Permission to Speak: An Access Control Logic

Nikhil Dinesh Aravind Joshi Insup Lee Oleg Sokolsky

Department of Computer and Information Science
University of Pennsylvania

FLACOS 2008
November 27-28, 2008

PRECISE
PENN RESEARCH IN EMBEDDED COMPUTING AND INTEGRATED SYSTEMS ENGINEERING

Outline

1. Introduction and motivation
2. System Architecture
3. Inference component
4. Policies and conformance
5. Examples

PRECISE
PENN RESEARCH IN EMBEDDED COMPUTING AND INTEGRATED SYSTEMS ENGINEERING

Background

Goal: analysis of regulated operations

- Bloodbanks (in the US, subject to FDA regulations)
- Medical records (in the US, subject to HIPAA)

Regulatory documents

- Natural language
  - Explicit references to connect sentences
  - Lots of exceptions
- Translate to logic one sentence at a time
  - Provide traceability
  - Reduce complexity

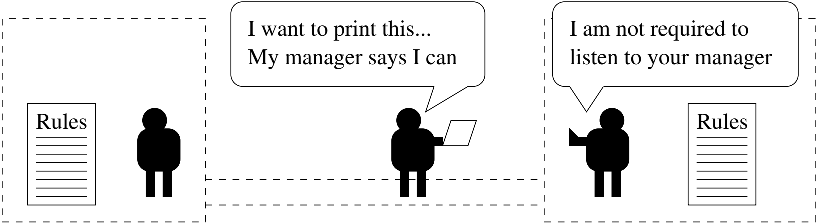
- This talk: access control

PRECISE
PENN RESEARCH IN EMBEDDED COMPUTING AND INTEGRATED SYSTEMS ENGINEERING

The problem

The problem of access control:

- Should a request be granted?



Questions to answer:

- which policies need to be consulted in granting access?
- which policies are violated and who is to blame?

Access control vs. conformance

Policy-based regulation

- A policy specifies what actions are permitted to happen and what are required to happen
- A policy is issued by an authority
  - A large system may have multiple sources of authority
- Possible actions include
  - Performing access
  - Delegating or authorizing access
  - Delegating the right to authorize access

- Access control is a special case of conformance checking

Deontic policies

Need a framework to combine

- Permission and obligation: deontic modalities
- "Saying": policy/credential introduction

Challenges

- Representation and authorization
- Positive and negative permissions
- Nested deontic modalities

Representation in access control

The **saying** modality

A says φ in the laws $I(A)$: $\mathrm{says}_{I(A)}\varphi$

Representation

- B speaks using the authority of A
  - Allows us to handle authorization and delegation
  - B should be able to make only authorized statements
    - Clear interplay with the notion of permission
- Many formalizations in access control literature
  - Hand-off axiom
  - Many pitfalls to avoid
  - No explicit representation of permissions

Representation: our approach

Axiom of representation

If A says that B is allowed to say φ, then if B says φ, A says φ

$$(\mathrm{says}_{I(A)}(\mathcal{P}_B \mathrm{says}_{I(B)}\varphi) \wedge \mathrm{says}_{I(B)}\varphi) \Rightarrow \mathrm{says}_{I(A)}\varphi$$

Advantages

- Decidable logic with complete semantics
- Hand-off and “speaking for” are obtained as a consequence
  - “speaking for” is representation on all formulas

Positive and negative permissions

A hospital H allows a patient A to access her records

$$\varphi = \mathrm{says}_{I(H)} \mathcal{P}_A(\mathrm{access}(A, A))$$

Suppose the patient listens to music. Is that permitted?

Permission as provability

- Positive permission:
  - Is $\varphi \Rightarrow \mathrm{says}_{I(H)}(\neg \mathcal{O}_A \neg \mathrm{music})$ provable?
- Negative permission:
  - Is $\varphi \Rightarrow \mathrm{says}_{I(H)} \mathcal{O}_A \neg \mathrm{music}$ not provable?

