



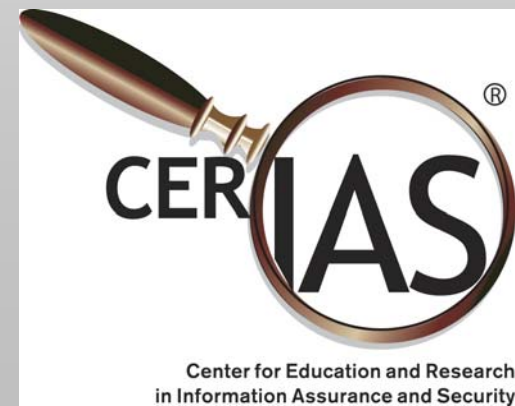
Privacy-Preserving Database Systems

Elisa Bertino

bertino@cerias.purdue.edu

4/27/2009

Department of Computer Science



Outline

- Motivations
- Research Directions in Privacy-Preserving Database Systems
- P3P – Overview and Critique
- Purpose-Based Access Control for Relational Databases and Extended RBAC
- Purpose-Based Access Control for Complex Objects
- Generalized Fine-Grained Access Control Models for Relational Databases
- Conclusions

Motivations

- Privacy is an important issue today
 - Individuals feel
 - Uncomfortable: ownership of information
 - Unsafe: information can be misused
 - (e.g., identity thefts)
 - Enterprises need to
 - Keep their customers feel safe
 - Maintain good reputations
 - Protect themselves from any legal dispute
 - Obey legal regulations

Definition



- **Privacy** is the ability of a person to control the availability of information about and exposure of him- or herself. It is related to being able to function in society anonymously (including pseudonymous or blind credential identification).
- **Types of privacy** giving raise to special concerns:
 - Political privacy
 - Consumer privacy
 - Medical privacy
 - *Information technology end-user privacy; also called data privacy*
 - Private property

Data Privacy



- Data Privacy problems exist *wherever uniquely identifiable data relating to a person or persons are collected and stored, in digital form or otherwise.* Improper or non-existent disclosure control can be the root cause for privacy issues.
- The most common sources of data that are affected by data privacy issues are:
 - Health information
 - Criminal justice
 - Financial information
 - Genetic information

Data Privacy

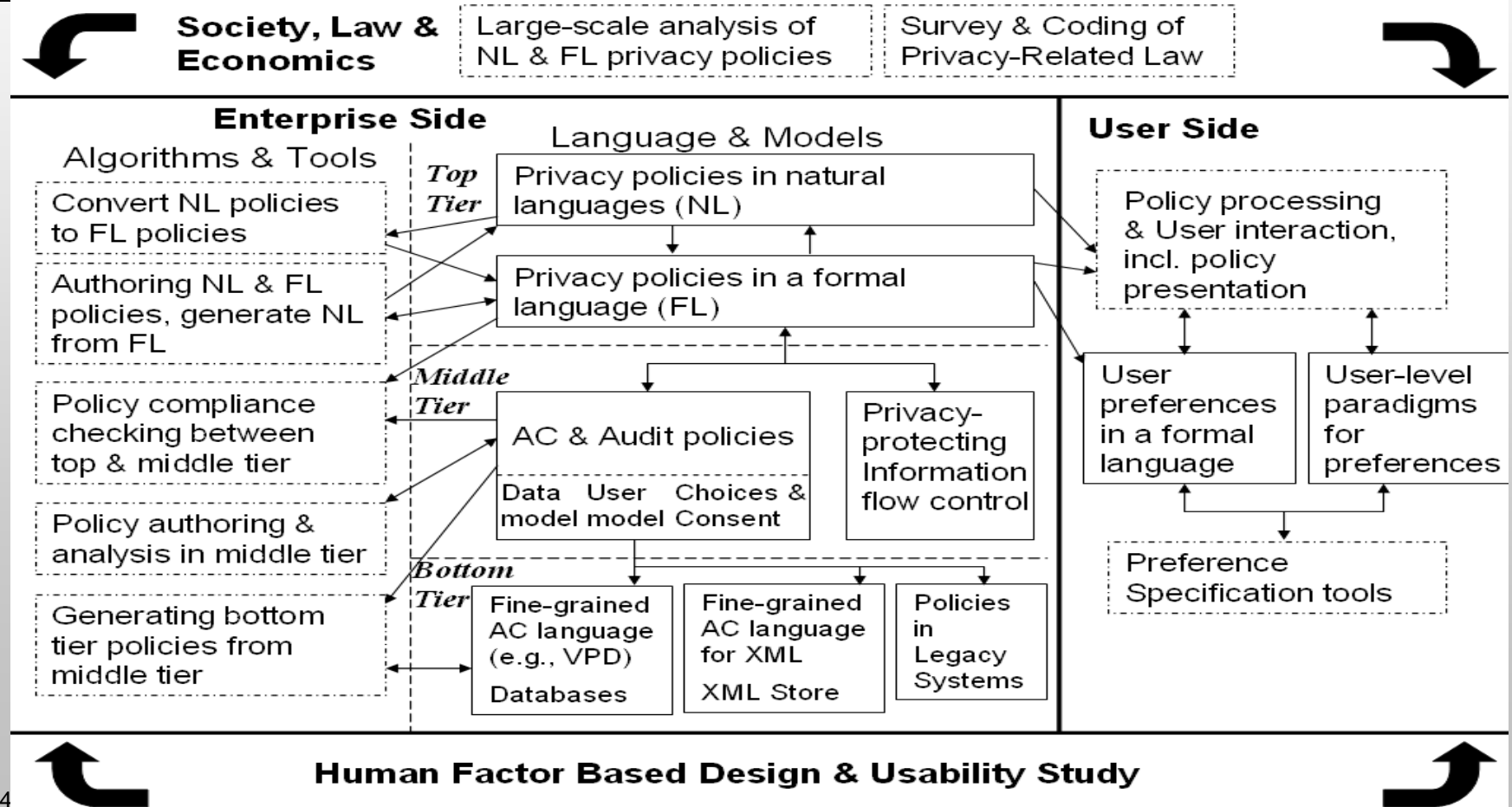
- The challenge in data privacy is to share data while protecting the personally identifiable information.
 - Consider the example of health data which are collected from hospitals in a district; it is standard practice to share this only in aggregate form
 - The idea of sharing the data in aggregate form is to ensure that only non-identifiable data are shared.

- The legal protection of the right to privacy in general and of data privacy in particular varies greatly around the world.

Technologies with Privacy Concerns

- ❑ Biometrics (fingerprints, iris) and face recognition
- ❑ Video surveillance, ubiquitous networks and sensors
- ❑ Cellular phones
- ❑ Personal Robots
- ❑ DNA sequences, Genomic Data

on-line Privacy Protection A Comprehensive Framework



Research Directions in Privacy-Preserving Database Systems

Research Directions in Privacy-Preserving Database Systems



- Anonymization Techniques
- Privacy-Preserving Data Mining
- DBMS with support for P3P and Hippocratic Databases
- Fine-Grained Access Control Techniques

Anonymization Techniques



Motivations - Latanya Sweeney's Finding

- In Massachusetts, the Group Insurance Commission (GIC) is responsible for purchasing health insurance for state employees
- GIC has to publish the data:
GIC(zip, dob, sex, diagnosis, procedure, ...)

Latanya Sweeney's Finding

- Sweeney paid \$20 and bought the voter registration list for Cambridge Massachusetts:

GIC(zip, dob, sex, diagnosis, procedure, ...)
VOTER(name, party, ..., zip, dob, sex)

Anonymization Techniques



Motivations - Latanya Sweeney's Finding

zip, dob, sex

- William Weld (former governor) lives in Cambridge, hence is in VOTER
- 6 people in VOTER share his **dob**
- only 3 of them were man (same **sex**)
- Weld was the only one in that **zip**
- Sweeney learned Weld's medical records !

Anonymization Techniques

Idea of k -anonymity



- Developed by Latanya Sweeney, the goal is to prevent linking a record from a set of released records to a specific individual
- Under k -anonymity, there will be at least k individuals to whom a given record indistinctly refers
- The k individuals appear in the released records

Anonymization Techniques

Example of k -anonymity

Given a table T of data,
“suppress” or “generalize”
entries of T so that for every
row, $k-1$ other rows look
identical

example →

first	last	age	race
Harry	Stone	34	Afr-Am
John	Reyser	36	Cauc
Beatrice	Stone	47	Afr-Am
John	Ramos	22	Hisp



first	last	age	race
*	Stone	30-50	Afr-Am
John	R*	20-40	*
*	Stone	30-50	Afr-Am
John	R*	20-40	*

Anonymization Techniques

Open issues



- Efficiency – *given an arbitrary table, what's the minimum number of entries that must be “suppressed” in order to achieve k-anonymity?*
 - *NP-hard*
- Efficient maintenance of anonymized views of data
- Use of anonymization techniques and other techniques (randomization, result sampling) when computing query replies

Privacy Preserving Data Mining



The problem

- The goal of data mining is to extract knowledge from data
- Most data mining applications operate under the assumption that all data is available at a single central repository, called a *data warehouse*
- This poses a huge privacy problem because violating only a single repository's security exposes all data

Privacy Preserving Data Mining

Approaches



- Data swapping and randomization
 - Because the data do not any longer reflects real world values, it can't be used to violate individual privacy

- Extension of data mining techniques to preserve privacy
 - Extensions have been developed for association rule mining techniques and for classification trees techniques

- Distributed privacy-preserving data mining based on secure multi-party computation (*SMC*) techniques
 - It is used when several parties own different portions of the data; each party wish to share the data mining results without however disclosing the original data to the other parties

Privacy Preserving Data Mining

Open issues



- Efficiency – especially for techniques based on SMC
- Inference from data mining results
- Metrics to evaluate privacy and data quality
- Privacy-preserving data mining techniques driven by *data quality*

