

Policy-Driven Systems for Security, Privacy and Governance: Semantic Analysis

Technology Perspectives
TC009SN

Abstract

- > Access control systems depend on the administrator formulating sets of policies determining the conditions for access. These policies can include many rules, and can change over time: additions, modifications and deletions can be performed. When administrators perform these operations, they can unintentionally compromise the integrity of the policy sets. Other problems that can occur are related to possible incompleteness or to rule overlap. Incompleteness, inconsistency and other potentially harmful policy integrity problems will be called 'semantic anomalies'. We will demonstrate a prototype Policy Assistant that flags anomalies in sets of EEM policy rules, thereby assisting administrators in maintaining their integrity (**see Demo Booth 333**). The same principles used for the Policy Assistant can be extended to the semantic analysis of privacy and E-Governance systems. Realistic examples are given throughout.

Biography

> **Luigi Logrippo**

Professor, Université du Québec en Outaouais, Canada

- > Luigi Logrippo's first degree was in Law, but he moved quickly to Computer Science when this was still a very new field. He then had a research career in Formal Techniques in software design with application to telecommunications systems. Recent developments in software methods for security, privacy, and e-governance make it possible for him to make good use of the various aspects of his expertise.
- > Luigi has degrees from the University of Rome, the University of Manitoba, and the University of Waterloo. Before moving to the UQO he was a professor of Computer Science at the University of Ottawa (Canada) for almost thirty years, as well as Department Chair there for two terms.

Agenda

- > Policy Systems
- > Access Control Systems
- > CA's EEM
- > Semantic Policy Assistant
- > Consistency, Completeness, Redundancy, Auditability, Conformance: Detection and Resolution of Semantic Issues
- > Issues with Delegation
- > High-level Enterprise Policies
- > Application to Privacy, Examples: PIPEDA, SOX
- > Future Application: E-Governance
- > Benefits and Summary

Policy Systems

- > Enterprises are directed by complex policies
 - Hundreds of them
 - Subject to change

- > These policies can be encoded and interpreted in order to realize enterprise goals (often through XML syntax)

- > Examples:
 - Firewalls
 - Security rules
 - Access control rules
 - Privacy rules
 - E-governance ...

Phases of Research

- > Analysis and implementation of access control systems
 - Immediate benefits
 - Reduction of risks to CA-EEM users
 - Help in maintaining large sets of policies
 - Benefits from future research
 - Compilation of policies from enterprise security goals
 - High-level control of policies by security administrator

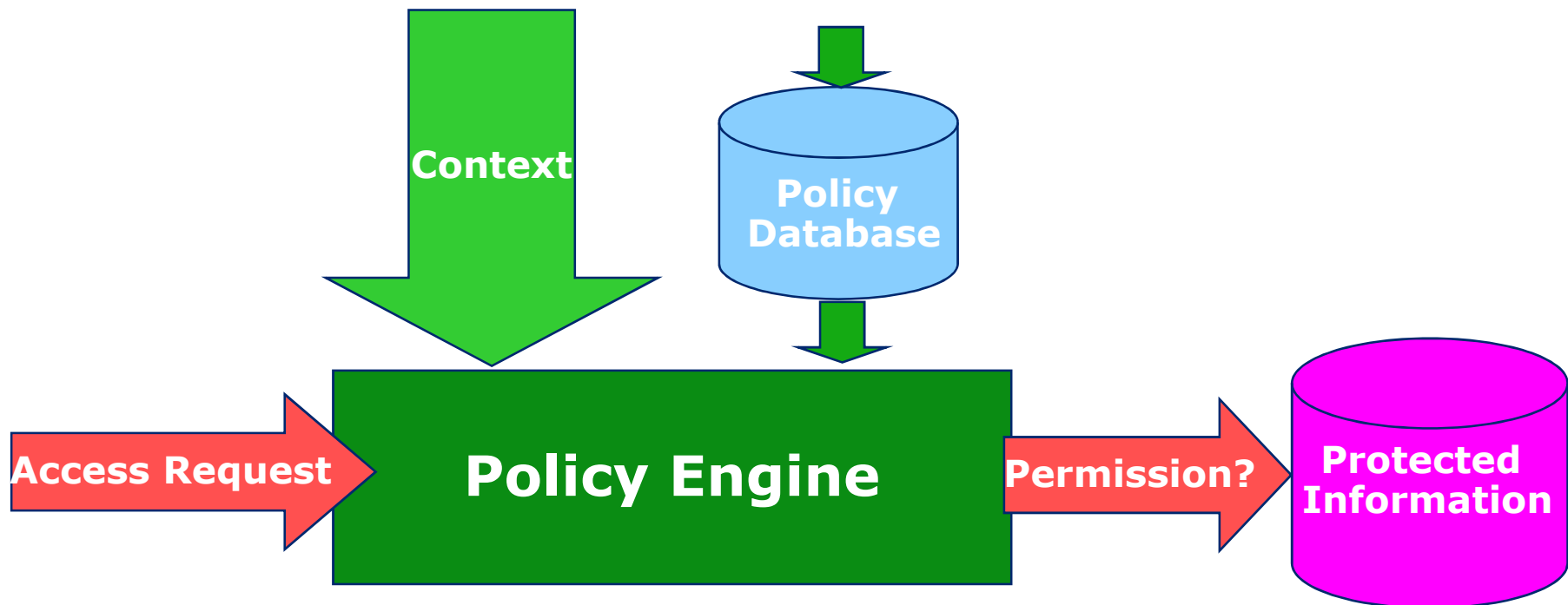
- > Analysis and implementation of security and privacy law
 - Enterprise compliance can be formally checked and audited
 - Laws and enterprise regulations will be clearer, better structured, directly related to each others