Nested deontic modalities

Parents (*A*) should not let their children (*B*) play by the road

Possible interpretations:

- Positive permission: *A* should not give permission to play
  - Too weak?
- Negative permission: *A* should tell *B* not to play
  - Arguably, adequate
- *A* should physically prevent *B* from playing
  - Too restrictive?

In the regulated setting

If *B* plays by the road, who is to blame: *A* or *B*?

Nested deontic modalities: our approach

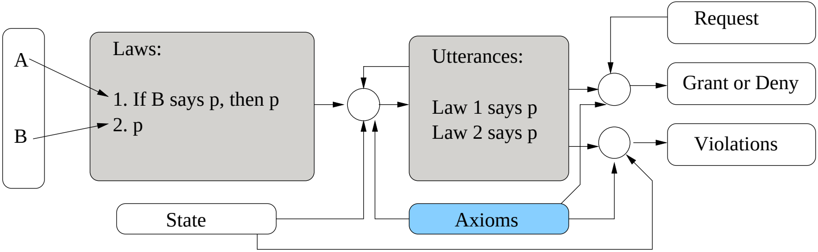
Saying is crucial for the analysis

A hospital (H) permits patients (A) to permit their family (B) to access their information

- H says that A is *permitted to say* that B is permitted to access
  - $\mathrm{says}_{I(H)}\mathcal{P}_A\mathrm{says}_{I(A)}\mathcal{P}_B\mathrm{access}(A, B)$
- Now, when A gives permission
  - $\mathrm{says}_{I(A)}\mathcal{P}_B\mathrm{access}(A, B)$
- We should be able to infer that H permits access to B
  - $\mathrm{says}_{I(H)}\mathcal{P}_B\mathrm{access}(A, B)$
- In other words, A represents H on $\mathrm{access}(A, B)$.

PRECISE
PENN RESEARCH IN EMBEDDED COMPUTING AND INTEGRATED SYSTEMS ENGINEERING

System architecture



Utterances and conformance

- Evaluation of policies yields a set of utterances
- Access control: is a request permitted by utterances?
- Conformance: do actions satisfy obligations in utterances?

Logic of saying and obligation

Syntax of L

$$\varphi \quad ::= \quad \alpha \mid \varphi \wedge \varphi \mid \neg\varphi \mid \text{says}_{Id}\psi \mid \text{says}_{I(y)}\psi$$
$$\psi \quad ::= \quad \varphi \mid \psi \wedge \psi \mid \neg\psi \mid \mathcal{O}_y\varphi$$

- Atomic predicates: $\alpha = p(y_1, \ldots, y_j)$
  - Predicates are applied to objects or variables: $y_i \in X \cup O$
  - E.g. $\text{access}(A, B)$ - access of A's medical records by B
- Saying is parameterized on a set of laws
- Syntax enforces alternation between saying and obligation

PRECISE
PENN RESEARCH IN EMBEDDED COMPUTING AND INTEGRATED SYSTEMS ENGINEERING

Axiomatization

A1 All substitution instances of propositional tautologies.

A2 $\mathcal{Q}(\varphi \Rightarrow \psi) \Rightarrow (\mathcal{Q}(\varphi) \Rightarrow \mathcal{Q}(\psi))$ (for all modalities $\mathcal{Q}$)

A3 $\mathrm{says}_{Id}\varphi \Rightarrow \mathrm{says}_{Id'}\varphi$ (for all $Id \subseteq Id'$)

A4 $\mathcal{O}_A\varphi \Rightarrow \mathcal{P}_A\varphi$ (for all $A \in O$)

A5 $\mathrm{says}_{Id_A}(\mathcal{P}_B \mathrm{says}_{Id_B}\varphi) \Rightarrow (\mathrm{says}_{Id_B}\varphi \Rightarrow \mathrm{says}_{Id_A}\varphi)$ (for all $\{A, B\} \subseteq O$, $Id_A \subseteq I(A)$, and $Id_B \subseteq I(B)$)

