SWPW 2005 – Galway, Ireland

REWERSE reasoning on the web

Web Policy Zeitgeist Panel

Piero Bonatti, November 7th, 2005

REWERSE WG I2 - policy language

Topics

1) Policies are not just about security or privacy

2) Policies are not islands

3) Policies must be integrated with ontologies

4) Policies must be norms

5) Policies must be X

1. Policies are not just about security / privacy

Aren't these policies, too?

- Business rules
- Quality of Service directives
- Visa eligibility criteria
- ...

All these policies make decisions

- grant/deny access, establish eligibility (visas), make discounts

Based on similar pieces of information

- user / agent / server properties
  - age, nationality, profile, identity, reputation, certifications...

Policies are not (only) passive objects

Policies may specify

- Event logging
- Communications and notifications
- Workflow triggering
  - such as (partly) manual registration procedures
- ...

So policies are about

- *Decision support + behavior specifications*
  - **declarative** (despite the word “behavior”)

2. Policies are not islands

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

Short term solution

- Mediation and integration techniques
  - maybe wrappers
    - instantiation efforts needed
  - *a challenge for SW interoperation approaches in the long run*

2. Policies are not islands

What about standard security mechanisms?

- They are so robust and efficient
  - Border routers
  - Firewalls
  - DBMS access control, maybe Web Server access control (well...)
- Can't we exploit them in our smart frameworks?
  - Further advantage: avoid bottlenecks
    - Centralized security monitors for high-level policies (XACML)
  - Standard mechanisms are already widely supported

credits: Arnie Rosenthal

3. Policies must be integrated with ontologies

Why, policies are *already* integrated with ontologies!

- Rules immersed in the policy
  - Definition of authenticated user
  - Definition of accepted ID
  - Definition of accepted credit card
  - Trusted CA
- Currently formulated in terms of **credentials** and **declarations**
  - x.509
  - web forms
  - rules
  - *No complex prerequisites!*

3. Policies must be integrated with ontologies

Challenges:

- Interoperability on a larger scale
  - interplay with legacy software and third parties
  - lightweight evidence can be based on any web contents
  - how to explain requirements in a machine-understandable way?
  - a standard semantic web issue – ontologies
  - **still lightweight?...**
  - **infrastructural prerequisites**

4. Policies must be norms

How strict? Which logic? A lesson from IMPACT:

- **Deontic** “Agent Programs”
- Approach 1:
  - what is **possible** determines a space of allowed actions
  - what is **obligatory** determines concrete actions
  - Redundant – eventually we didn't use possibility
- Approach 2:
  - “obligatory” implies “possible”
  - whatever concrete action you make becomes automatically possible
  - Possibility is useless
- Should we really start with “traditional” approaches?

4. Policies must be norms

In our scenarios norms establish:

- A space of **possible choices**
  - Please disclose a library card or a driving licence
  - Release VISA or MASTERCARD credit card
- Maybe **preferences**
  - A SSN is more sensitive than a library card
- And **constraints**
  - Date of birth and address should not be disclosed together
- It is not immediately clear which is the right choice
- **Is it really so much about deontic logic?**
  - Optimization, games, ...?

5. Policies must be X

X = ACCESSIBLE

- Users should understand the policies applied by the systems that users interact with
- and users should be able to personalize those policies
  - With pre-defined policies: machine violated in **5 min.**
  - With personalized policy: secure for **2 weeks** (end of experiment)
  - Know what your policy does *not* check (and avoid pitfalls)
  - A social problem:
    - Everybody's machine is on the internet
    - Millions of computers can be exploited for attacks
      - *by taking advantage of the users' lack of technical competence*

5. Policies must be accessible

Challenges:

- Make rules intelligible to the common user
- Use natural language?
  - Suitably restricted to avoid ambiguities
- Explain policies and system decisions
  - A classical AI problem – perfectly in line with SW
- *Encourage people to personalize their policies*

Conclusions (?)

Polices & Trust Negotiation are important because:

- They might prove the effectiveness of semantic web ideas in the **short term**
  - Nontrivial shared ontologies based on X.509 + rules

Rule-based policies are important because:

- They might alleviate **users' lack of awareness** about their system's policy
  - The main cause of today's world-scale security problems

It is important to keep promises and deliver solutions