Positioning within CA's Strategy

- > This is a look at the future, an exploratory research project
 - Functionalities that we propose for implementation in EEM-like products in the future
- > CA does not (yet) have product plans associated with this project
 - We are outside CA's development path
- > The functionalities we propose have not, to our knowledge, been implemented in any similar commercial product, by CA or by the competition

Access Control Systems

For Hospitals, Banks, the Military

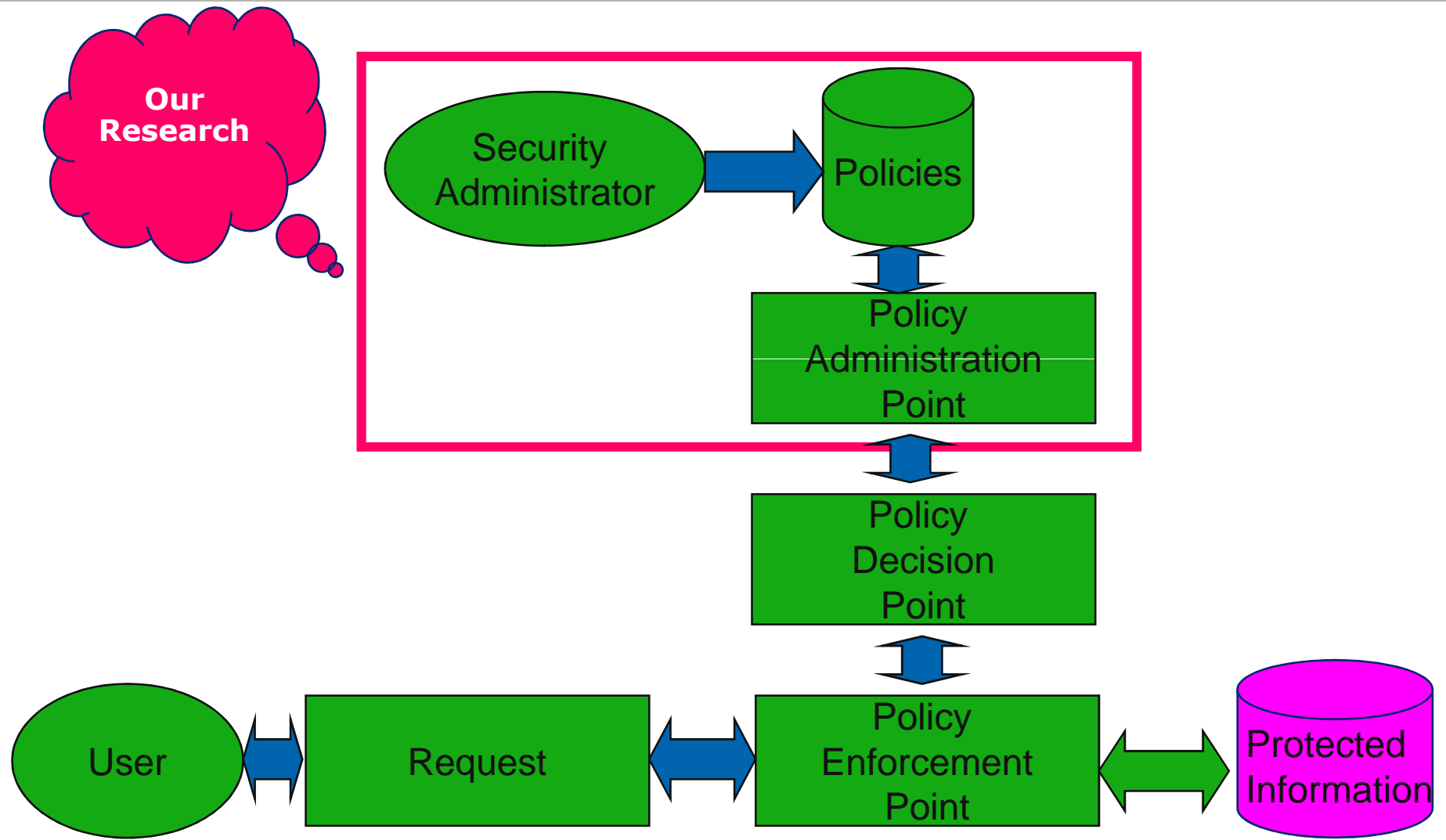


Ex. of **policy**:
if a nurse requests access to a patient file,
allow only if patient is in her ward

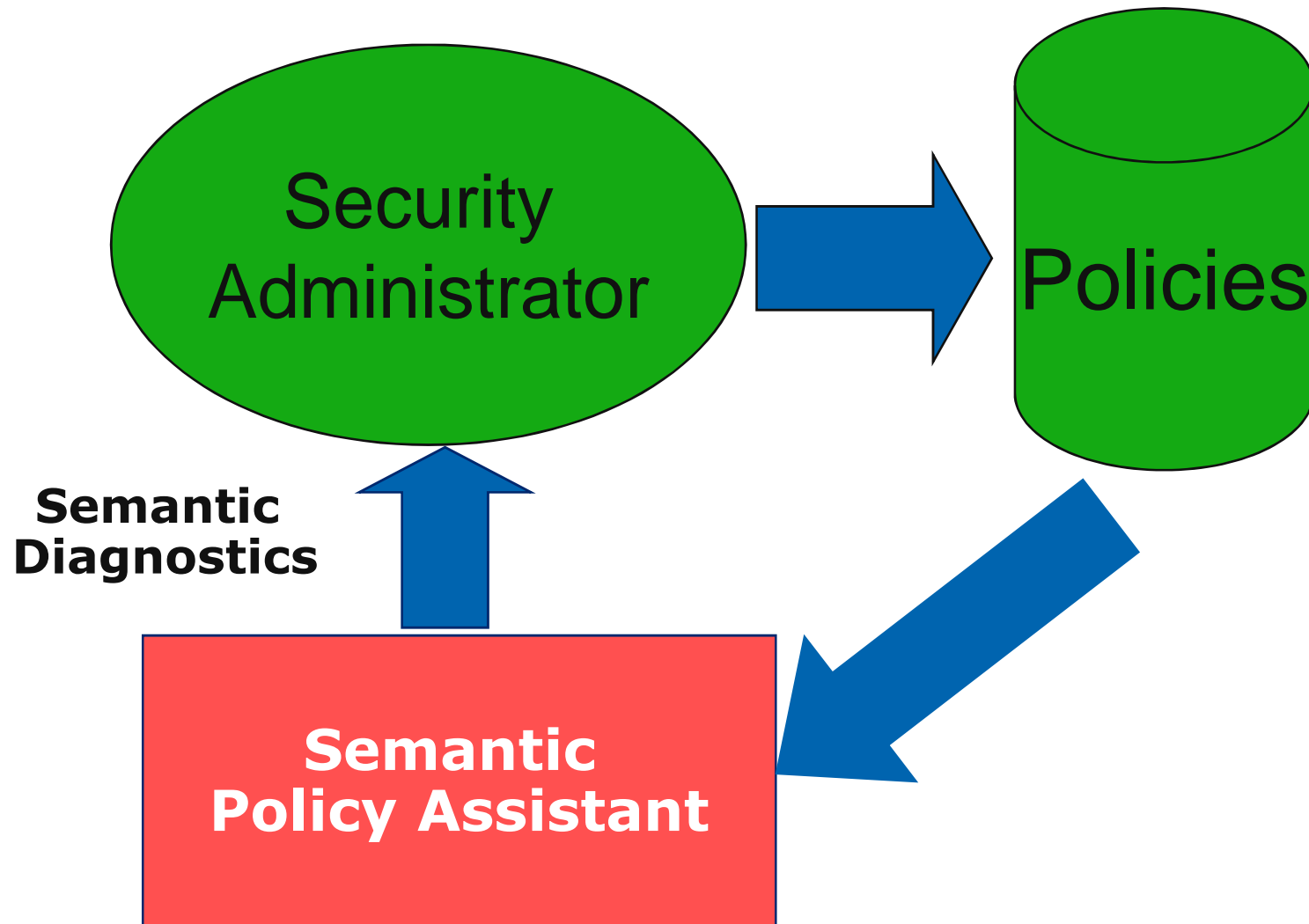
CA Embedded Entitlement Manager (EEM)

