Principled Audit Mechanisms for Privacy Protection

Anupam Datta
Carnegie Mellon University

CyLab Briefing
November 11, 2011
Healthcare Privacy

Desiderata: Respect privacy expectations in the flow and use of personal information within and across organizational boundaries
Personal Information is Everywhere

Desiderata: Generality + application to specific domains of importance in society (e.g., healthcare – HIPAA Privacy Rule)
Example from HIPAA Privacy Rule

A covered entity may disclose an individual’s protected health information (phi) to law-enforcement officials for the purpose of identifying an individual if the individual made a statement admitting participating in a violent crime that the covered entity believes may have caused serious physical harm to the victim.

- **Concepts in privacy policies**
  - **Actions:** send(p1, p2, m)
  - **Roles:** inrole(p2, law-enforcement)
  - **Data attributes:** attr_in(prescription, phi)
  - **Temporal constraints:** in-the-past(state(q, m))
  - **Purposes:** purp_in(u, id-criminal))
  - **Beliefs:** believes-crime-caused-serious-harm(p, q, m)

**Preventive enforcement (access control or runtime monitoring) does not suffice**
A Research Agenda

Design principled audit mechanisms for enforcing privacy policies

Simple audit tools already available commercially
(FairWarning, Cerner’s P2P Sentinel, …)
The Big Picture

Privacy Policy

Organizational audit log

Complete formalization of HIPAA, GLBA

Computer-readable privacy policy

Audit

Automated audit for black-and-white policy concepts

Detect policy violations

Oracles to audit for grey policy concepts
Key Challenge for Auditing

Audit Logs are Incomplete

Future: store only past and current events
Example: Timely data breach notification refers to future event

Subjective: no “grey” information
Example: May not record evidence for purposes and beliefs

Spatial: remote logs may be inaccessible
Example: Logs distributed across different departments of a hospital
Abstract Model of Incomplete Logs

Model all incomplete logs uniformly as 3-valued structures

\[ \mathcal{L}(P) \in \{tt, ff, uu\} \]

Define **semantics** (meanings of formulas) over 3-valued structures
reduce: The Iterative Algorithm

\[ \text{reduce } (\mathcal{L}, \varphi) = \varphi' \]
Example from HIPAA Privacy Rule

A covered entity may disclose an individual’s protected health information (phi) to law-enforcement officials for the purpose of identifying an individual if the individual made a statement admitting participating in a violent crime that the covered entity believes may have caused serious physical harm to the victim.

\[\forall p_1, p_2, m, u, q, t.\]

\[(\text{send}(p_1, p_2, m) \land \text{inrole}(p_2, \text{law-enforcement}) \land \text{tagged}(m, q, t, u) \land \text{attr_in}(t, \phi)) \implies (\text{purp_in}(u, \text{id-criminal}))\]

\[\exists m'. \Diamond \text{state}(q, m') \land \text{is-admission-of-crime}(m') \land \text{believes-crime-caused-serious-harm}(p_1, q, m')\]
Example

\[ \varphi = \forall p_1, p_2, m, u, q, t. \]
\[ (\text{send}(p_1, p_2, m) \land \text{tagged}(m, q, t, u) \land \text{attr}_\text{in}(t, \phi)) \]
\[ \supset \text{inrole}(p_2, \text{law-enforcement}) \land \text{purp}_\text{in}(u, \text{id-criminal}) \land \exists m'. (\text{\& state}(q, m') \land \text{is-admission-of-crime}(m') \land \text{believes-crime-caused-serious-harm}(p_1, m')) \]

Finite Substitutions
\{ p_1 \rightarrow \text{UPMC},
    p_2 \rightarrow \text{allegeny-police},
    m \rightarrow \text{M2},
    q \rightarrow \text{Bob},
    u \rightarrow \text{id-bank-robber},
    t \rightarrow \text{date-of-treatment}
\}
\[ \text{m'} \rightarrow \text{M1} \]
Implementation and Case Study

- Implementation and evaluation over simulated audit logs for compliance with *all* 84 disclosure-related clauses of HIPAA Privacy Rule

- Performance:
  - Average time for checking compliance of each disclosure of protected health information is 0.12s for a 15MB log

- Mechanical enforcement:
  - reduce can automatically check 80% of all the atomic predicates
Other Applications of reduce

- Disclosure Accounting
  - What disclosures have been made of Alice’s information?

