

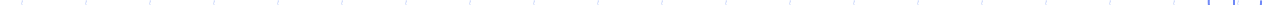


Designing a Publish Subscribe Substrate for Privacy/Security in Pervasive Environments

Lukasz Opyrchal
Miami University
Oxford, OH

Atul Prakash
University of Michigan
Ann Arbor, MI

Amit Agrawal
Indian Institute of Technology
New Delhi, India



Introduction

◆ Emerging pervasive applications

- RFID tags and other sensors
 - ◆ Tracking objects, people, etc.
- Cell phones
 - ◆ Location-based services

◆ Privacy concerns

- People do not wish to have their movements available to everyone¹

¹ R. J. Harper, 1995

Privacy

- ◆ The ability of an individual to control the terms for acquisition and usage of their personal information
- ◆ How to build applications and services while providing means to users to have control over the conditions of distribution of their data

Policies we are interested in

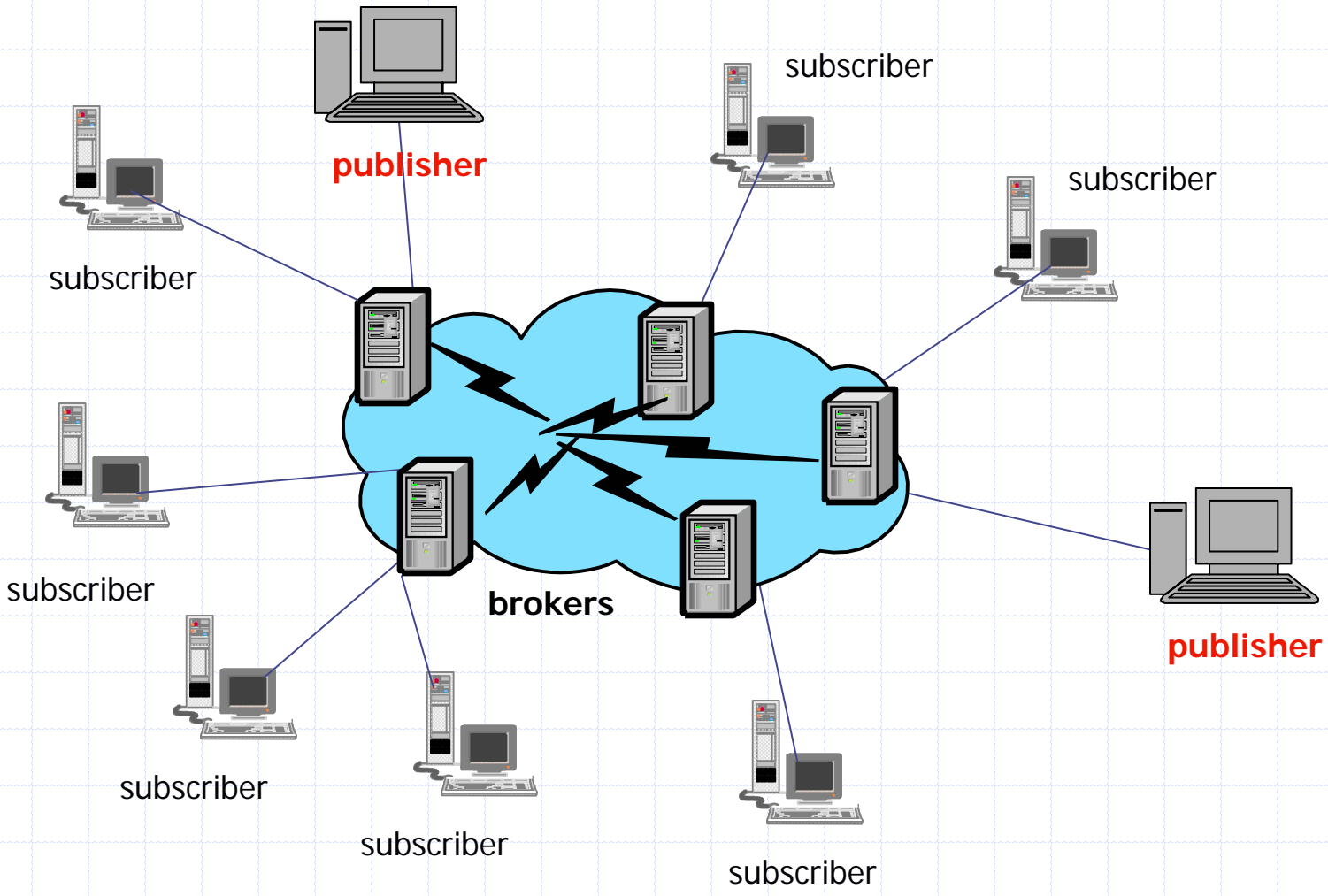
- ◆ Environment-dependent sharing
 - Share info at certain times,
 - Share info in certain locations,
 - Share info during special events, etc.
- ◆ Privacy-protected access to services
 - Location-based notification
 - Without revealing ones location