- > Allows the formulation of access control rules
- > Can be queried to determine if access should be granted in given cases

Detailed View



Zeroing In: The Semantic Policy Assistant



Semantic Diagnostics and Validation

For Policy Rules

- > Consistency
 - Do rules contradict each other?
 - This must be corrected
- > Completeness
 - Are all cases considered?
 - New rules may have to be added
- > Redundancy
 - Overlapping rules
 - Do housekeeping
- > Auditability and conformance
 - Do rules correctly implement laws and regulations?
- > All this constitutes **semantic policy validation**

Automation of Semantic Validation

- > Consistency, completeness, redundancy, conformance are logical issues
- > The relevant checks can be automated using formal methods and logic engines

Example of Inconsistency

- > A hospital can have many dozens of access control policies
 - They are updated over time to reflect changing situations
- > Policy 1 (added in March):
 - Alice **has access to patient records**
- > Policy 2 (added in September):
 - Nurses **do not have access to patient records** when patients are not in their ward
 - At the moment of adding Policy 2, administrator must be warned that may restrict rights given to Alice, a nurse, in Policy 1



See demo!

Resolution of Inconsistency

- > Tools such as CA EEM give priority to denials
 - With both policies, a nurse will be allowed to see a patient's record only if patient is in her ward
 - However this may not correspond to administrator's **intentions**
 - Administrator must be informed of inconsistency and asked to check
 - Demo will show how this can be done

Simple Delegation: Principles

- > Adding permissions that one did not have
 - No problem, that's what delegation is for
- > Adding permissions that were explicitly denied
 - Inconsistency
- > Adding permissions that one already had
 - Redundancy

Example of Delegation Inconsistency

- > Technicians cannot access patients' files
- > A doctor leaving for a trip delegates to a technician her access rights
 - EEM tool will first of all check for denials and will not grant access to technician
 - The result is not what delegator expected
 - So flag anomaly
 - At the time of delegation



See demo

Delegation Can Violate Enterprise Policies

- > Suppose enterprise policy (separation of duty):
 - No employee can have access to both items:
 - Account name - Account balance
- > Policy 1:
 - Bob can read account names
 - Bob can delegate this right
- > Policy 2:
 - Mike can read account balances
- > Policy 3:
 - Bob delegates his rights to Mike
 - **Attempt to add Policy 3 should generate warning**



EEM Policy Evaluation Algorithm

- > Check for explicit denials
 - "Matching" appropriate policies
 - Evaluating the matched policy filters
- > If no explicit deny was found:
 - Check for explicit grants
 - If no explicit grant was found:
 - Consider delegations
- > These rules resolve some inconsistencies, but result may not correspond to administrator's goals

Context Information

- > In order to see certain inconsistencies, it may be necessary to use context information:
 - Policy 1: *Only doctors can see the medical records of patients, together with their names*
 - Policy 2: *If a doctor has access to certain information, her supervisor can also access it*

- > In the context of some hospitals, it can perhaps be seen that
 - Some doctor's supervisors are not doctors
 - Leading to an inconsistency with Policy 1

Completeness

- > Completeness can be checked with respect to criteria
- > E.g., in a hospital, different employees can access different rooms at different times
- > Check that *at any given time there is a head nurse that can access room AB-1234*
- > If not, signal to administrator and request correction

Completeness and Compliance: Chinese Wall

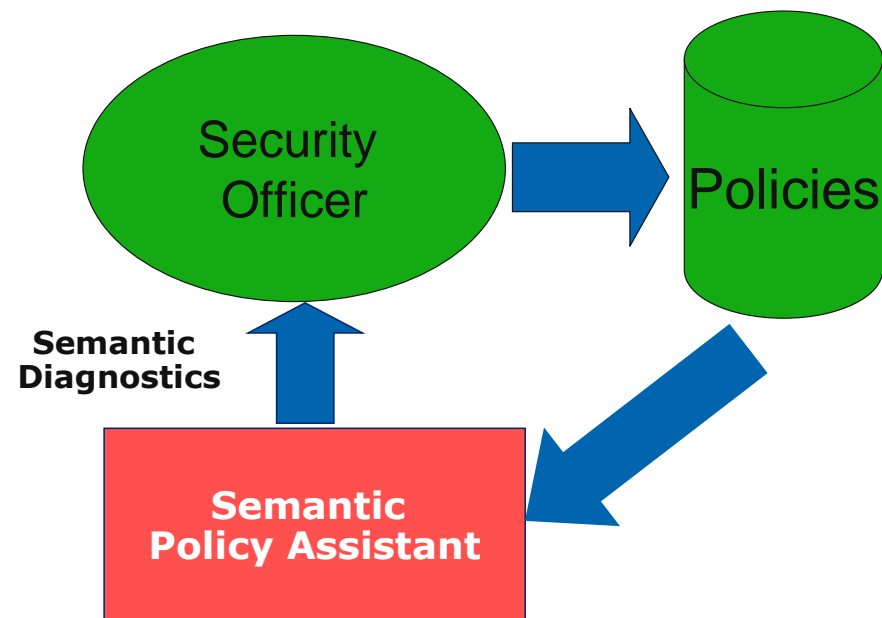
- > In a consulting company
 - Department A consults for Company X
 - Department B consults for Company Y
- > Companies X and Y are in competition
 - Assure that no information can be leaked between Depts A and B
- > Check that all the rules are in place to prevent such leaks
 - E.g., employees working for Department A cannot read any files owned by employees working for Department B
 - And vice-versa, etc.