- Online Advisory Tool
  - Does HIPAA permit this disclosure? Under what conditions?
The Big Picture

Privacy Policy

Organizational audit log

Complete formalization of HIPAA, GLBA

Computer-readable privacy policy

Audit

Automated audit for black-and-white policy concepts

Detect policy violations

Oracles to audit for grey policy concepts
Auditing Grey Concepts

- **Challenge:** Auditing is imperfect
  - Human auditor can only check a subset of grey concepts due to budgetary constraints

- **Question:** How should auditor allocate the audit budget?

\[ \varphi' = \text{purpose}(u, \text{treatment}) \]
Learning to Audit

Auditing budget: $3000/ cycle
Cost for one inspection: $100
Only 30 inspections per cycle

Access divided into 2 types
- 30 accesses
- 70 accesses

Loss from 1 violation (internal, external)
- $500, $1000
- $250, $500

Auditor access: 100 accesses
Audit Mechanism Choices

Consider 4 possible allocations of the available 30 inspections

Weights

Choose allocation probabilistically based on weights
### Audit Mechanism Run

<table>
<thead>
<tr>
<th>No. of Access</th>
<th>Actual Violation</th>
</tr>
</thead>
<tbody>
<tr>
<td>30</td>
<td>2</td>
</tr>
<tr>
<td>70</td>
<td>4</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Int. Caught</th>
<th>Ext. Caught</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
</tr>
</tbody>
</table>

**Observed Loss**: $2000, $1500, $1000, $1000

**Estimated Loss**: $750, $1250, $1250, $1500

**Updated weights**: 0.5, 0.5, 2.0, 1.5

Learning from experience: weights updated using observed and estimated loss
Regret Minimizing Audits

- Learns from experience to recommend budget allocation for audit in each audit cycle
- Budget allocation is *provably close to optimal fixed strategy* in hindsight (e.g., budget allocation)
Audit Mechanisms for Privacy Protection

Privacy Policy

Computer-readable privacy policy

Organizational audit log

Complete formalization of HIPAA, GLBA

Audit

Automated audit for black-and-white policy concepts

Detect policy violations

Learning to audit for grey policy concepts
Thanks! Questions?
Additional Slides
Syntax of Policy Logic

- First-order logic with restricted quantification over infinite domains (challenge for reduce)
- Can express timed temporal properties, “grey” predicates

\[
\begin{align*}
\text{Atoms} & \quad P ::= p(t_1, \ldots, t_n) \\
\text{Formulas} & \quad \varphi ::= P \mid T \mid \bot \mid \varphi_1 \land \varphi_2 \mid \varphi_1 \lor \varphi_2 \mid \forall \bar{x}.(c \supset \varphi) \mid \exists \bar{x}.(c \land \varphi) \\
\text{Restrictions} & \quad c ::= P \mid T \mid \bot \mid c_1 \land c_2 \mid c_1 \lor c_2 \mid \exists \bar{x}.c
\end{align*}
\]
reduce: Formal Definition

General Theorem: If initial policy passes a syntactic **mode check**, then finite substitutions can be computed.

Applications: The entire HIPAA and GLBA Privacy Rules pass this check.
Mode Analysis: Formally

- Given ground variables $\chi_I$, compute variables $\chi_O$ whose instances satisfying $c$ can be computed

$$\chi_I \vdash c : \chi_O$$

- Defined by rules

$$\forall k \in \mathit{I}(p_0). \text{fv}(t_k) \subseteq \chi_I \quad \chi_O = \chi_I \cup \bigcup_{j \in \mathit{O}(p_0)} \text{fv}(t_j)$$