DBMS with support for P3P

The main idea of P3P



- P3P – Platform for Privacy Preferences
- The privacy policies of the sites are published using XML syntax
- Users also specify their privacy requirements
- The user agents can automatically check to see if the policies are compliant
- <http://www.w3.org/P3P/>

DBMS with support for P3P

- DBMS with support for P3P have the goal of providing an integrated support for privacy policies in enterprises
- They are characterized by privacy-related metadata and specialized components that extend DBMS functions and architectures in order to directly support privacy policies expressed according to languages like P3P

Hippocratic Databases

The notion



- The notion of Hippocratic Database
 - Incorporates privacy protection within relational database systems
 - Establishes a number of guiding principles
 - Encompasses an architecture that uses privacy metadata, which consists of privacy policies and privacy authorizations stored in two tables
 - A privacy policy defines for each attribute of a table the usage purpose(s), the external-recipients and retention period
 - A privacy authorization defines which purposes each user is authorized to use
 - Specific to relational data model

Hippocratic Databases

The 10 guiding principles



- **Purpose Specification.** For personal information stored in the database, the purposes for which the information has been collected shall be associated with the information.
- **Consent.** The purposes associated with personal information shall have the consent of the donor of the personal information.
- **Limited Collection.** The personal information shall be limited to the minimum necessary for accomplishing the specified purposes.
- **Limited Use.** The database shall run only those queries that are consistent with the purposes for which the information has been collected.
- **Limited Disclosure.** The personal information stored in the database shall not be communicated outside the database for purposes other than those for which there is consent from the donor of the information

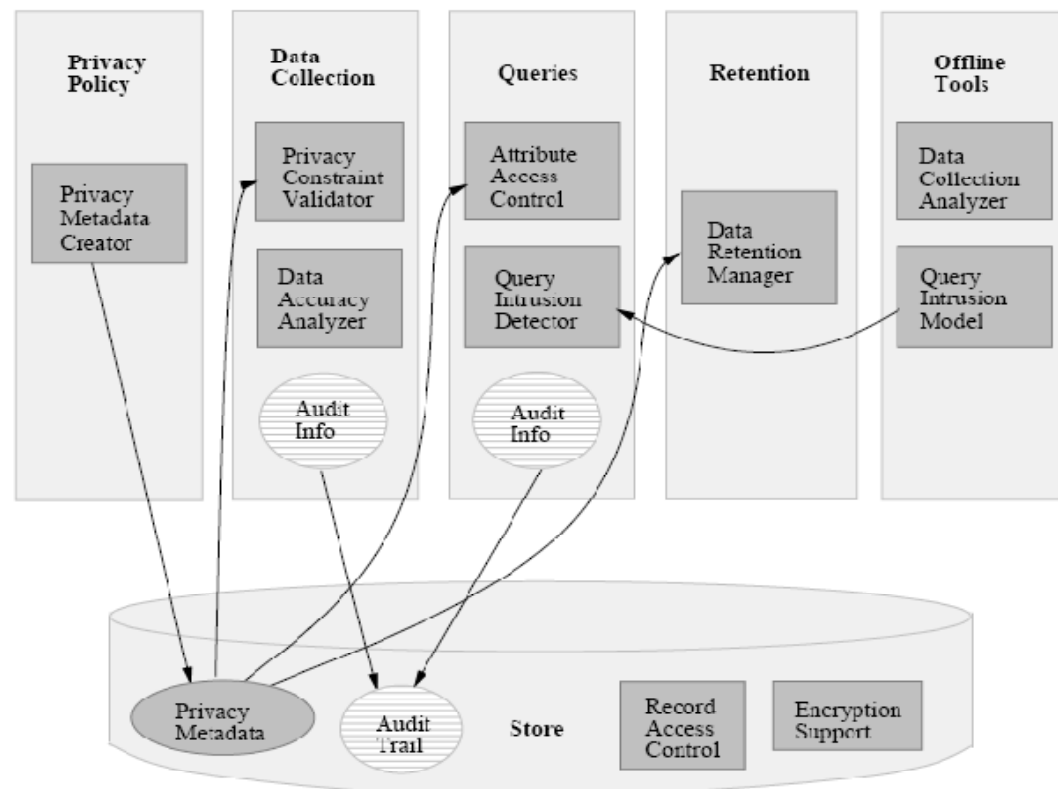
Hippocratic Databases

The 10 guiding principles



- ❑ **Limited Retention.** Personal information shall be retained only as long as necessary for the fulfillment of the purposes for which it has been collected.
- ❑ **Accuracy.** Personal information stored in the database shall be accurate and up-to-date.
- ❑ **Safety.** Personal information shall be protected by security safeguards against theft and other misappropriations.
- ❑ **Openness.** A donor shall be able to access all information about the donor stored in the database.
- ❑ **Compliance.** A donor shall be able to verify compliance with the above principles. Similarly, the database shall be able to address a challenge concerning compliance.

Hippocratic Databases Architecture



P3P – Overview and Critique

P3P and APPEL

- Platform for Privacy Preferences (P3P)
 - A standard means for enterprises to make privacy promises to their users
 - Websites encodes the privacy practice in a machine-readable format (XML)
 - what information is collected, who can access the data for what purposes, and how long the data will be stored by the sites
 - It does not provide any mechanism to ensure that these promises are consistent with the internal data processing

- APPEL is A P3P Preference Exchange Language
 - Specify user's privacy preferences
 - What privacy practice is acceptable
 - Encoded in XML
 - Enable automatic checking against website's P3P policy

Interaction Model of P3P

- Enterprises
 - collect users' information and provide services
 - Specify their privacy policies
- Users
 - Provide necessary information and get service
 - Specify their privacy preferences

P3P Privacy policies

- High level description or promise of an enterprise's privacy practice
 - Encoded in machine-readable XML
 - Posted on their websites
- Major components
 - What information will be collected?
 - For what purpose?
 - Who may see the information?
 - For how long the information will be kept?

P3P Privacy Policies

Main Elements of a Policy

- One ENTITY element: *identifies the legal entity making the representation of privacy practices contained in the policy*
- One ACCESS element: *indicates whether the site allows users to access the various kind of information collected about them*
- One DISPUTES-GROUP element: *contains one or more DISPUTES elements that describe dispute resolution procedures to be followed when disputes arise about a service's privacy practices*
- Zero or more EXTENSION elements: *contain a website's self-defined extensions to the P3P specification*
- One or more STATEMENT elements: *describe data collection, use, and storage. A STATEMENT element specifies the data (e.g. user's name) and the data categories (e.g. user's demographic data) being collected by the site, as well as the purposes, recipients and retention of that data.*

An Example P3P Policy

```
<policies><policy>  
  <Entity> ... </Entity> //describe the website  
  <Access> ... </access> // how to retrieve your data  
  <disputes> ... </disputes> //how to solve disputes  
  <statement>  
    <purpose><admin required=opt-in/></purpose>  
    <recipient><public/></recipient>  
    <retention><indefinitely></retention>  
    <data-group>  
      <data ref=#user.home.postal></data>  
    </data-group>  
  </statement> <statement> ... </statement>  
</policy> </policies>
```

User's Privacy Preferences

- What privacy practice is acceptable
 - Can be viewed as a query on privacy policies
 - Match the preference with privacy policies
 - Only when the query is satisfied, a user should further interact with the enterprise

Limitation of P3P

- Adoption of P3P is slow
 - Only syntax of policy languages is defined
 - Semantics is overlooked
 - Potential inconsistency exists
- The syntax of P3P is very flexible
 - Multiple statements
 - Same data may appear in several statements
 - Not clear what are the relationships between statements

Limitation of P3P (cont'd)

- Different parties thus may have different interpretations of the same policy
 - “The same P3P policy could be represented to users in ways that may be counter to each other as well as the intent of the site.” “... This results in legal and media risk for companies implementing P3P that needs to be addressed and resolved if P3P is to fulfill a very important need.” [Sch02]

Limitation of P3P (cont'd)

- Preference language of P3P
 - Syntax-based: query the representation of a policy instead of its meaning
 - Policies with the same meaning may be treated differently by the same preference
 - Hard to use and error prone

Potential semantic inconsistencies in P3P policies



- ❑ Multiple retention values that apply to one data item
- ❑ Conflicting purposes and retention values
- ❑ Conflicting purposes and recipients
- ❑ Conflicting purposes and data items

A Data-Centric Relational Semantics for P3P [YNA04]



- A P3P policy is mapped into a database with five tables
 - *d-purpose*: <data, purpose, required>
 - *d-recipient*: <data, recipient, required>
 - *d-retention*: <data, retention>
 - *d-category*: <data, category>
 - *d-collection*: <data, optional>
- A preference is thus modeled as a query over the database

Consistency Issues in P3P

- Integrity inconsistency
 - The collection of Alice birthday is both required and opt-in
 - Integrity constraints help detect such conflicts
- Semantic inconsistency
 - Conflicting purpose and retention
 - Historical vs. no-retention
 - Conflicting purpose and data
 - Contact vs. not collecting contact info

Integrity Constraints

- Data-centric constraints
 - Primary key uniquely identifies a tuple
- Vocabulary semantic constraints
 - If purpose is historical, then retention cannot be no-retention
 - If a recipient is public, then retention should be indefinitely
 - Require detailed vocabulary analysis

A Semantics-based Preference Language (SemPref)



- Query the meaning of a policy instead of its representation
 - Easy to understand
 - Much simple structure
 - Easy to design GUI interface for users

Purpose-Based Access Control for Relational Databases

Motivations and Goals

□ Motivation

- Traditional access controls are focused on
 - which users are performing which actions on which data objects.
- However, privacy policies are concerned with
 - which data object is used for which purposes
 - “We will collect and use customer identifiable information for billing purposes and to anticipate and resolve problems with your service.”