A6 $\mathrm{says}_{Id_A}(\mathcal{P}_B \mathrm{says}_{Id_A}\varphi) \Rightarrow \mathrm{says}_{Id_A}\varphi$ (for all $\{A, B\} \subseteq O$, and $Id_A \subseteq I(A)$)

R1 From $\vdash \varphi \Rightarrow \psi$ and $\vdash \varphi$, infer $\vdash \psi$

R2 From $\vdash \varphi$, infer $\vdash \mathcal{Q}(\varphi)$ (for all modalities $\mathcal{Q}$)

PRECISE

PENN RESEARCH IN EMBEDDED COMPUTING AND INTEGRATED SYSTEMS ENGINEERING

Decidability

Provability is decidable for the propositional case

For all $\varphi \in L$, $\vdash \varphi$ is decidable

Complexity

- Satisfiability checking is NEXPTIME-complete
- A variant of axioms **A5**, **A6** allows PSPACE satisfiability
  - A strictly larger set of formulas is provable
  - Open question: is it adequate in access control applications?

PRECISE
PENN RESEARCH IN EMBEDDED COMPUTING AND INTEGRATED SYSTEMS ENGINEERING

Policies

Logic programming framework

- A policy is a collection of statements

$$(id)\ \varphi \mapsto \psi$$

- Each statement has a unique *id*
- Preconditions $\varphi \in L_\varphi$
  - Obligations must be in the scope of saying
- True preconditions must have true postconditions
  - Postconditions may make more preconditions true

PRECISE
PENN RESEARCH IN EMBEDDED COMPUTING AND INTEGRATED SYSTEMS ENGINEERING

States and assignments

State

- Objects known to the system
- Interpretation of predicates w.r.t. objects
- Example:
  - Objects: A, B, C, d
  - Predicates: $\text{patient}(A)$, $\text{patient}(B)$, $\text{relative}(A, C)$, $\text{access}(B, C)$, $\text{test}(B, d)$

Evaluation of ground formulas

- Policies are evaluated in a given state
- Assignments map variables in the formula to objects

Utterances

- The first step in checking conformance is to determine what has been said.

Utterance is a nugget of saying

$$v(\mathrm{says}_{\{id\}}\psi, S)$$

- Policy contains $(id)\ \varphi \mapsto \psi$
- S is a state, v is an assignment

Utterance pairs (U, U')

- Utterance set U corresponds to true preconditions
- Utterance set U' corresponds to non-false preconditions

Computing utterances (I)

Evaluation of preconditions

- Evaluation is up to an utterance pair: $\mathbf{tv}_{(U,U')}(\varphi, S, v)$
- Interesting case: the saying modality

$$\mathbf{tv}_{(U,U')}(\mathrm{says}_{Id}\psi, S, v) = \begin{cases} \top \text{ if } U \vdash v(\mathrm{says}_{Id}\psi, S) \\ \bot \text{ if } U' \not\vdash v(\mathrm{says}_{Id}\psi, S) \\ ? \text{ otherwise} \end{cases}$$

Consistent utterance pair $U \subseteq U'$

For all policy statements $(id)\ \varphi \mapsto \psi$

- If $v(\mathrm{says}_{\{id\}}\psi, S) \in U$, $\mathbf{tv}_{(U,U')}(\varphi, S, v) = \top$
- If $v(\mathrm{says}_{\{id\}}\psi, S) \notin U'$, $\mathbf{tv}_{(U,U')}(\varphi, S, v) = \bot$

Computing utterances (II)

Fixed point computation

- Initialization: $U = \emptyset$, $U' =$ utterances for all postconditions
- Computation step:
  - Compute $\mathbf{tv}(U, U')$ for all preconditions
  - Add utterances whose preconditions evaluate to $\top$ to U
  - Remove utterances whose preconditions evaluate to $\perp$ from U'
- Stop when fixed point is reached

