ontologies for context inferences
- Definition of ontologies for policies inferences
- Policies expansion through ontologies definition
- Ontology-based evaluation of conditions

Example

- `>age(User,18) :- driverLicence(type="B", issuer="ItalianPublicAdministration")`
Context representation/reasoning (2)

• Enhanced Context representation containing
  • Approximate information
  • Time-variant information
  • Uncertain information

Examples
• Users position
• Mobile information
• Facial expression
Definition of location-based conditions

• Ability to express, evaluate and enforce access control policies based on location information

Location-based conditions examples

• inArea(User, “Room1”)
• disjoint(User, “Italy”)
• density(“Room7”, 1, 10)
Context and Privacy Preferences

• Definition of a new category of privacy preferences regarding time-variant and approximate context information
• Privacy preferences will affect the evaluation of conditions based on enhanced context information

Preferences examples

• When evaluating LBS conditions on me
  • *Determine my location with a minimum accuracy of 10 meters*
  • *Determine my location degrading the measure by a certain percentage, with respect to location technology accuracy*
R&D challenges context awareness (1)

Context information is a set of metadata clearly identifying entities of interest in the domain.

A well-understood and shared context representation and a secure infrastructure making it available provide:

- Capability of parties to negotiate common knowledge and exploit a shared vocabulary.
- Capability of parties to verify policy conditions.
R&D challenges context awareness (2)

- **Protect privacy of context information.** User context information should only be provided to authorized entities.
- **Describe entities via context ontologies.** User context information must be made accessible by entities, dealing with its semantics in a clear and unambiguous way.
- **Develop a metadata distribution architecture.** Context information should be made available to any authorized entity at any time.
  - Still unauthorized information leaks should be prevented.
R&D challenges context awareness (3)

• **Semantic portfolio.** Support controlled access to contextual resources subject to user-specified privacy constraints
  
  • Existing standards (e.g., OWL Semantic Web reasoning engine, location tracking functionality, etc.) need to be combined with new enforcement techniques
Secondary use

Constraints on secondary use

• **Agreed** between the parties (server/client)

• Expressed within the **rules** or as **separate rules**

• Need to be obeyed (propagated/satisfied) by the policy of the server
Data handling policies (1)

Specify how PII is used and processed

- Attribute-based language
- Support for purposes
- Support for provisions and obligations
- Support for disputes and remedies (human readable)
- Different types of specifications
  - Server-side
  - Customized
  - User- and server-side
Data handling policies

- Data handling policies composition
  - Data handling policies defined at different level of the data schema
  - Support for multiple purposes
- Automatic negotiation of preferences between users and servers
  - Servers propose a set of policies
  - Users, automatically, customize through their preferences
Data Handling Policies (3)

- **Data Handling policies matching**
  - Definition of compact policies to boost data handling policies comparison and evaluation
  - Definition of policies templates customizable by the end users
R&D challenges secondary use

- Management of data handling policies lifecycle
- Definition of policies allowing the protection of the users data after chains of releases
- Support for machine readable remedies and disputes
Conclusions

- Current standards are evolving independently to address open issues
- Some (not all) of the aspects are being covered
- But: putting the different aspects together requires some重新思考
- Some aspects not covered by current standards
  - Data handling
  - Credential/declaration management
  - Support for anonymity/privacy
  - Support for policy communication (sanitization)
  - Support for negotiation
Thank you for your attention
Backup slides
• **declarations:**
  - information uttered by the party and not certified by any authority (e.g., identity, address, hobbies)

• **credentials:** digital certificates \((c,K)\)
  - \(c\): signed content (credential name, attribute list)
  - \(K\): public digital signature verification key

• **built-in mathematical predicates**

• **conditions:**
  - state
  - trust
  - location
  - ........
We assume a **semi-structured** organization of credentials

- **Credential term**: expression of the form `credential_name(attribute_list)`
  - `credential_name`: name of the credential
  - `attribute_list`: list of elements of the form “attribute_name=value_term”

**Example**

- `driver-license(name=“John Doe”)`
Authorization

\langle subjects \rangle

CAN \langle actions \rangle

\langle objects \rangle

[FOR \langle purposes \rangle]

[IF \langle conditions \rangle]

[FOLLOW \langle obligations \rangle]

- **subjects**: boolean expression of credentials and declarations
- **objects**: boolean expression of conditions on metadata
- **conditions**: boolean expression of generic conditions
Restriction

\(\langle \text{subjects} \rangle\)

CAN \(\langle \text{actions} \rangle\)

\(\langle \text{objects} \rangle\)

[FOR \(\langle \text{purposes} \rangle\)]

[ONLY IF \(\langle \text{conditions} \rangle\)]

[FOLLOW \(\langle \text{obligations} \rangle\)]

- **subjects**: boolean expression of credentials and declarations
- **objects**: boolean expression of conditions on metadata
- **conditions**: boolean expression of generic conditions
Some support of **variables**:  
- **user**: user requesting access  
- **object**: data to be accessed  

Support of any kind of **predicates** (provided evaluation):  
- **dynamic**: defining conditions that can be brought to satisfactions at run-time processing of the request  
- **trust**: assessing trustworthiness of server  
- **location**: making enforcement dependent on location of requestor  

Support of ontologies and abstractions (subject, object, portfolio ontologies)
Subjects

<subject_id> WITH <subject_expression>

- **subject_id**: identifier (individual or group) defined in the ontology. Allows indexing of access rules
- **subject_expression**: boolean formula over credentials and declaration terms. It uses predefined variable user to refer to actual requestor

**Examples**

- declaration(user.name = "Bob", user.age > 18)
- credential(passport(user.nationality = "Italian"), K₁)
<object_id> WITH <object_expression>

- **object_id**: identifier (individual resource or class thereof) defined in the ontology. Allows indexing of access rules

- **object_expression**: boolean formula over credentials and declaration terms. It uses predefined variable `object` to refer to actual requestor

**Examples**

- declaration(`object.creator = “Bob”`)

- declaration(`object.creation_date < “1971”`)

- declaration(`object.creator = user`)
Conditions

- boolean expression of conditions
- each term has the form `predicate_name(arguments)`
- different types of conditions can be stated inside a rule:
  - trust-based conditions
  - location-based conditions
  - state-based conditions

Examples

- `filled_in_form(user, "form1")`
- `payment(user, "subscription1")`
Examples of rules (1)

Researchers CAN access Restricted Datasets IF declaration(payment(user, Restricted Datasets))

Researchers can access restricted dataset if they have paid for the access
Examples of rules (2)

Any-User WITH
credential(DriversLicence(Permit.CarPermit="true", Issuer.Country="IT"), K_{DL}) AND declaration(User.Age=17))

CAN rent_a_car
Mercedes WITH type="CLK"

IF credential(eCoin(Value>100Euro), K_{EC}) AND declaration(in_area("Italy"))

Users older that 17 who have a valid italian driver licence can rent a Mercedes CLK if they have provided an eCoin for more than 100€ for the access and they are in Italy
Examples of rules (3)

Any-User WITH declaration(user.citizenship="EU") CAN download NationalSurvey IF metadata.downloadable = “yes”

European citizens can download national surveys if they are marked as “downloadable”