□ Goal

- The notion of purpose must play a major role in access control.
 - Access decisions should be made based on purpose.

Motivations and Goals

□ Motivation

- The comfort level of privacy varies from individual to individual and depends on the type of information.
- E.g., disclosing purchase history or browsing habits in return for better service such as site cite personalization.
- E.g., Disclosing address vs. credit card number

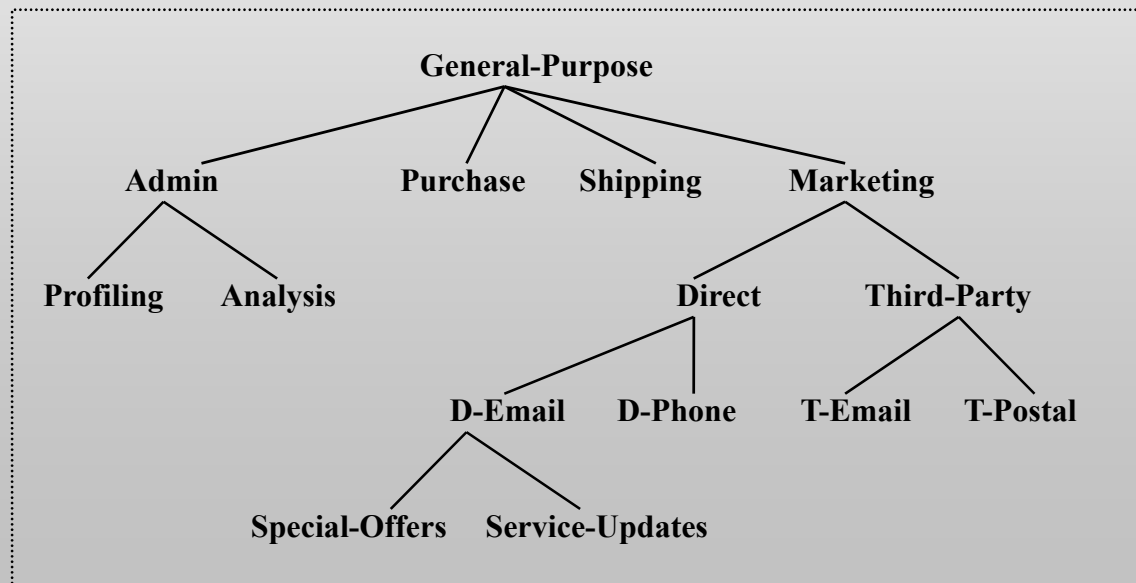
□ Goal

- The granularity of access control must be fine.
 - E.g., tuple-level, cell-level in RDBMS.

Definition of Purpose

□ Purpose

- Describes the reasons data are collected and used
- Organized in a tree structure



Definition of Purpose

- Two types of purpose
 - Intended Purpose
 - Associated with each data item
 - Regulates the usage of data
 - Access Purpose
 - Purpose for accessing a particular data item
 - Associated with data access; i.e. queries

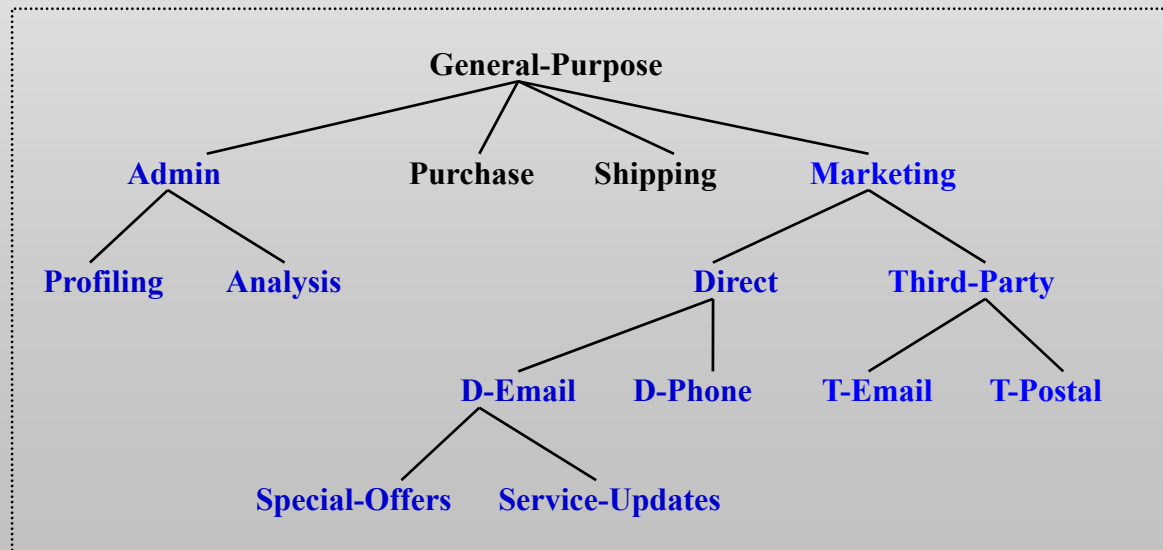
Intended Purpose - Definition

- Intended Purpose
 - Associated with data and regulate data usage
 - $IP = \langle AIP, PIP \rangle$
 - AIP - Allowed Intended Purposes
 - Data access for purposes in AIP is allowed
 - Translation of user preferences
 - PIP - Prohibited Intended Purposes
 - Data access for purposes in PIP is never allowed
 - Restrictions by organizational requirements or privacy laws

Intended Purpose - Entailment

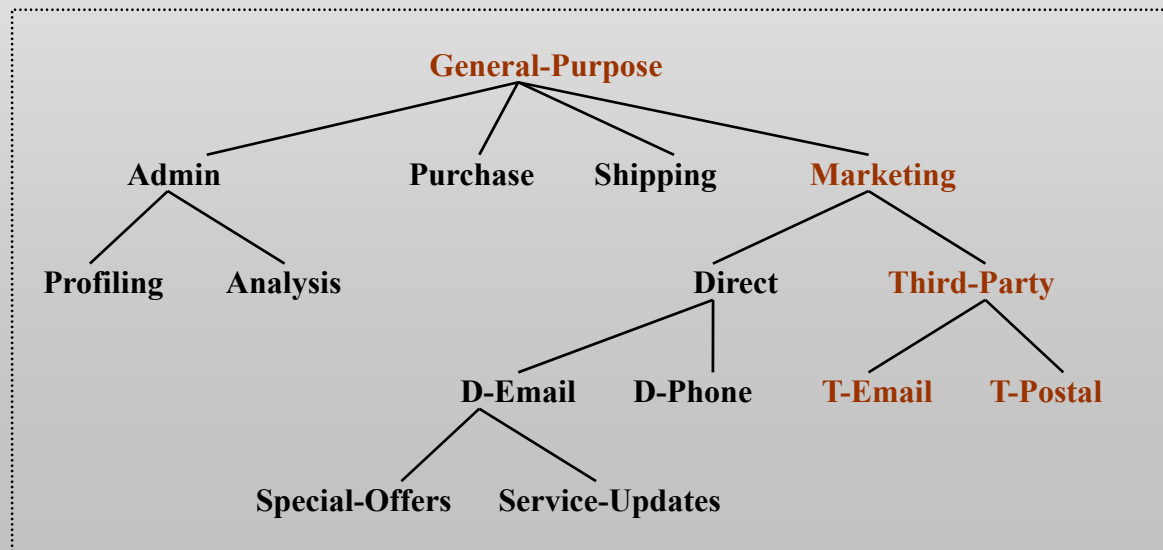
□ Intended Purpose Entailment

- $IP = \langle \{Admin, Marketing\}, \{Third-Party\} \rangle$
- $AIP^\downarrow = \text{Descendants (Admin)} \cup \text{Descendants (Marketing)}$



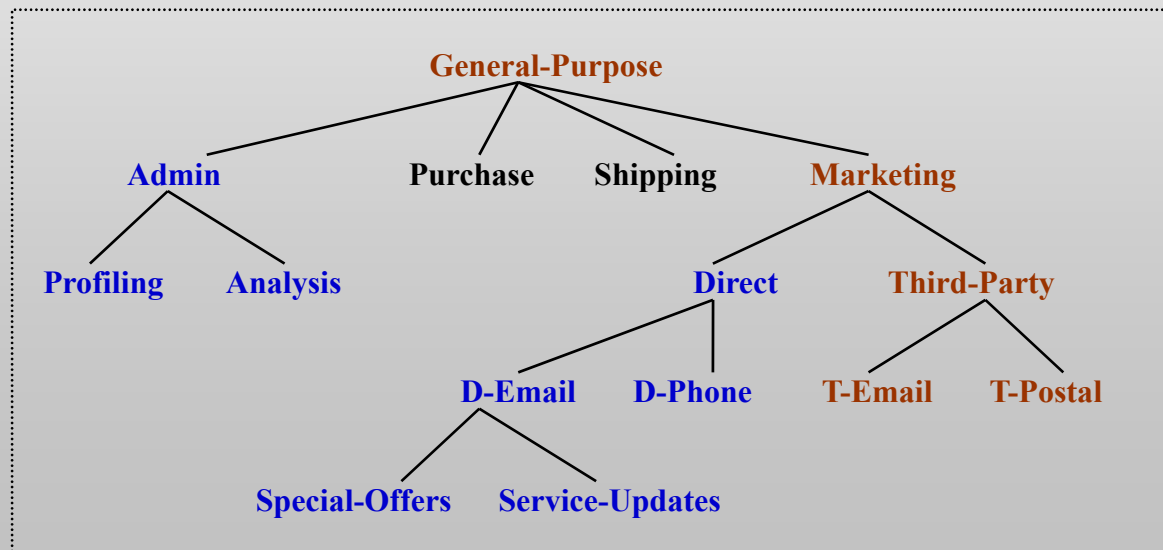
Intended Purpose - Entailment

- Intended Purpose Entailment
 - $IP = \langle \{\text{Admin, Marketing}\}, \{\text{Third-Party}\} \rangle$
 - $PIP^\uparrow = \text{Descendants}(\text{Third-Party}) \cup \text{Ancestors}(\text{Third-Party})$