Rest of presentation

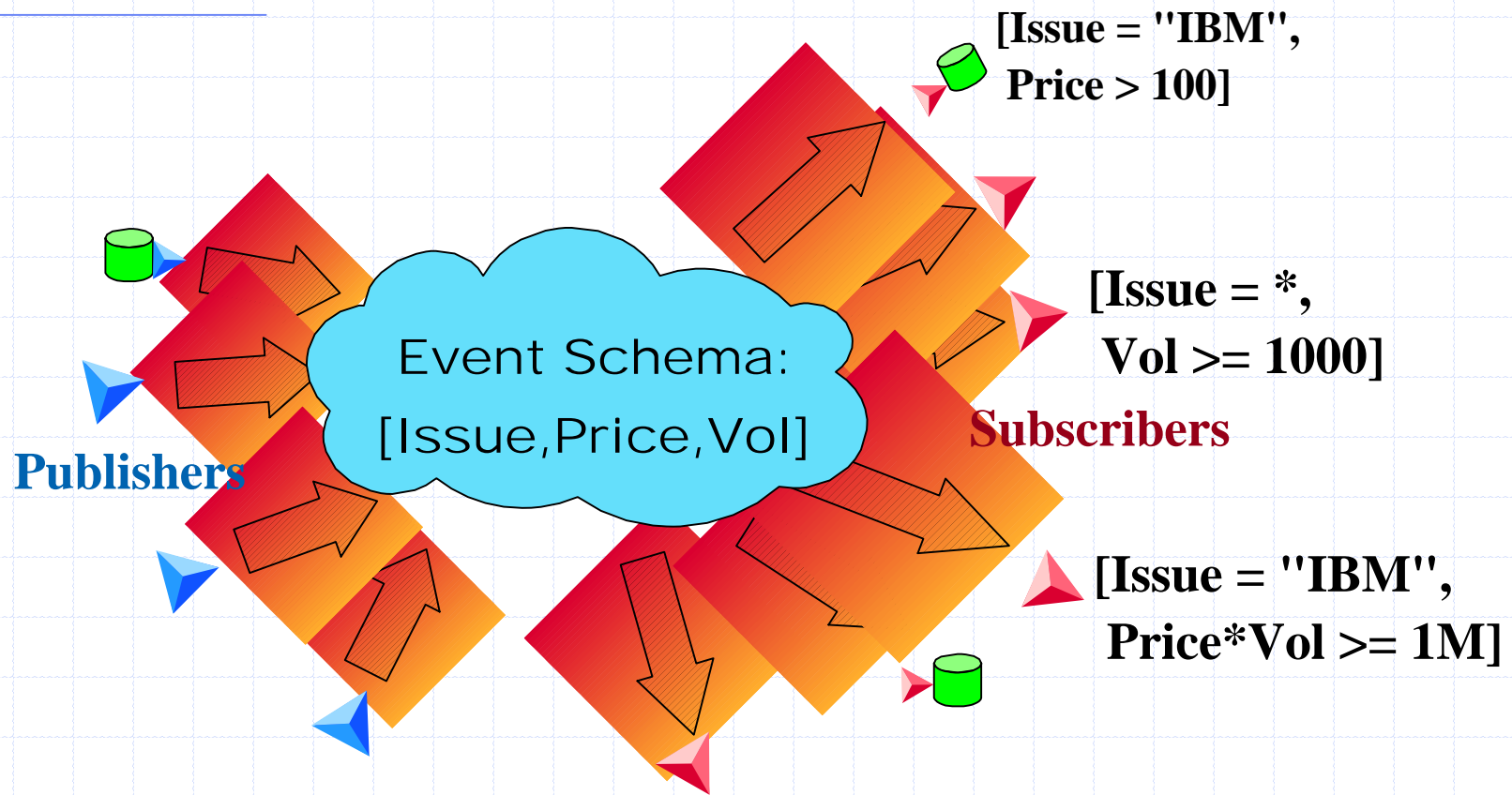
- ◆ Content-based publish subscribe
- ◆ Policy model
- ◆ Prototype publish subscribe system

