Accountable Information Usage

Massachusetts Institute of Technology
Lalana Kagal

18 Sep, 2013
Team Profile

Decentralized Information Group

Hal Abelson (PI)
Tim Berners-Lee
Lalana Kagal (PI)
Joe Pato
Gerry Sussman
K. Krasnow Waterman
Daniel Weitzner

PhD students
Masters students
Undergraduates
Customer Need

• Useful for policy compliant information sharing and exchange in decentralized environments such as
  – DHS Fusion
  – Emergency responders
  – Health Information Exchange
  – Juvenile Justice

Approach

- **Information Accountability**: When information has been used, it should be possible to determine what happened, and to pinpoint use that is inappropriate. This requires the ability to
  - express information use policies
  - monitor / reason over information use
  - provide redress
- Moves focus from “what you know” about me to “what you do with it”
- Not just access control but usage control

[Diagram of protection mechanisms]

Graphic courtesy Ilaria Lecardi
Approach

• An accountable information system is able to determine
  – Whether each use of data was/is compliant
  – with the relevant rules (laws, regulation, or policy)
  – for particular data, parties, and circumstances
Technologies

- (Distributed) Domain knowledge: Linked Data technologies
  - Web standards for developing and maintaining structured data
- Rules: AIR Rule Language
  - Production rules (if-then-else) over any set of vocabularies
  - Modular development of rules and ability to traverse from rule to rule (linked rules)
- Reasoner: AIR Reasoner
  - Fetch data from any addressable source as needed during processing
  - Produce plain language explanations
  - Produce full statement of dependencies
- System
  - Input, rules, and output are all in the same form (Linked Data)
Data Model

• Linked Data provides information management at Web-scale
  – By leveraging Web protocols and technologies
  – Network Effect: Exponential value of being part of the Web
  – All specifications for Linked Data are open Web standards
  – Reuse of existing well developed and studied Web technology
  – Applications
    • Biomedical research
    • Counter terrorism
    • Providing transparency in government
Air Rule Language & Reasoner

- machine-readable rule/policy language
- based on Linked Data technologies
- focused on justification generation, ease of specification, rule reuse, and builtins for use of distributed data
- Has been used in various projects for information accountability, policy compliance, trust frameworks, access control, etc.
Benefits

• Automates policy compliance checking of information transactions
• Allows information to be shared safely
• Operates in an inherently decentralized fashion
  – Fetches data and rules from distributed sources as needed during processing
• Instead of usual YES/NO answer that policy tools provides, the system provides justifications
  – Plain language explanations
  – Full statement of dependencies
Current Status

• Current project
  – FICAM Interoperability
  – HIPAA scenario demonstration (Added task)
  – Reasoner Scalability
Task 1: FICAM Backend Attribute Exchange Interoperability

- Demonstrated ability to interoperate with DHS Identity Management Testbed
- Able to serve appropriate certificates, create appropriate signatures
- Able to fetch the Distinguished Name from JHU
- Able to convert RDF -> SOAP and SOAP -> RDF
- MIT tools able to use the JHU served sender and receiver attributes in the reasoning to achieve decisions

FICAM Backend Attribute Exchange
Interoperability – System Design
Task 2: HIPAA Scenario Development

- Showing one path of the HIPAA Privacy Rule
- Request for disclosure without patient consent or referral
  - Used when patient is unconscious or incompetent
  - Used when patient-signed consent form is unavailable
- Using as many real components as possible
  - Data standards
  - Anonymized records

Scenario (developed in collaboration with SHARPS performer Vanderbilt)
- Dr. Special requests disclosure of Mr. Sick’s primary care electronic health Record from General Hospital
HIPAA Scenario – System Design
Task 3: Reasoner Scalability

- AIR reasoner is a production-rule system in python
- Forward chained reasoning
- RETE algorithm for pattern matching
- Truth Maintenance System (TMS) for dependency tracking
  - the reasoner maintains the premises (rules and facts) of each conclusion
Next Steps

Year 2:
• NIEM interoperability
• Enhanced capabilities for handling incompleteness
• Pilot Implementation/Demonstration Project
Contact Information

Lalana Kagal
lkagal@csail.mit.edu

Hal Abelson
hal@csail.mit.edu