Intended Purpose - Entailment

- Intended Purpose Entailment
 - $IP = \langle \{\text{Admin, Marketing}\}, \{\text{Third-Party}\} \rangle$
 - $IP^* = AIP^{\downarrow} - PIP^{\uparrow}$



Intended Purpose Labeling

- Intended purposes are associated with a relation R according to one of the following methods.
 1. (Relation-based) a pair $\langle R, ip \rangle$
 - Access to any data element in instances of R is governed by ip
 2. (Attribute-based) a set $\{\langle A_i, ip_i \rangle \mid A_i \wedge \text{Attributes}(R) \wedge ip_i \in IP\}$
 - Access to data element a_i in any instance of R is governed by ip_i
 3. (Tuple-based) a relation scheme $R_{tl}(A_1, \dots, A_n, l)$
 - l is a column having IP for its domain
 - Access to any data element in the j^{th} tuple in any instance of R is governed by l_j
 4. (Element-based) a relation scheme $Rel(A_1, l_1, \dots, A_n, l_n)$
 - l_i ($i = 1, \dots, n$) is a column having IP for its domain
 - Access to data element a_i in any instance of R is governed by l_i

Intended Purpose Labeling

□ Element-based labeling

c_id	c_id_ip	name	name_ip	email	email_ip
1001	<{G}, ∅>	John	<{G}, {M}>	john@aa.edu	<{P, S}, {M}>
1002	<{G}, ∅>	Paul	<{G}, ∅>	p23@oh.com	<{G}, ∅>
1003	<{G}, ∅>	Jack	<{G}, ∅>	Jack03@very.net	<{G}, {T}>

□ Tuple-based labeling

c_id	street	city	state	zip-code	addr_ip
1001	232 Oval Drive	West Lafayette	IN	47907	<{G}, {A, M}>
1002	433 State Road	Chicago	IL	46464	<{G}, ∅>
1003	9898 First Ave	San Francisco	CA	94037	<{G}, {T}>

Purpose Compliance

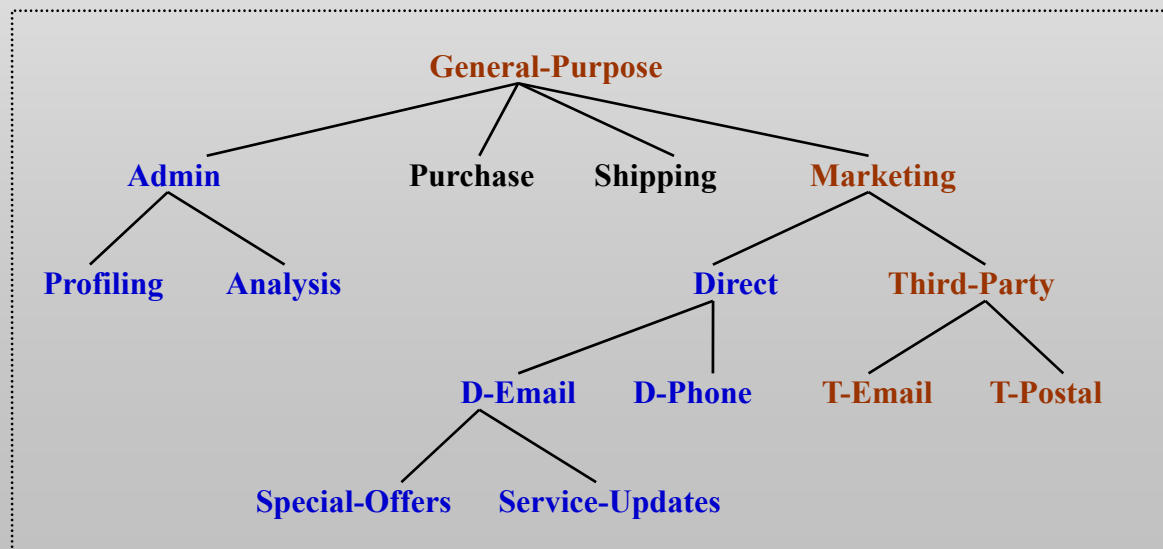
- Intended purposes tell how data should be used.
- Access purpose tells how data will be used.

- Purpose Compliance
 - $AP \Rightarrow_{PT} IP$ iff $AP \in IP^*$
 - $AP \notin PIP^\uparrow$ and $AP \in AIP^\downarrow$

 - Data access is allowed only if $AP \Rightarrow_{PT} IP$

Purpose Compliance - Example

- $IP = \langle \{\text{Admin, Marketing}\}, \{\text{Third-Party}\} \rangle$
 - $AP_1 = \text{D-Email} : AP_1 \Rightarrow_{PT} IP$
 - $AP_2 = \text{T-Email} : AP_2 \not\Rightarrow_{PT} IP$
 - $AP_3 = \text{Marketing} : AP_3 \not\Rightarrow_{PT} IP$



Access Purpose - Definition

- Access Purpose
 - Purpose for accessing a particular data item
 - Associated with each data access (i.e., query)
 - Ex. Select name from customer **For Marketing**

- How do we determine access purposes?
 - That is, how does the access control system determine with what purpose a particular user is trying to access a particular data item using a query?

Access Purpose - Determination

- Possible approaches
 - Users explicitly state their access purposes when querying
 - Need to trust the users
 - Register every application or procedure with an access purpose
 - Not applicable if they are complex
 - Dynamically determine from the current context of the system
 - Difficult to capture all possibilities

Access Purpose - Verification

- Our approaches
 - Users are required to explicitly state their access purposes when querying
 - E.g., Select email from Customer **for Marketing**
 - Then the system verifies if the stated access purposes are valid
 - i.e., The system checks if the user is indeed allowed to access data with the stated purpose for a given circumstances

Access Purpose - Verification

- To facilitate the verification process, users are granted authorizations for access purposes.
 - Now the problem becomes similar to authorizing access permissions.

 - We can rely on RBAC model
 - Roles and access purposes have close relations.
 - RBAC is already used in many systems.

Access Purpose - Verification

- Limitations of RBAC
 - Insensitive to system environments.
 - E.g., Time-of-day, location, application-type
 - Access purposes should be sensitive to such contexts.
 - The RBAC administration is not easy.
 - Exceptions are not allowed.
 - May require frequent reconstructions of roles.

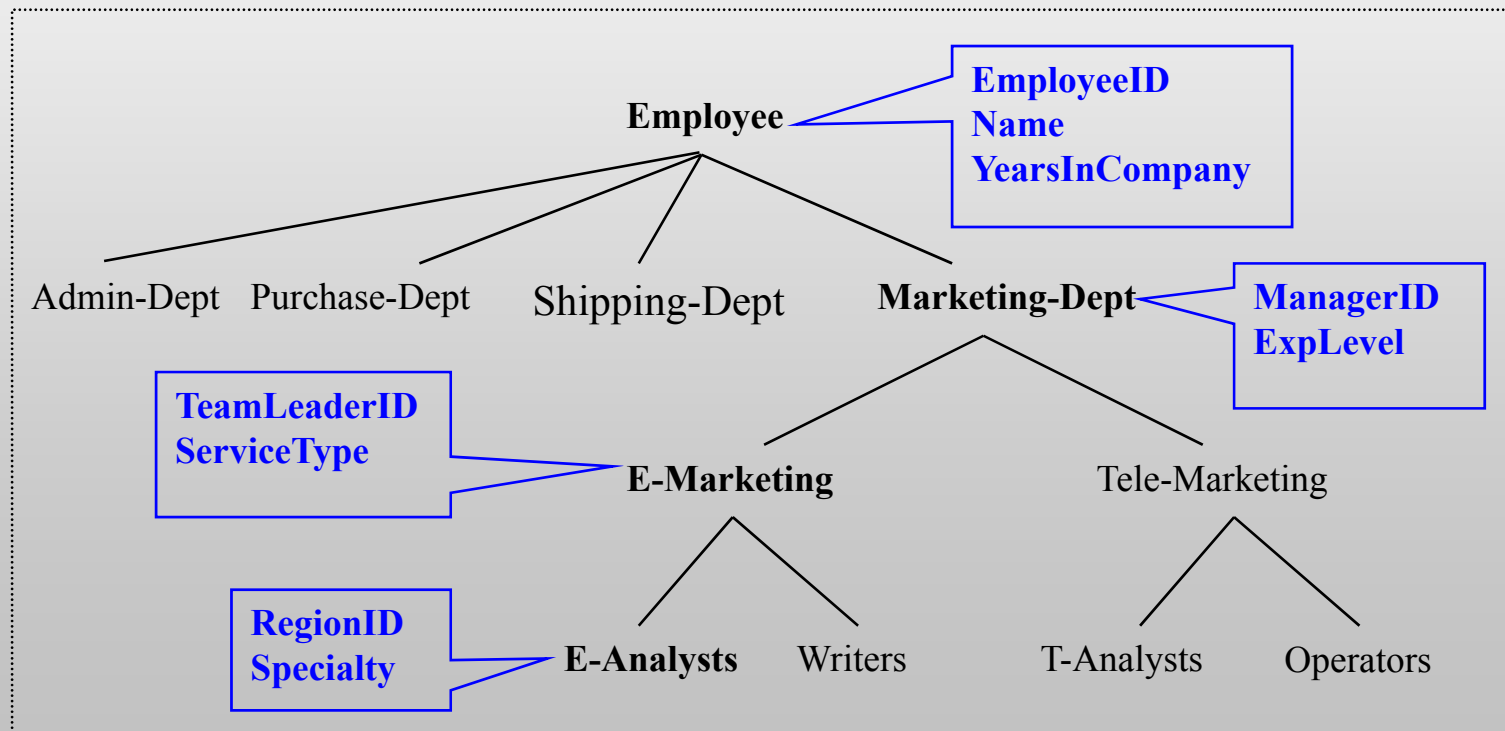
Access Purpose - Determination

□ Role Attributes

- Every role r is associated with a set of attributes that are defined for r or inherited from the ancestor roles of r .
- When a user is assigned to a role r , the values for the role attributes of r are specified for the user.
- The role attribute values of the user are available to the access control system from the time the user activates r to the time the user deactivates r .