Resolution of Incompleteness

> Requires feedback of administrator

Immediate Goal of Our Project

- > Develop a *semantic policy assistant* to check consistency and completeness of policies
- > And help administrator to resolve problems detected



Principles of Solution

- > For finding inconsistencies in sets of rules one can use:
 - Boolean satisfaction algorithms (SAT)
 - Constraint satisfaction algorithm
 - Which essentially are equivalent
- > They look for solutions, or determine insolvability, for sets of logical constraints
- > Abstract policy execution to guarantee that all cases have been covered

Example

- > Satisfy the following set of constraints:
 - Professors can access student files
 - Secretaries can access professor files
 - Some individuals can access both student files and professor files
- > A SAT or constraint satisfaction algorithm will find that all three constraints are *satisfied*
 - By choosing as “individuals” secretaries that are also professors
- > Adding a fourth constraint:
 - Secretaries cannot be professors and vice-versa
- > Will make the set of constraints unsatisfiable

Computational Difficulty: Presence of Complex Filters

- > A doctor can access a patient's file only
 - During her regular work hours,
 - If the patient is in her ward and
 - If the file is not rated "confidential"

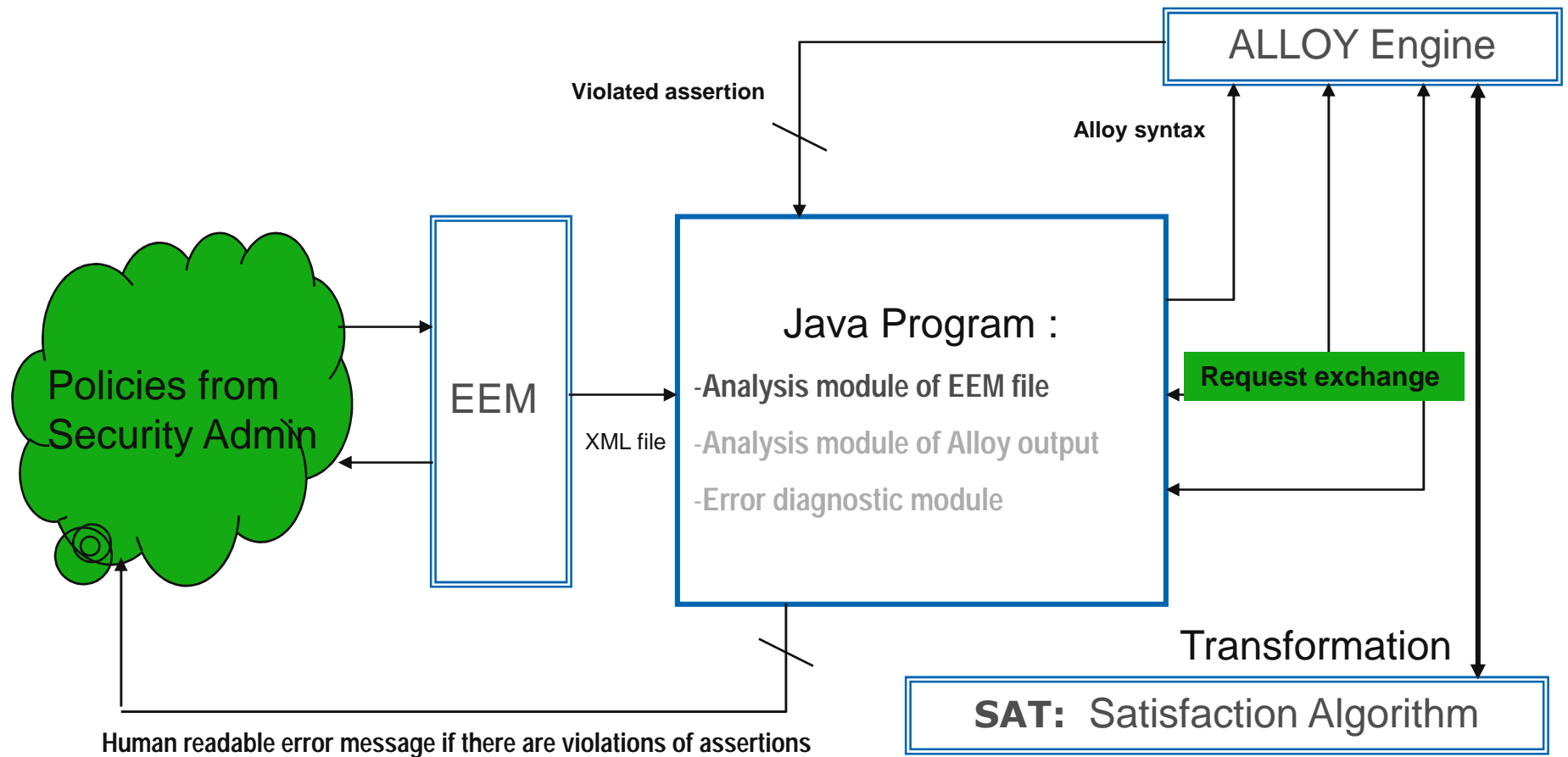
- > Evaluation of these filters can involve evaluation of complex information structures:
 - Enterprise organization
 - Definition of terms
 - Etc.