Correctness

- The partially ordered set of consistent utterances has a least fixed point
- Computation is monotonic

Conformance

Conformance is satisfaction of obligations

- A conforms to the laws Id:

$$\text{If } S \models_{(U,U')} \mathrm{says}_{Id} \mathcal{O}_A \varphi, \text{ then } S \models_{(U,U')} \varphi$$

Access control is permission by the laws of the owner

- A can perform an action p controlled by B

$$S \models_{(U,U')} \mathrm{says}_{I(B)} \mathcal{P}_A p$$

Conformance with nested deontic modalities

Example

- Owners of parking lots must forbid parking by lot entrance
- Our interpretation:
  - Owners of parking lots must introduce rules that forbid parking near lot entrance
  - (P) $\mathrm{owner}(x) \wedge \mathrm{driver}(y) \mapsto \mathcal{O}_x \mathrm{says}_{l(x)} \mathcal{O}_y \neg \mathrm{pk}(y, x)$

Conformance

- If an owner A does not introduce any rules and $\mathrm{pk}(B, A)$
  - B conforms to (P) but A does not conform to (P)
- If A introduces $\mathrm{driver}(y) \mapsto \mathcal{O}_y \neg \mathrm{pk}(y, A)$
  - A conforms to (P) but B does not conform to (P)

A more elaborate example

Health Insurance Portability and Accountability Act (HIPAA)

- Regulates the uses and disclosures of health information
- Hospitals have local policies, must be HIPAA compliant
- Users give written consent, also part of the regulation

1. An individual **has a right** to access her PHI, except for:
   i. Psychotherapy notes;
   ii. PHI compiled for a legal proceeding; or
   ...

What is a right?

Formalization

Our interpretation

1 An individual **is permitted to require the hospital to permit** to access her PHI, except for:

- i Psychotherapy notes;
- ii PHI compiled for a legal proceeding; or
- ...

- Let $\varphi(x, y, z) = \mathrm{ind}(x) \wedge \mathrm{says}_{I(HIPAA)}\mathrm{ce}(y) \wedge \mathrm{info}(z, x, y)$

(1) $\varphi(x, y, z) \wedge \neg\mathrm{says}_{\{i,ii\}}\mathrm{list}(z) \mapsto \mathcal{P}_x\mathrm{says}_{I(x)}\mathcal{O}_y\mathrm{says}_{I(y)}\mathcal{P}_x\mathrm{access}(x, z)$

Hospital and user policies

Conformant policies

- A permissive hospital: $\top \mapsto \mathcal{P}_A \mathrm{access}(A, r)$
- A hospital who only wants to give access when HIPAA requires it:
  - $\top \mapsto \mathcal{P}_{HIPAA} \mathrm{says}_{I(HIPAA)} \mathcal{O}_H \mathrm{says}_{I(H)} \mathcal{P}_A \mathrm{access}(A, r)$
  - *H* permits *HIPAA* to require it to permit *A* to access.

HIPAA consent forms

- $\top \mapsto \mathcal{O}_H \mathrm{says}_{I(H)} \mathcal{P}_A \mathrm{access}(A, r)$
- Registrars care only about obligations imposed by the hospital

Happy end: $\mathrm{says}_{I(H)} \mathcal{P}_A \mathrm{access}(A, r)$ is derived

PRECISE
PENN RESEARCH IN EMBEDDED COMPUTING AND INTEGRATED SYSTEMS ENGINEERING

Conclusions

- Logic to represent regulatory documents
  - permission, obligation, cross-referencing
  - multiple sources of authority
- Aimed at checking conformance
  - conformance is decidable and reasonably efficient in practice
- Cross-references can be compiled away for acyclic regulation
  - lose traceability (counterexample generation)
- Designed with NLP in mind
  - Parser is work in progress

PRECISE
PENN RESEARCH IN EMBEDDED COMPUTING AND INTEGRATED SYSTEMS ENGINEERING