Access Purpose - Determination

□ Example of role attributes



Access Purpose - Verification

□ System Attributes

- Given a system S , a set of attributes is available to the access control system at all times.
- The system attributes are defined by system administrators for the application needs.
- The values of the system attributes in a system state s specify the environment of the system in the state s .

Access Purpose - Verification

□ Conditional Role

- A conditional role cr is defined as a 2-tuple $\langle r, C \rangle$.
 - $r \in \mathbf{R}$
 - C is a finite propositional logic formula which may use the logical operators \wedge and \vee , and each predicate is of the form $x \phi y$, where $x \in r.\text{Attributes}$ or $x \in \mathbf{S}.\text{Attributes}$ and $y =$ a constant, and $\phi \in \{<, \leq, >, \geq, =, \neq\}$.
 - E.g., $\langle \text{E-Marketing}, (\text{time} \geq 9 \text{ am}) \wedge (\text{time} \leq 5 \text{ pm}) \rangle$

- An activated role r belongs to a conditional role $cr_i = \langle r_i, C_i \rangle$ in a system state s if and only if the following conditions are satisfied:
 1. $r \in \text{Descendants}(r_i)$
 2. The evaluation of C_i under $r(u).\text{Attributes}$ and $\mathbf{S}(s).\text{Attributes}$ is true.

Access Purpose - Determination

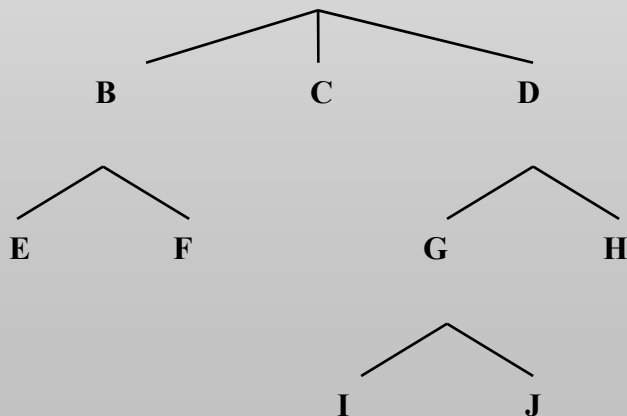
□ Access Purpose Verification

- Given an access purpose ap and a role r activated by a user u , ap is valid for u under r if there exists an access purpose authorization $\langle ap_i, cr_j \rangle$, where $ap_i \in \mathbf{P}$ and $cr_j = \langle r_j, C_j \rangle$ is a conditional role defined over \mathbf{R} and \mathbf{S} , satisfying the following conditions:

1. $ap \in \text{Descendants}(ap_i)$
2. r belongs to the conditional role cr_j .

Implementation - Metadata Storage

- Purpose Tree
 - Purposes are encoded as bit strings
 - Stored in *pt_table*



p_id	p_name	parent	code	aip_code	pip_code
1	A	-	0x200	0x3FF	0x3FF
2	B	1	0x100	0x130	0x330
3	C	1	0x080	0x080	0x280
4	D	1	0x040	0x04F	0x24F
5	E	2	0x020	0x020	0x320
6	F	2	0x010	0x010	0x310
7	G	4	0x008	0x00B	0x24B
8	H	4	0x004	0x004	0x244
9	I	7	0x002	0x002	0x24A
10	J	7	0x001	0x001	0x249

Ex) A = '1000000000'
 {A}* = '1111111111'

Implementation - Metadata Storage

- Intended purpose labels for a relation with n columns
 - Relation-based Labeling
 - A single entry in privacy policy table
 - Attribute-based Labeling
 - n entries in privacy policy table
 - Tuple-based Labeling
 - Extended with $(n + 2)$ columns
 - Element-based Labeling
 - Extended with $(n + 2n)$ columns

Implementation

□ Purpose Compliance Check

- *Comp_Check* (Number ap, Number aip, Number pip)

Returns Boolean

1. if (ap & pip) \neq 0 then
2. return False;
3. else if (ap & aip) = 0 then
4. return False;
5. end if;
6. return True;

- Requires two bitwise-AND operations.

Implementation

□ Access Control by Query Modification

■ Modifying_Query (Query Q)

Returns a modified privacy-preserving query Q'

1. Let R_1, \dots, R_n be the relations referenced by Q
2. Let P be the predicates in WHERE clause of Q
3. Let a_1, \dots, a_m be the attributes referenced by Q, attributes in both projection list and P
4. Let AP be the access purpose encoding of Q
5. for each R_i where $i = 1, \dots, n$ do
6. if (R_i is relation-based labeling AND
 $\text{Comp_Check}(AP, R_i.aip, R_i.pip) = \text{False}$ then
7. return ILLEGAL-QUERY;
8. else if R_i is attribute-based labeling then

Implementation

11. for each a_j which belongs to R_i do
12. if $\text{Comp_Check}(AP, a_j.\text{aip}, a_j.\text{pip}) = \text{False}$ then
13. return ILLEGAL-QUERY;
14. end if;
15. end for;
16. else if R_i is tuple-based labeling then
17. add ‘AND $\text{Comp_Check}(AP, R_i.\text{aip}, R_i.\text{pip})$ ’ to P ;
18. else if R_i is element-based labeling then
19. for each a_j which belongs to R_i do
20. add ‘AND $\text{Comp_Check}(AP, a_j.\text{aip}, a_j.\text{pip})$ ’ to P ;
21. end for;
22. else // R_i is a relation without labeling
23. do nothing;
24. end if;
25. end for;
26. return Q with modified P ;

Implementation

□ Example of Query Modification

```
Select name, phone  
From customer  
For Marketing
```



Customer table : Element-based Labeling
Marketing = '512'

```
Select name, phone  
From customer  
Where comp_check(512, name_aip, name_pip)  
and comp_check(512, phone_aip, phone_pip)
```

Implementation

□ Example of Query Modification

```
Select name, city  
From customer as C, address as A  
Where C.c_id = A.c_id  
For Shipping
```



Customer table : Element-based Labeling
Address table : Tuple-based Labeling
Shipping = '1024'

```
Select name, city  
From customer as C, address as A  
Where C.c_id = A.c_id  
    and comp_check(1024, Address_aip, Address_pip)  
    and comp_check(1024, name_aip, name_pip)  
    and comp_check(1024, A.c_id aip, A.c_id pip)
```

Implementation

□ Example of Query Modification

```
Select product
From order
Where c_id = 1101
For Profiling
```