$$\chi_I \vdash p_0(t_1, \ldots, t_n) : \chi_O$$

$$\chi_I \vdash c_1 : \chi \quad \chi \vdash c_2 : \chi_O$$

$$\chi_I \vdash c_1 \land c_2 : \chi_O$$
Mode Analysis: Theorem

**Theorem 4.7** (Totality of $\widehat{\text{sat}}$). If $\chi_I \vdash c : \chi_O$, then for all structures $\mathcal{L}$ and all substitutions $\sigma$ with $\text{dom}(\sigma) \supseteq \chi_I$, $\widehat{\text{sat}}(\mathcal{L}, c\sigma)$ is defined and, further, for each substitution $\sigma' \in \widehat{\text{sat}}(\mathcal{L}, c\sigma)$, $\chi_I \cup \text{dom}(\sigma') \supseteq \chi_O$. 

Computing Finite Substitutions

Assume: The function sat(L, P) computes all substitutions σ for variables in P such that L |= Pσ, if certain argument positions in P are ground.

\[
\begin{array}{ll}
\hat{\text{sat}}(L, p_O(t_1, \ldots, t_n)) & = \text{sat}(L, p_O(t_1, \ldots, t_n)) \\
\hat{\text{sat}}(L, \top) & = \{\bullet\} \\
\hat{\text{sat}}(L, \bot) & = \emptyset \\
\hat{\text{sat}}(L, c_1 \land c_2) & = \bigcup_{\sigma \in \hat{\text{sat}}(L, c_1)} \sigma + \hat{\text{sat}}(L, c_2\sigma) \\
\hat{\text{sat}}(L, c_1 \lor c_2) & = \hat{\text{sat}}(L, c_1) \cup \hat{\text{sat}}(L, c_2) \\
\hat{\text{sat}}(L, \exists x. c) & = \hat{\text{sat}}(L, c) \setminus \{x\} \quad (x \text{ fresh})
\end{array}
\]
Experimental Evaluation

<table>
<thead>
<tr>
<th>Exp No.</th>
<th>Ave. Time per disclosure (s)</th>
<th>Total time (s)</th>
<th>Memory used (KB)</th>
<th>Prob. of violation</th>
<th>Log size (MB)</th>
<th>Number of entries/indexed (Y or N)</th>
<th>send</th>
<th>tagged</th>
<th>attr_in_db</th>
<th>the rest</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.27</td>
<td>1453.12</td>
<td>592440</td>
<td>0.10</td>
<td>2.68</td>
<td>5401/N, 4947/N, 5100/N</td>
<td>-/N</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>0.27</td>
<td>1456.43</td>
<td>592536</td>
<td>0.10</td>
<td>2.99</td>
<td>5401/Y, 4947/N, 5100/N</td>
<td>-/N</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>0.27</td>
<td>1448.53</td>
<td>592604</td>
<td>0.10</td>
<td>3.14</td>
<td>5401/Y, 4947/Y, 5100/N</td>
<td>-/N</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>0.26</td>
<td>1378.62</td>
<td>657436</td>
<td>0.10</td>
<td>3.29</td>
<td>5401/Y, 4947/Y, 5100/Y</td>
<td>-/N</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>0.04</td>
<td>209.21</td>
<td>662080</td>
<td>0.10</td>
<td>3.98</td>
<td>5401/Y, 4947/Y, 5100/Y</td>
<td>-/N</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>0.07</td>
<td>725.19</td>
<td>1336808</td>
<td>0.10</td>
<td>7.68</td>
<td>10866/Y, 9040/Y, 9314/Y</td>
<td>-/Y</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>0.12</td>
<td>2674.77</td>
<td>2375564</td>
<td>0</td>
<td>15.11</td>
<td>21684/Y, 16945/Y, 17565/Y</td>
<td>-/Y</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>0.12</td>
<td>2680.25</td>
<td>2667132</td>
<td>0.10</td>
<td>15.10</td>
<td>21742/Y, 17182/Y, 17732/Y</td>
<td>-/Y</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>0.11</td>
<td>2306.93</td>
<td>3565964</td>
<td>1</td>
<td>14.41</td>
<td>21626/Y, 16865/Y, 17465/Y</td>
<td>-/Y</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 2: Experimental evaluation of single runs of the algorithm reduce
Experimental Evaluation