- > Comparison of many rules involving such filters is computationally hard

How the Logical Engine Works

- > We are using Alloy, a software package developed at MIT
 - Specialized for logical checks of software specifications
- > EEM rules, expressed in XML, are translated into Alloy language
- > Alloy translates them into Boolean expressions:
 $((a \parallel !b) \&\& !(c \parallel !a)) \parallel ((!d \&\& f \&\& c) \parallel (f \parallel a \parallel !c) \dots$
 - It then tries to satisfy them
- > Possible outcomes:
 - The set of policies is inconsistent, logically impossible
 - Can be falsified in certain cases
 - A counterexample is displayed
 - It is always true: it means the policies always allow access

Architecture of our Semantic Policy Assistant (Demo Booth 333)



Direct Implementation

- > Alloy implementation is a prototype, using off-the-shelf packages
- > We are working towards implementations not using any pre-existing package

Automatic Compilation of Enterprise Goals

- > The next step of our project
- > Example of enterprise goal:
 - Employees consulting for two different enterprises cannot share files
- > Enterprise goals can be used to compile or check policies:
 - This example should result in rules of the type:
 - If an employee in Department A requests to read a file owned by an employee in Department B → DENY
 - It should be possible to *compile* such rules starting from enterprise goals
 - Given enough information on enterprise structure

Enterprise Goals Can be Fairly Complex

- > The following rule applies to employees in low security clearance roles:
 - Those who can issue payments cannot approve them, and vice-versa
- > However, high-security clearance roles can both issue and approve payments
- > These rules cannot be violated by using delegation
- > **How can we generate access control rules to satisfy these goals?**

Yes, How to Compile?

> Need:

- A formal model of the enterprise
- A formal language for specifying the enterprise goals
 - Compilation is then possible

> Difficulty for administrator can be reduced by using a formal language that is close to the language he normally uses

Going Beyond: Privacy and Other Laws

Application to Law

- > Laws must be consistent and complete
- > The *implementation* of law must be
 - Consistent with the law
 - Complete, all cases should be considered
- > Methods similar to what we have seen apply, however the logic and the mechanisms are complex
- > Additional problem of translating legalese into logic

A Method and Tool for Checking Compliance

- > Compliance of the implementation of privacy law in an organization
- > Again, this is possible if the following can be expressed in a logic-based language:
 - Structure of the organization
 - Privacy law
 - Enterprise regulations