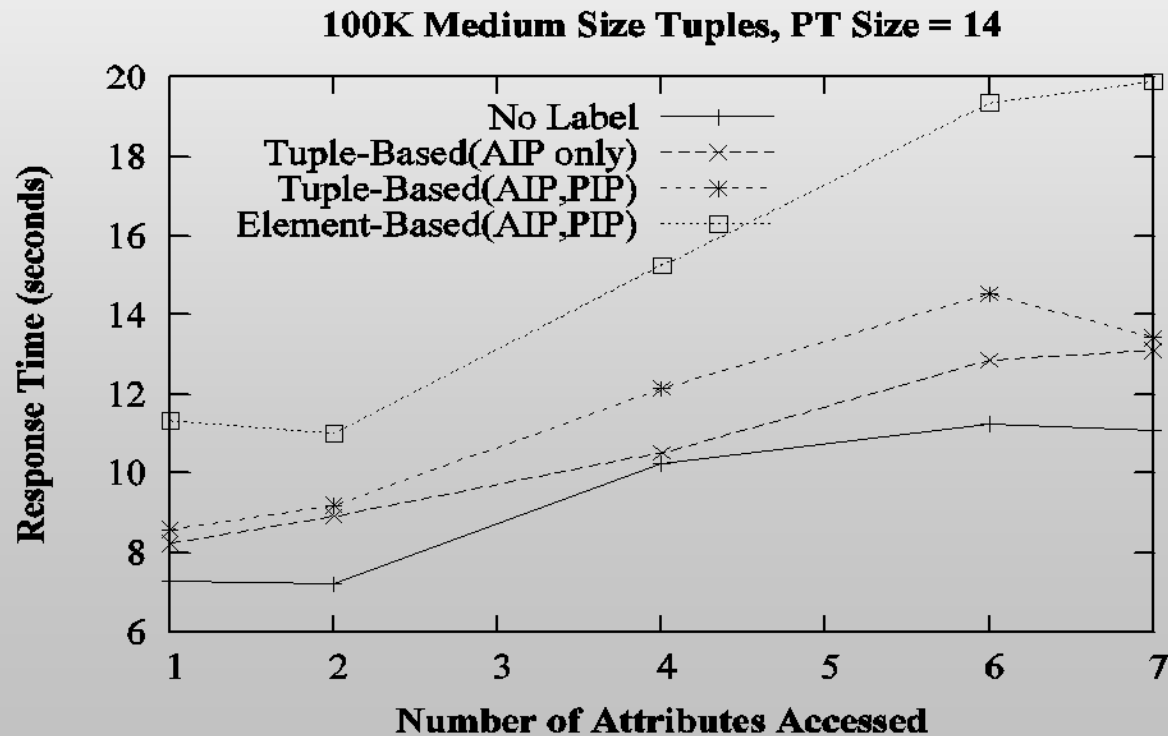


**order table : Relation-based Labeling
Profiling = '256'**

```
Select product
From order
Where c_id = 1101
```

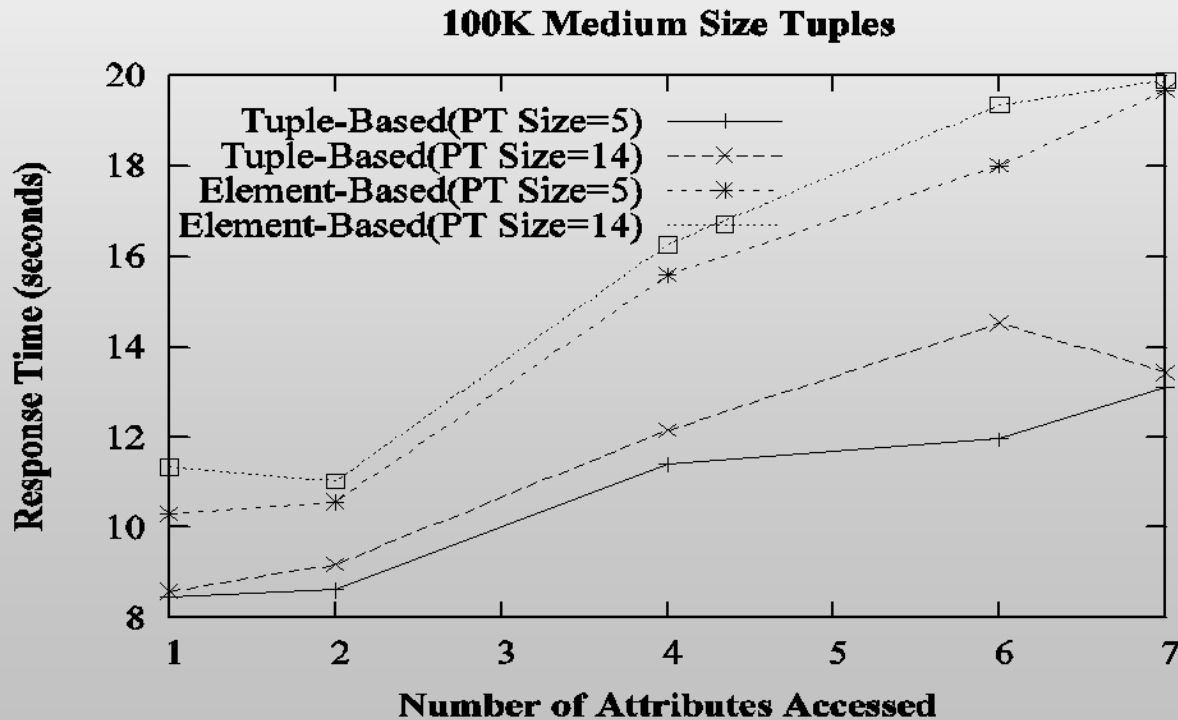
Experimental Results

□ Labeling Scheme and Performance



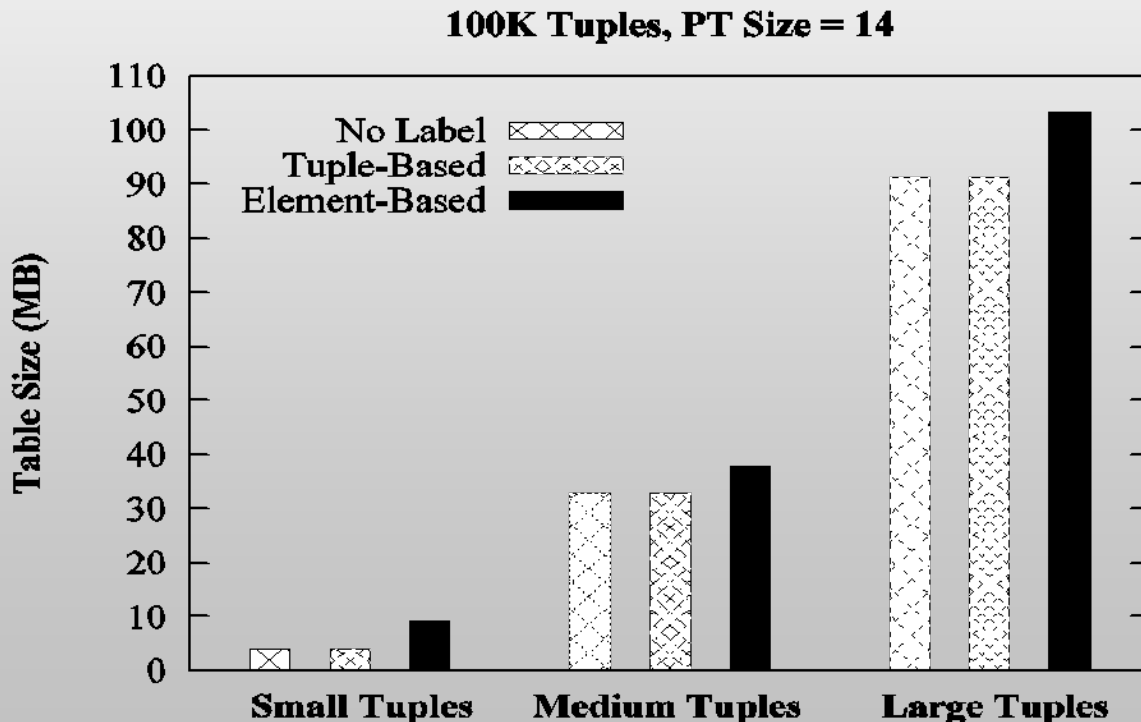
Experimental Results

□ Purpose Size and Performance



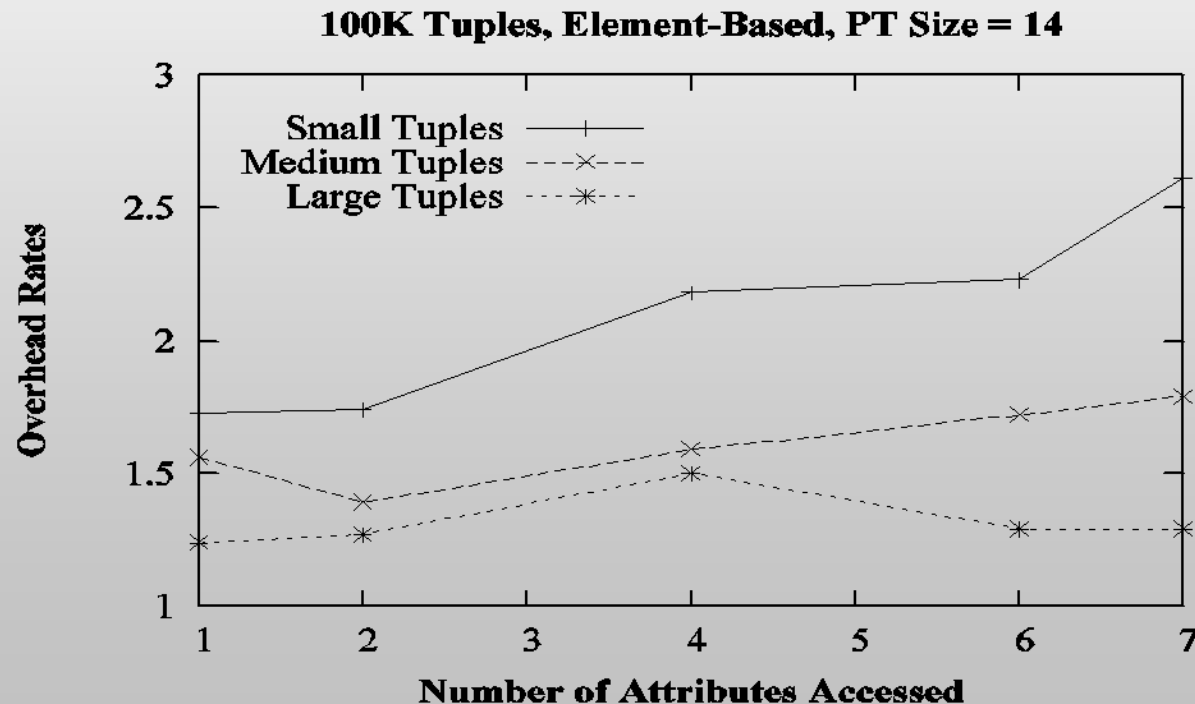
Experimental Results

□ Storage Overhead



Experimental Results

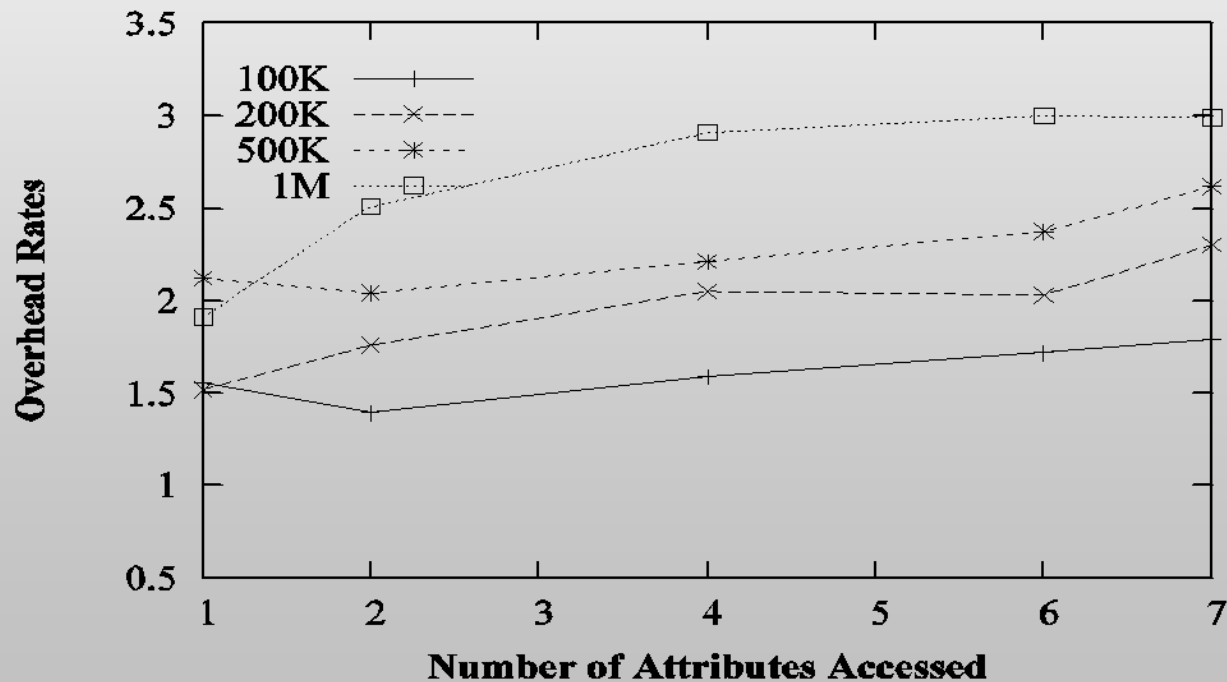
□ Storage Overhead and Performance



Experimental Results

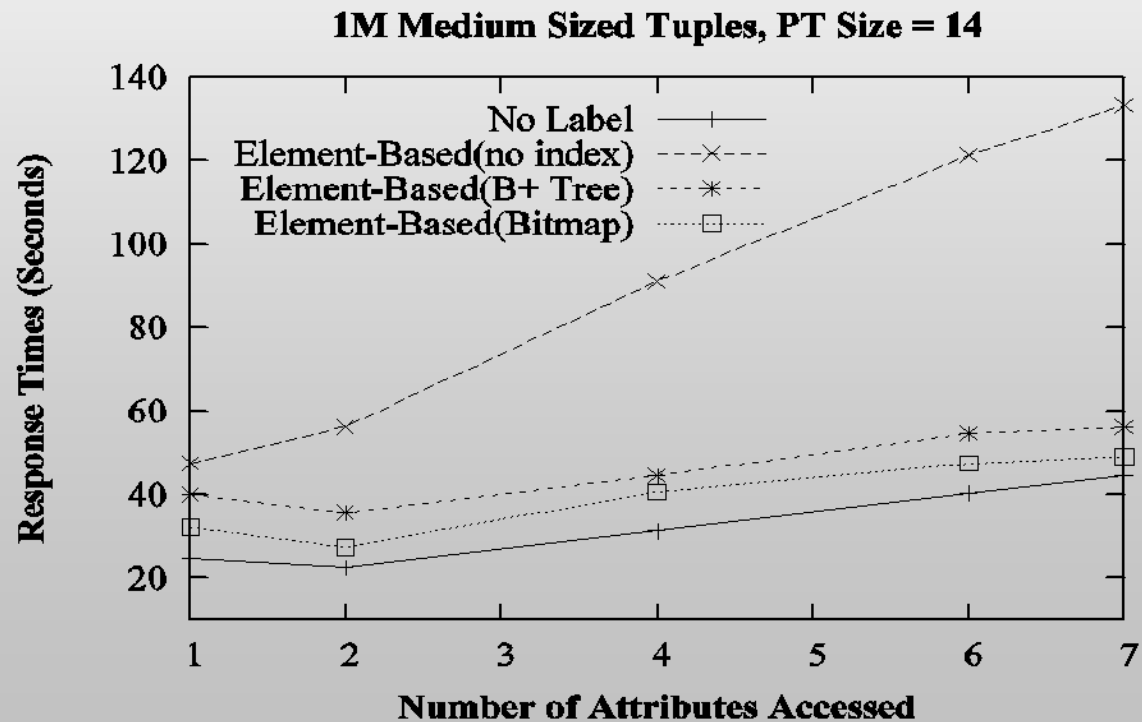
□ Cardinality and Performance

Medium Sized Tuple, Element-Based, PT Size = 14



Experimental Results

□ Cardinality and Performance (Using index)



Purpose-Based Access Control for Complex Objects

Intended Purpose Labeling for Complex Objects



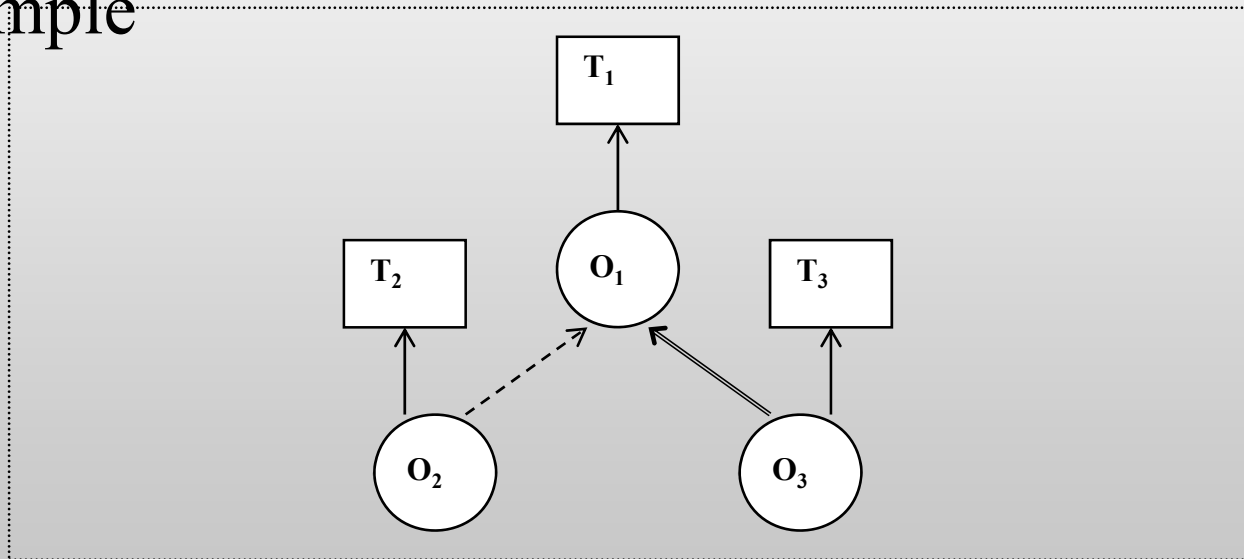
- Motivations
 - Labeling relational data is straightforward.
 - What about complex data with hierarchies?
 - E.g., XML, file systems, object-oriented data

Hierarchical Data Model

- Hierarchical Data Model, \mathcal{H}
 - Represented as a 5-tuple $\langle T_{\mathcal{H}}, O_{\mathcal{H}}, IO_{\mathcal{H}}, SO_{\mathcal{H}}, RO_{\mathcal{H}} \rangle$
