Flexible and Usable Policies

P.A. Bonatti, University of Napoli “Federico II”
and coordinator of
**REWERSE** is one of the 2 EC FP6 Networks of Excellence devoted to the SW

- Ranked 6<sup>th</sup> among 1-1.5K IST proposals in FP6
- 27 institutions (academy + industry)
- > 5 million euro
- focussed on rule-based techniques
- policies identified as crucial area
- WG I2 devoted to policy specification, composition, conformance
- using a broad notion of policy – today we focus on privacy
From Access control to Privacy policies

- Same goal
  - Protecting confidentiality

- Local and Remote access control
  - Sticky policies
    - Nonstandard integration & (law) compliance needs

- Under extreme flexibility requirements
  - Interacting with all sorts of services (*interoperability*)
    - Navigating the Internet, Pervasive computing, ...
  - In extremely dynamic contexts
    - New business models, virtual organizations
    - Fast & easy composition / integration / harmonization
    - Pervasive computing environments – time & location
    - Walking through an airport, sharing info with new friends, using airport services, ...

- New, more expressive languages are needed
Importance of user awareness & control

- Automated information negotiations
  - Essential for usability, but
  - Users may loose control
  - What information is released and when?
  - Explaining policies & negotiations

- No *one size fits all* information release policies
  - In security default policies have already proved their limits
  - Policy personalization should be at everybody's reach
  - User-friendly personalization for untrained users
Mechanisms for Privacy Policies extend standard mechanisms

- Trust-based disclosure decisions, info negotiation
  - Example: Credential negotiation
  - From need-to-know to need-for-goal (purpose based)
  - Balancing risks and benefits
  - Minimizing the amount and sensitivity of disclosed information
  - New languages before new enforcement mechanisms

- Policy matching and comparison for
  - Disclosure decisions (compliance with privacy requirements)
  - Service selection

- Policy negotiation
  - Conflict resolution
  - Preference handling
  - Incentives to information disclosure

- Explanation facilities & Controlled NL Front ends
Semantic processing
more than interoperability

- Heterogeneous policies
  - Ontologies for interoperability & integration

- One policy, many uses
  - Enforcement, negotiation, comparison, explanation, ...
  - Policies as knowledge bases
    - b.t.w. policies may contain ontologies...
  - Declarative policy languages are needed
    - enhance also user awareness & control...
PROTUNE
REWVERSE's Trust Negotiation Framework
Current state & work in progress

To be released by Dec 2006 on sourceforge
- Locally enforced, trust-based privacy policies
- Policy-driven negotiations
- Explanation facility
  - Policies and negotiations

Sticky policies (nonlocal enforcement)
- Joint work with M. Winslett and C. Zhang in CCS 2005

Future work
- More on sticky policies
- Policy comparison & negotiation
- Protune on small mobile appliances
Policy-driven Negotiations
uniform treatment of all policies


Step 1: Alice requests a service from Bob
Step 2: Bob discloses his policy for the service
Step 3: Alice discloses her policy for VISA
Step 4: Bob discloses his BBB credential
Step 5: Alice discloses her VISA card credential
Step 6: Bob grants access to the service
The policy language
how to represent Bob's policy

- Protune adopts rules (natural!)
- Arbitrary boolean combinations of items
- Restrictions on their attributes
- Possibly recursive conditions
  - Credential chains (~ transitive closure)

```
allow(download(paper1.pdf)) ←
id(Document),
Document.name : User,
credit_card(Card),
Card.name : User.
```
The policy language
how to represent Bob's policy

- Policies may define **concepts**
  - Policies may include ontologies
  - Released along with requirements to explain them in a machine readable format

allow(download(paper1.pdf)) ←
  id(Document),
  Document.name : User,
  credit_card(Card),
  Card.name : User.

Policies contain concept definitions

id(Document) ←
  credential(Document),
  Document.type : T,
  Document.issuer : CA,
  isa(T,id),
  trusted_for(CA,id).

More concepts
The policy language
Alice's privacy policy

- Expressed in a uniform way

release(visa_card, Requester) ←
  BBB_member(Requester),
  purpose : purchase(Item),
  Item.cost > 100.