<table>
<thead>
<tr>
<th>Policy size (KB)</th>
<th>Residual policy size (KB)</th>
<th>Ave. Time per disclosure (s)</th>
<th>Total time (s)</th>
<th>Memory used (KB)</th>
<th>Log size (MB)</th>
<th>Number of disclosures</th>
</tr>
</thead>
<tbody>
<tr>
<td>39.52</td>
<td>4654.07</td>
<td>0.04</td>
<td>209.68</td>
<td>662080</td>
<td>3.98</td>
<td>5401</td>
</tr>
<tr>
<td>4654.07</td>
<td>9369.32</td>
<td>0.75</td>
<td>8193.76</td>
<td>4063472</td>
<td>7.68</td>
<td>10866</td>
</tr>
<tr>
<td>9369.32</td>
<td>19186.84</td>
<td>2.99</td>
<td>65009.30</td>
<td>4033484</td>
<td>15.1</td>
<td>21742</td>
</tr>
</tbody>
</table>

**Figure 3:** Experimental evaluation of iterative runs of the algorithm reduce
Other Applications of Reduce

**Runtime monitoring**

For policies that do not mention unbounded future obligations or grey concepts

(Special case usually addressed in the runtime verification literature)
Related Work

**Specification Languages & Logics for Privacy Policies**

P3P[Crànor et al.], XACML[OASIS], EPAL[Backes et al.], *Logic of Privacy and Utility* [Barth et al.], PrivacyAPIs [Gunter et al.], …
Related Work

**Logical Specification of Privacy Laws**

*Logic of Privacy and Utility* [Barth et al.]: Example clauses from HIPAA and GLBA
Privacy APIs [Gunter et al.]: HIPAA 164.506
Datalog HIPAA [Lam et al.]: HIPAA 164.502, 164.506, 164.510
Related Work

Runtime monitoring in MFOTL

[Basin et al ’10]

Pre-emptive enforcement
Efficient implementation
Assumes past-completeness of logs
Less expressive mode checking
(“safe-range check”)
Cannot express HIPAA or GLBA
Related Work

Iterative Model Checking

[Thati, Rosu ’05]

Propositional logic
Cannot express privacy legislation
Formal Properties of Reduce

Correctness

**Theorem 3.2** (Partial correctness of reduce). If \( \text{reduce}(\mathcal{L}, \varphi) = \psi \) and \( \mathcal{L} \leq \mathcal{L}' \), then (1) \( \mathcal{L}' \models \varphi \) iff \( \mathcal{L}' \models \psi \) and (2) \( \mathcal{L}' \models \overline{\varphi} \) iff \( \mathcal{L}' \models \psi \).

**Theorem 3.3** (Totality of reduce). If \( \vdash \varphi \) then there is a \( \psi \) such that \( \text{reduce}(\mathcal{L}, \varphi) = \psi \) and \( \vdash \psi \).
Formal Properties of Reduce

Minimality of Output

**Theorem 3.5 (Minimality).** If $\vdash \varphi$ and $\text{reduce}(\mathcal{L}, \varphi) = \psi$, then $\text{atoms}(\mathcal{L}, \psi) \subseteq \text{atoms}(\mathcal{L}, \varphi) \cap \{P \mid \mathcal{L}(P) = \text{uu}\}$. 
Formal Properties of Reduce

Complexity

Theorem 3.4 (Complexity of reduce). Assuming that computing each output returned by sat takes unit time, the algorithm \( \text{reduce}(\mathcal{L}, \varphi) \) lies in the intersection of the complexity classes \( \text{TIME}(|\mathcal{L}|^{O(|\varphi|)}) \) and \( \text{PSPACE}(|\varphi|) \).
Summary of Results

1. Privacy laws represented in logic
   - Informed by theory of contextual integrity [Nissenbaum]
   - First complete formalization of HIPAA and GLBA [WPES 2010]

2. Automatic audit of incomplete logs [CCS 2011]
   - Applies to significant part of HIPAA, GLBA
   - Outputs residual policy involving grey predicates
   - Efficient, practical implementation

3. Learning algorithm guides human audit of grey concepts in a manner that minimizes risk [CSF 2011]