 1. $T_{\mathcal{H}}$ is a set of types.
 2. $O_{\mathcal{H}}$ is a set of objects.
 3. $IO_{\mathcal{H}}: O_{\mathcal{H}} \rightarrow T_{\mathcal{H}}$ is a function that assigns a type to each object.
 4. $SO_{\mathcal{H}}: O_{\mathcal{H}} \rightarrow O_{\mathcal{H}}$ is a function that assigns a parent to each object.
 5. $RO_{\mathcal{H}}$ is a function that maps the reference-of relations between objects.

Hierarchical Data Model

□ Example



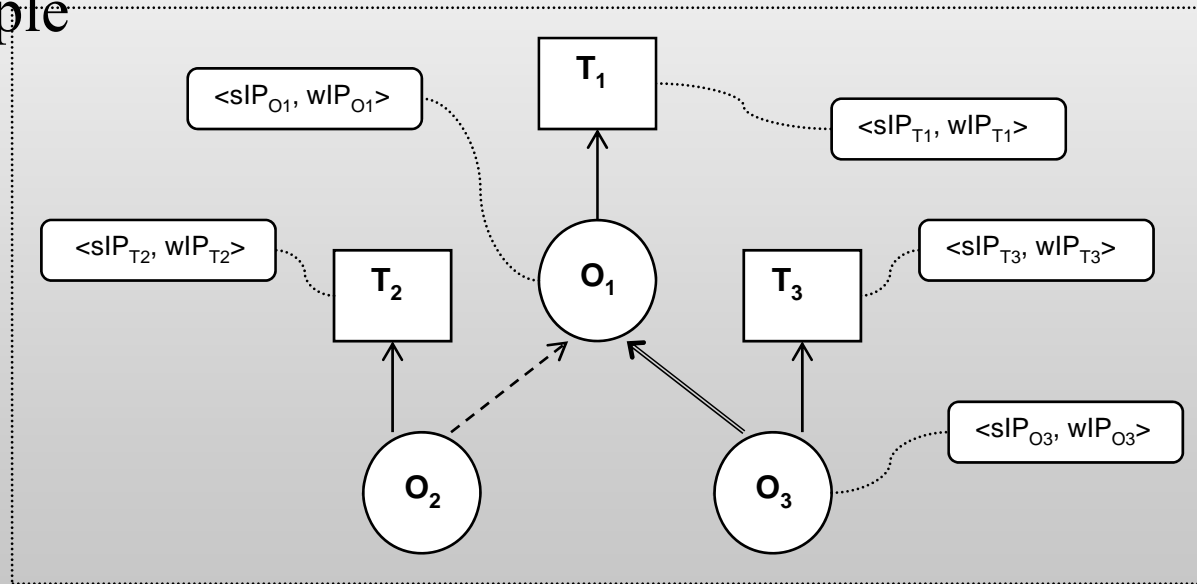
□: types, O: objects, \rightarrow : instance-of relations,
 \Rightarrow : subelement-of relations, $--\rightarrow$: reference-of relations

Intended Purpose Labeling

- Issue of granularity
 - Fine-grained access control requires
 - Fine-grained labeling (i.e., label every data item)
 - However, it may result in redundant storage.
 - We should allow coarse-grained labeling.
 - Solution: implicit authorization
 - A purpose label at a type applies to all objects that are instances of the type.
 - Similarly, a purpose label at an object applies to all the subelements of the object.
 - It does not apply to the reference-of relations.