Decision based on trust, purpose and risk

Ontology

BBB_member(Requester) ←
  credential(C),
  C.issuer : “BBB”,
  C.public_key : TheKey
  challenge(Requester, TheKey).
Minimal shared language

- Defined concepts are eventually grounded on a small number of primitives
  - X.509 credentials
  - declarations (similar to web forms)
  - connect(URI), challenge (actions)

- Based on which a negotiation engine may
  - Submit the required info (if the disclosure policy permits)
  - Execute an action (if the policy permits)
  - Query the user first (if the policy says so)
  - Refuse to comply (the negotiation may fail or proceed differently)
Minimal prerequisites for application

A common understanding of

- Rule semantics
- The shared primitives
  - Credential format (X.509 standard), declaration forms, connect and challenge
- No further semantic infrastructure needed
- Lightweight reasoning (Horn clauses)

Claim

- Technologically feasible
- Even on small mobile appliances
Policies are not (only) passive objects

Policies may specify

- Event logging
- Communications and notifications
  - e.g. query-the-user
- Workflow triggering
  - such as (partly) manual registration procedures

i.e. Policies may specify actions

- To be interleaved with the decision process
Strong, Soft, and Lightweight Evidence

Trust sources

- **Strong evidence**
  - e.g. *digital credentials* (id, credit cards, subscriptions)

- **Soft evidence**
  - e.g. *numerical reputation measures*

- **Lightweight evidence**
  - e.g. “*accept buttons*” (copyright/license agreements)

They can be integrated for balancing:

- trust level
- risk level
- computational costs
- usability (fetching credentials, personal assistants)
Strong, Soft, and Lightweight Evidence

How can individuals prove their eligibility?

- **Strong evidence**
  - e.g. digital credentials

- **Soft evidence**
  - e.g. numerical reputation measures

- **Lightweight evidence**
  - e.g. “accept buttons”

They should be integrated for balancing:

- trust level
- risk level
- computational costs
- usability (fetching credentials, personal assistants)

E.g. micropayments vs. buying plane tickets
Exploiting “external” systems

Decisions need data, information, and knowledge

- Each organization has its own
  - Already available through legacy software and data
  - A realistic solution *must* interoperate with them

- Third parties
  - Credit card sites for validity checking
  - Credential repositories

- Variety of web resources
- Protune: special syntax for external calls
Explanation mechanism

Main challenge:

- Finding the right tradeoff between
  - Explanation quality (2nd generation explanation facilities)
    - Remove irrelevant information
    - User-friendly denotation of internal objects
    - User-oriented description of reasoning
  - Framework instantiation effort
    - The framework needs to be adapted to each application domain
    - Expensive in 2nd generation EF (ad hoc KB and engine)
    - Reduce the need for specialized staff
- Computational load
Protune's explanation facility

- **Supported queries**
  - Why / Why not (for explaining negotiations)
  - How to (for explaining policies)
  - What if (for validating policies)

- **Explanations can be built on clients**
  - Almost no overhead on servers
  - Scalable approach
Controlled natural language specs

- We are aiming at specifications like

  *Credit cards can be released to BBB members if the cost of the purchased item is at least 100 euros*

- Based on an evolution of the Attempto system for controlled natural language processing

  http://www.ifi.unizh.ch/attempto/
Protune is evolvable

- Two powerful mechanisms
  - Rule libraries
  - Metapolicies
- For language extensions
- For controlling negotiations
DISCUSSION

[ More on http://reworse.net/i2/ ]
Why-not demo
sample screenshot

it is not allowed to download paper_0123.pdf because:

- Rule [3] cannot be applied:
  - paper_0123.pdf is not public [details]
- Rule [4] cannot be applied:
  - I find no User such that the User is authenticated [details]
- Rule [5] cannot be applied:
  - I find no User such that the User is authenticated [details]
  - I find no User such that the User paid for paper_0123.pdf [details]

Policy file
Why-not demo
sample screenshot

it is not allowed to download paper_0123.pdf because:

- Rule [3] cannot be applied:
  - paper_0123.pdf is not public [details]
- Rule [4] cannot be applied:
  - I find no User such that the User is authenticated [details]
- Rule [5] cannot be applied:
  - I find no User such that the User is authenticated [details]
  - I find no User such that the User paid for paper_0123.pdf [details]

Policy file
Why-not demo
sample screenshot

the User is not authenticated because:

- Rule [7] cannot be applied:
  - I find no Credential such that the Credential is an id [details]
- Rule [8] cannot be applied:
  - I find no Form such that the Form is a declaration [details]
- Rule [9] cannot be applied:
  - the procedure on http://lol.com/register.php has not (yet) been successfully completed [details]

Policy file
Why-not demo after one more step...

the Card is not a valid credential because:

- Rule [19] cannot be applied:
  - c012 is a credential whose issuer is Open University

**but**
- I find no Key such that the Key is the public key of Open University

Policy file