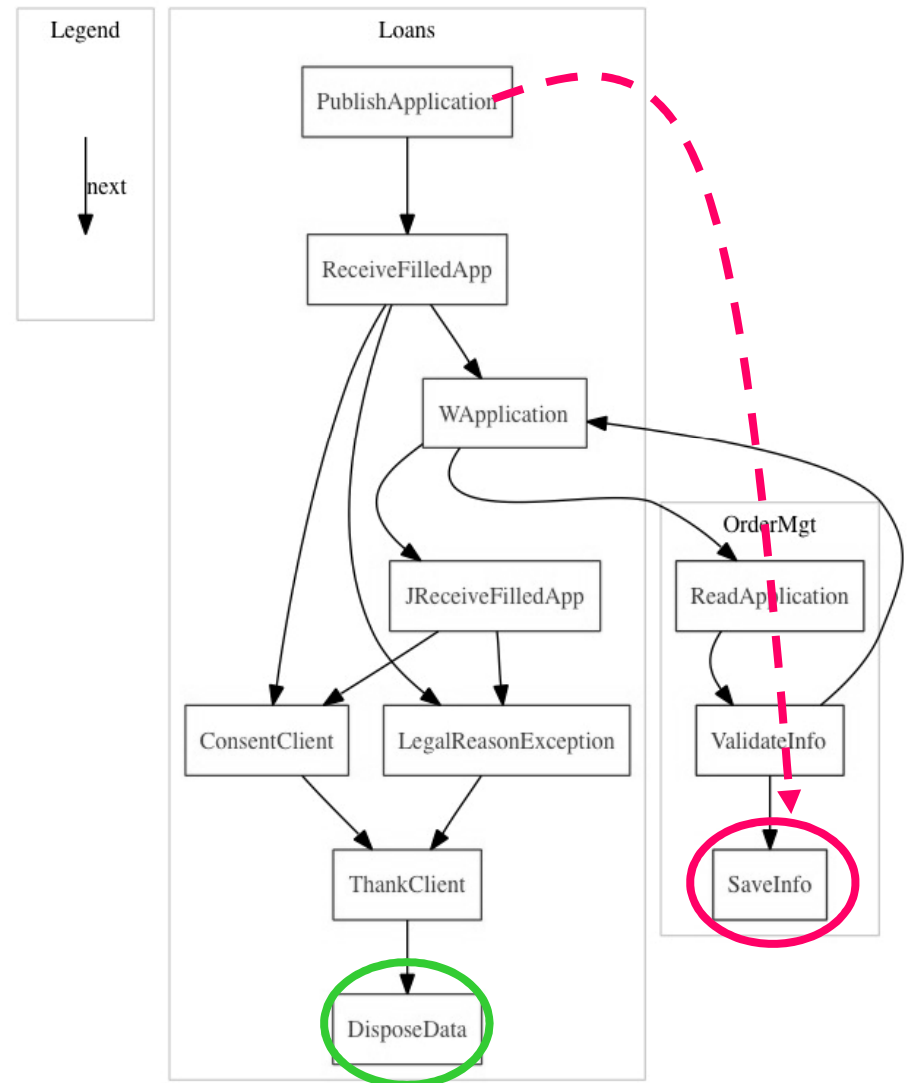
Example 1

- > Privacy law states that in case of delegation of the role of Chief Privacy Officer:
 - A request to the organization for the name of the privacy officer
 - Should yield the name of the delegate,
 - Rather than the name of the delegator

- > Our tool discovers that in a company, this is not implemented
 - The name yielded is the name of the delegator

Example 2: Violation of Privacy Law

- > Law states that private info must be disposed of at the end of a transaction
- > The law and a company's regulation were represented in the language of our tool
- > Our tool discovered that the info can be passed to another branch of the company, which can retain it
 - Violation scenario



Other Case Studies

Under Development

- > Selected articles of Canada's privacy protection law (PIPEDA)
- > Selected sections of Sarbanes-Oxley
 - Art 404

E-Governance

- > This work is a bridge towards automating and validating systems for e-governance

Conclusion of Part 1

- > By using logic tools, access control rules can be checked for consistency, completeness and compliance
- > Leading to correction of potential security breaches
 - **Benefits:**
 - Reduction of risks to CA EEM users
 - Help in maintaining large sets of policies

Conclusion of Part 1 (cont.)

> Longer-term solution is automatic compilation of access control rules from enterprise policies

- **Benefits:**

- High-level understanding and control of access rules by security administrator
- Enterprise structure is taken into consideration automatically
- Tighter sets of rules, error reduction

> **See Demo Booth 333**

Conclusion of Part 2

- > Parts of regulations and laws can be expressed in logic language
- > Logic tools can be applied, allowing:
 - Checking laws for consistency and completeness
 - Formal auditing of enterprise compliance
- > Applications areas:
 - Security and privacy laws and regulations
 - Laws and mechanisms for E-governance
- > **Benefits:**
 - Laws and enterprise regulations will be clearer, better structured
 - Enterprise compliance can be formally checked and audited
- > Much work still to be done, but preliminary results are promising

