Expressing Privacy Policies using Authorization Views

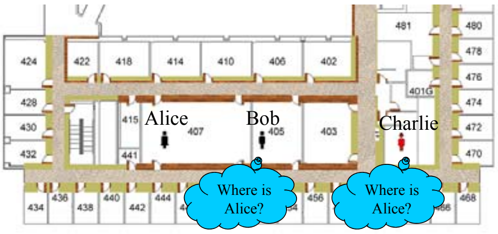
Vibhor Rastogi
RFID Security Group
University of Washington

V. Rastogi, E. Welbourne, N. Khoussainova, T. Kriplean, M. Balazinski, G. Borriello, T Kohno, D. Suciu

Introduction

- Ubiquitous context-aware computing systems
  - Interaction depends on context information
- RFID Ecosystem
  - An ubiquitous computing system at UW CSE
  - Building wide deployment of RFID readers
  - Users and objects are tagged
  - Information streamed to a central server
  - Users query the central server

RFID Ecosystem



Privacy issue: Access control

- Suppose a user asks a query
  - Is the answer public or private?
  - It depends on multiple factors [Belloti et. al.]
  - Context of the *Querier* and of the *Subject*
- Rule-based access control
  - Rules control the accessible information
  - Need to incorporate all the above factors
- Two Problems
  - Hard for users to manage [Lederer et. al.]
  - Context is often *inferred* and *uncertain* in nature

Our approach

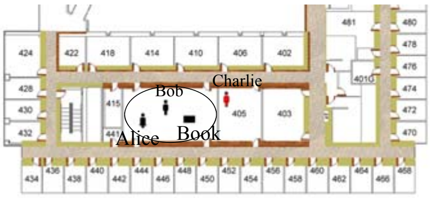
- Principles for designing access control policy
  - A constrained space of predefined rules
    - Less expressive, more usable
  - Rules intuitive for users to understand
    - Reflect modes of information access in the real world
    - Pertain to concrete events (Eg. Meeting)
- Implementation of access control policy
  - Use *Authorization views*
    - Allow us to efficiently handle inference & uncertainty

Agenda

- PAC rule for the RFID Ecosystem
- Extensions to PAC
  - Meeting Rule
  - Ownership rule
- General Design principles
- Authorization views
- Conclusion

PAC Rule

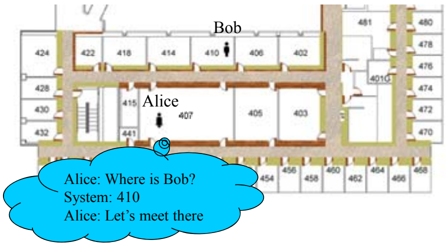
- Proposed by [Kriplean 07]



PAC Rule (Contd.)

- Provides a default level of privacy
- Enables many applications
  - Personal diary
    - Find information about past events, meetings & locations
  - Object tracker
    - Find the last location where the object was seen

The meeting scenario



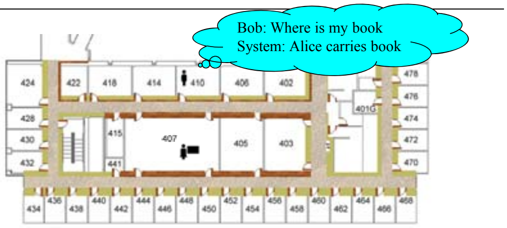
The meeting rule

- For this scenario, Bob enables the meeting rule

If A & B have Meeting then release B's location to A

- Bob is the controller
- Bob is also the subject

The ownership scenario



The ownership rule

- For this scenario, Bob enables the ownership rule

If A carries B's object then release B carries object to A

- Bob is the controller
- Alice is the subject

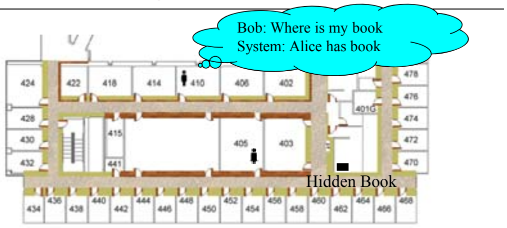
Extensions to PAC (contd.)

- Many possible scenarios and rules
  - If *context* then release *information* to *user*
- Rules classified into categories based on *context*
- Context can be deconstructed [Lederer 03]
  - Location-based (Where)
  - Event-based (When)
  - Role-based (Who)
  - Intention-based (Why)
  - Ownership-based (What)

General Design Principles

- Controller vs. Subject
  - If controller ≠ subject, access rule may be unsafe
  - For ownership rule, Alice's exact location hidden
- Choosing the right context critical
  - For ownership rule, context = *Alice carries book*

General Design Principles (contd.)



PEEX [Khoussainova et. al.] to infer context

Authorization views

- A database technique for fine grained access control
- For each rule an AV is defined
- A logical table that stores all accessible information
- User query on the original tables
  - Rewritten in terms of authorization views [Duschka]

Using authorization views

- Data stored in the table LocatedAt
  - LocatedAt(User, Location , Time)
- Each rule translated into AV

User	Location	Time
Alice	Atrium	5:45 PM
Bob	Atrium	5:45 PM
Bob	Kitchen	5:30 PM

LocatedAt

User U	User A	Location	Time
Alice	Bob	Atrium	5:45 PM
Bob	Alice	Atrium	5:45 PM

PACView = LocatedAt (U, L ,T) $\wedge$ LocatedAt(A, L, T)

Conclusion

- Designing simple & intuitive rules important
- We design ACP for the RFID Ecosystem
  - General design principles for safer & simple access control policies
- Authorization views
  - Simple and Flexible implementation