Hierarchical Data Model

□ Example



□: types, O: objects, \rightarrow : instance-of relations,
 \Rightarrow : subelement-of relations, $--\rightarrow$: reference-of relations

Conflict Resolution

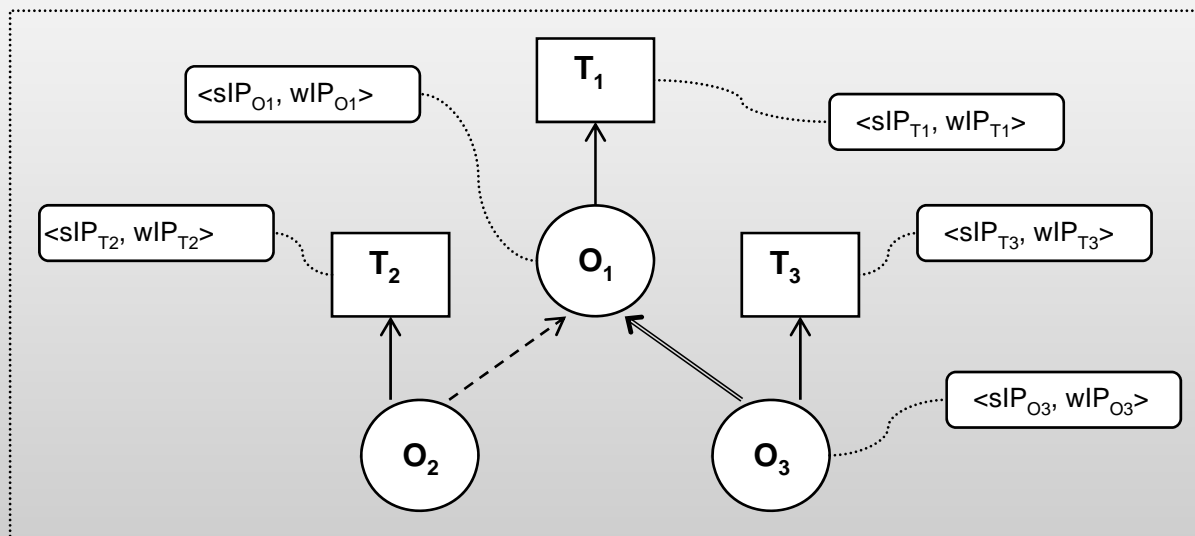
- The labeling scheme allows conflicts between labels.
 - A purpose label at a type t prohibits any instance of t from being accessed for Marketing.
 - A purpose label at an object o , which is an instance of t , may allow o to be accessed for Marketing.

- Two types of purpose label
 - Strong intended purpose label
 - It cannot be overridden by an intended purpose of an instance (or sub-element).
 - Weak intended purpose label
 - It can be overridden by an intended purpose of an instance (or sub-element).

Correctness Requirements for Labels

- IP Label : $\langle (sAIP, sPIP), (wAIP, wPIP) \rangle$
- Well-formed labels: ensures no conflict in a label.
 1. $(sAIP^\downarrow - sPIP^\uparrow) \cap wPIP^\uparrow = \emptyset$, that is, purposes allowed by the strong intended purpose cannot be prohibited by the weak intended purpose.
 2. $sPIP^\uparrow \cap (wAIP^\downarrow - wPIP^\uparrow) = \emptyset$, that is, purposes prohibited by the strong intended purpose cannot be allowed by the weak intended purpose.
- Consistent labels: ensures no conflict in a hierarchy.
 1. $(sAIP_A^\downarrow - sPIP_A^\downarrow) \cap sPIP_D^\uparrow = \emptyset$, that is, purposes strongly allowed at a node cannot be strongly prohibited at a descendant node.
 2. $sPIP_A^\downarrow \cap (sAIP_D^\downarrow - sPIP_D^\uparrow) = \emptyset$, that is, purposes strongly prohibited at a node cannot be strongly allowed at a descendant node.

Intended Purpose Inference



- To get an effective IP of an object,
 - intended purposes are inferred in the top-down order.
 - E.g., The effective IP of O₃ = ((T₁ + O₁) + T₃) + O₃

Inference Algorithms

Function *Get_Effective_IP* (Object o)

Input: an object o

Output: the effective intended purpose of o

if (o.parent is null) then

 return Merge_IP(o.type.IP, o.IP);

else

 IP temp = Get_Effective_IP(o.parent);

 temp = Merge_IP(temp, o.type.IP);

 return Merge_IP(temp, o.IP);

end if;

End;

Inference Algorithms

Function *Merge_IPs*(IP ip1, IP ip2)

Input: two intended purposes to be merged, ip1 and ip2

Output: the merged intended purpose, which ip2 is merged over ip1;

i.e., ip2 overrides ip1 in case of conflict.

if (ip1 is empty) then

 return ip2;

else (if ip2 is empty) then

 return ip1;

else

 merged = create a new IP;

 merged.sAIP = (ip1.sAIP)↓ ∪ (ip2.sAIP)↓;

 merged.sPIP = (ip1.sPIP)↑ ∪ (ip2.sPIP)↑;

 merged.wAIP = (ip1.wAIP)↓ ∪ (ip2.wAIP)↓;

 merged.wPIP = ((ip1.wPIP)↑ – (ip2.wAIP)↓) ∪ (ip2.wPIP)↑;

 return merged;

end if;

End;

Inference Algorithms

- Complexity
 - the computational complexity in $O(khn)$ and
 - the spatial complexity in $O(kn)$,
 - where k is the maximum height of the data hierarchy, h is the height of the purpose tree, and n is the total number of purposes.

- Correctness
 - If all intended purpose labels in the system are well-formed and consistent, then the effective intended purposes generated by IPI Algorithm are also well-formed and consistent.
 - Proof (see paper [BBL05]).

Query Compliance

- Let $Q = \langle o, ap \rangle$ be a query accessing an object o with the access purpose ap . Let $EIP_o = \langle sEIP_o, wEIP_o \rangle$ be the effective intended purpose of o . Q is said to be compliant to the intended purpose of o , denoted as $Q \Rightarrow o$, if and only if one of the following conditions satisfies:
 1. $ap \Rightarrow sEIP_o$; i.e., the access purpose of the query is compliant to the strong effective intended purpose of the data, or
 2. $ap \Rightarrow wEIP_o$; i.e., the access purpose of the query is compliant to the weak effective intended purpose of the data.

Generalized Fine Grained Access Control Models for Relational Databases

Approaches

- View-based approach
 - Proposed by Stonebraker and Wong for INGRES
 - Supported by commercial DBMS
 - It has several drawbacks

- Virtual Private Database (VPD) – Oracle

- Truman model

Challenges

- To develop declarative policy languages for fine-grained access control in database systems
- To develop query processing that enforce fine-grained access control with the following three properties:
 - Soundness
 - Security
 - maximality

Conclusions

- Other topics related to database security
 - Integrity and availability
 - Protection from insider threats through ID techniques
 - Support for operations on encrypted data – useful when dealing with outsourced databases
 - Support for private information retrieval

Selected References

- [AKS02] R.Agrawal, J.Kiernan, R.Srkant, and Y. Xu. Hippocratic databases. *Proceedings of VLDB Conference, 2002.*
- [BBL04] J.W. Byun, E.Bertino, N.Li. Purpose based access control for privacy protection in relational database systems. Technical Report 2004-52, CERIAS, Purdue University, 2004.
- [BBL05] J.W. Byun, E.Bertino, N.Li. Purpose based access control of complex data for privacy protection. *Proceedings of SACMAT, 2005.*
- [BS05] E.Bertino, R.Sandhu. Database security – concepts, approaches, challenges. *IEEE Transactions on Dependable and Secure Computing, 2(1):2-19, 2005.*
- [CV04] C.Clifton, J. Vaidya. Privacy-preserving data mining: Why, how, and when. *IEEE Security and Privacy, 2(6):19-27, 2004.*

Selected References

- [RMS04] S. Rivzi, A. Mendelzon, S. Sudarshan, and P. Roy. Extending query rewriting techniques for fine-grained access control. *Proceedings ACM SIGMOD Conference*, 2004.
- [Sch02] D.M. Schutzer. Citigroup P3P position paper. W3C Workshop on the Future of P3P. Available at <http://www.w3.org/2002/p3p-ws/pp/ibm-zuerich.pdf>
- [Swe02] L. Sweeney. *K*-anonymity: a model for protecting privacy. *International Journal on Uncertainty, Fuzziness and Knowledge-based Systems*, 10(5):557-570, 2002.
- [YLA04] T. Yu. N. Li, A. Anton. A Formal Semantics for P3P. *Proceedings of ACM Workshop on Secure Web Services (SWS)*, 2004.