With Thanks to Colleagues and Co-researchers

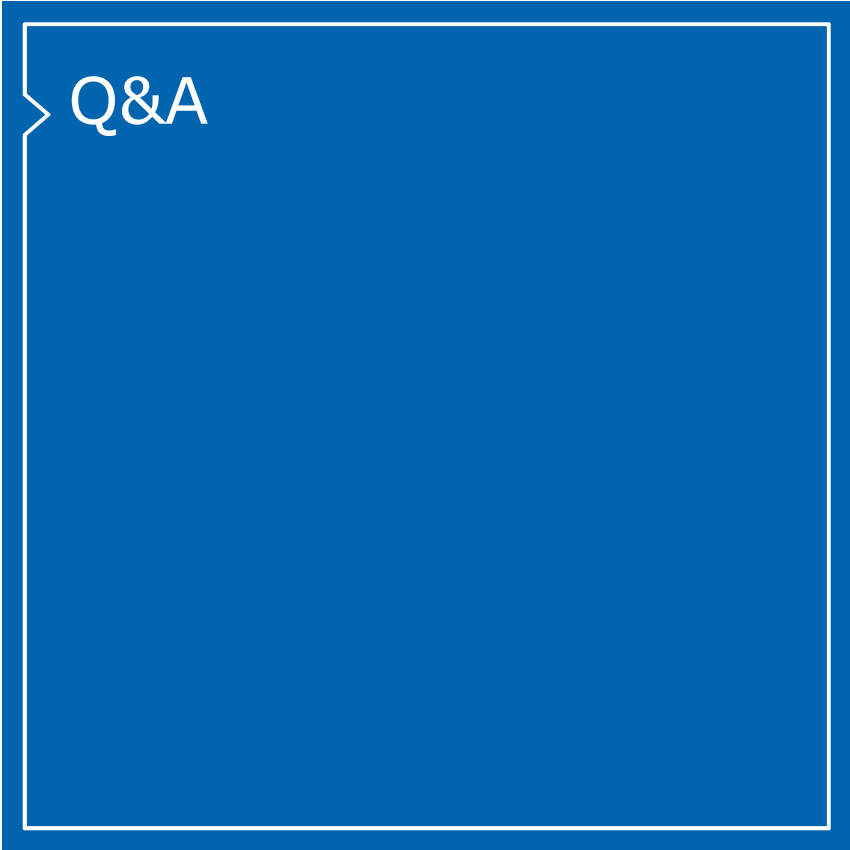
- > Sergei Mankowskii (CA Research Staff Member)
- > Kamel Adi (principal co-investigator)
- > Yacine Bouzida (post-doc)
- > Soufiene Boulares (Master's student)
- > Waël Hassan (PhD student)
- > Ikhlass Hattak (Master's student)
- > S. Yakin Layouni (Master's student)
- > Liwa Layouni (Intern)
- > Mahdi Mankai (MSc)
- > Nadera Slimani (PhD student)

Terms of This Presentation

This presentation was based on current information and resource allocations as of November 16, 2008 and is subject to change or withdrawal by CA at any time without notice. Notwithstanding anything in this presentation to the contrary, this presentation shall not serve to (i) affect the rights and/or obligations of CA or its licensees under any existing or future written license agreement or services agreement relating to any CA software product; or (ii) amend any product documentation or specifications for any CA software product. The development, release and timing of any features or functionality described in this presentation remain at CA's sole discretion. Notwithstanding anything in this presentation to the contrary, upon the general availability of any future CA product release referenced in this presentation, CA will make such release available (i) for sale to new licensees of such product; and (ii) to existing licensees of such product on a when and if-available basis as part of CA maintenance and support, and in the form of a regularly scheduled major product release. Such releases may be made available to current licensees of such product who are current subscribers to CA maintenance and support on a when and if-available basis. In the event of a conflict between the terms of this paragraph and any other information contained in this presentation, the terms of this paragraph shall govern.

For Informational Purposes Only

Certain information in this presentation may outline CA's general product direction. All information in this presentation is for your informational purposes only and may not be incorporated into any contract. CA assumes no responsibility for the accuracy or completeness of the information. To the extent permitted by applicable law, CA provides this document "as is" without warranty of any kind, including without limitation, any implied warranties or merchantability, fitness for a particular purpose, or non-infringement. In no event will CA be liable for any loss or damage, direct or indirect, from the use of this document, including, without limitation, lost profits, lost investment, business interruption, goodwill, or lost data, even if CA is expressly advised of the possibility of such damages.



Exhibition Center

Related CA and Partner Technology

> CA

- Booth 333 — CA Labs

> Partner

- Booth 333 — Université du Québec en Outaouais: "Policy Assistant" for the Formulation of User Policies in Access Control Systems

> Exhibition Center Tours

- Sign up at the Info Desk in the Exhibition Center