- ◆ Location tracking application

Publish Subscribe Systems



Content-Based Publish Subscribe



- SIENA, Elvin, Hermes
- IBM: Gryphon
- Microsoft: Herald

- Only rudimentary security solutions exist

Policy Dimensions

- ◆ Authorization/Authentication
 - existing solutions (Kerberos, certificates, etc.)
- ◆ Access Control
 - conditions under which an action can be performed
 - historically – coarse-grained
- ◆ Data Security
 - security guarantees (confidentiality, integrity, sender authenticity, etc.)
- ◆ Granularity of Security Guarantees
 - explained later

Entities

- ◆ Administrators
 - “high level” control over applications
- ◆ Owners
 - can authorize other entities to perform actions
- ◆ Publishers
- ◆ Subscribers
 - users, application, services (event filters)
- ◆ Event Delivery System
 - broker network

Entities

◆ Application

- application administrator
- consists of multiple event types
- LOC_APP application:
 - ◆ LOC_INFO and LOC_SERVICE event types

◆ Event type

- describes event schema

◆ Owner

- can authorize others to subscribe, receive and modify policy for its events
- one or more owners per event type

Policy Language

- ◆ Based on KeyNote [RFC 2704]

- ◆ Fields:

- Authorizer
- Licensees
- Conditions
- Signature

Sample Rules

```
Authorizer: "POLICY"
```

```
Licensees: admin
```

```
Conditions: (app_domain == "LOC_APP")  
            -> "true";
```

```
Authorizer: admin
```

```
Licensees: joe
```

```
Conditions: (app_domain == "LOC_APP") &&  
            (evtType == "LOC_INFO") &&  
            (user == "joe") &&  
            (owner == "joe")  
            -> "true";
```

Access Control

◆ Actions

- authenticate
- advertise
- publish
- subscribe
- receive
- change policy

Restricting Delegations

- admin delegates ownership rights to joe

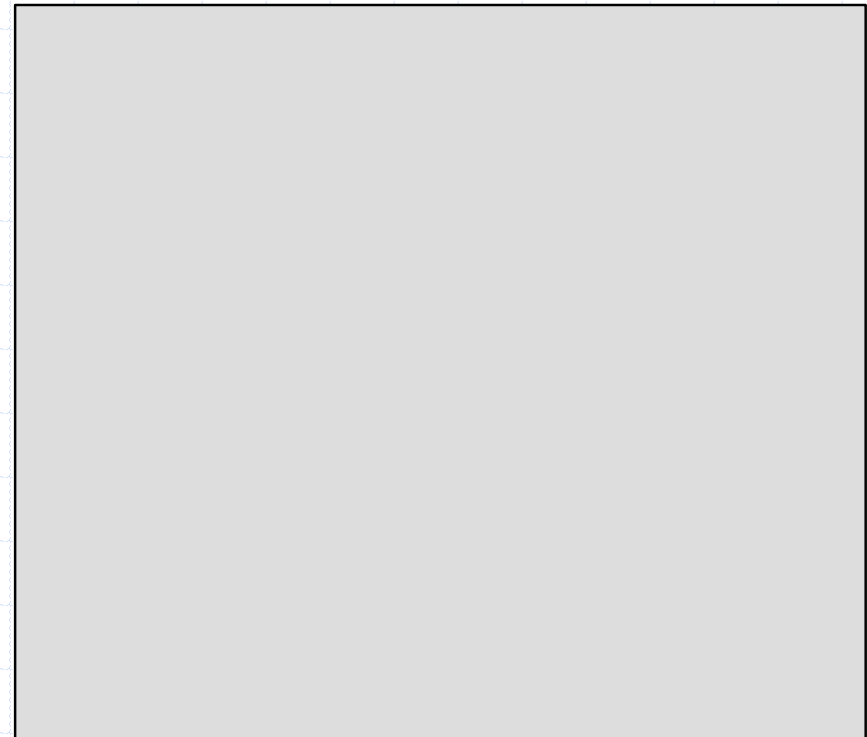


- joe delegates only "SUBSCRIBE" and "RECEIVE" rights to alice



Advertisements

- ◆ Application
- ◆ Event type
- ◆ Attribute names
- ◆ Owner
- ◆ Access control
- ◆ Data security
- ◆ Granularity



Access Control

- ◆ No-control
- ◆ Subscribe-time
 - Only check subscription requests
- ◆ Receive-time
 - Check before events are delivered
- ◆ Receive-Subscribe-time

Granularity of Security Guarantees

◆ System granularity

- ◆ confidentiality required
- ◆ no access control
- ◆ protect from system outsiders

◆ Event-type granularity

- ◆ authorization for all events of a type
- ◆ once authorized a user can read all events of that type

◆ Matching-set granularity

- ◆ determine set of interested and authorized subscribers for each event
- ◆ only subscribers from that set can gain access
- ◆ each event encrypted for a different subset of subscribers

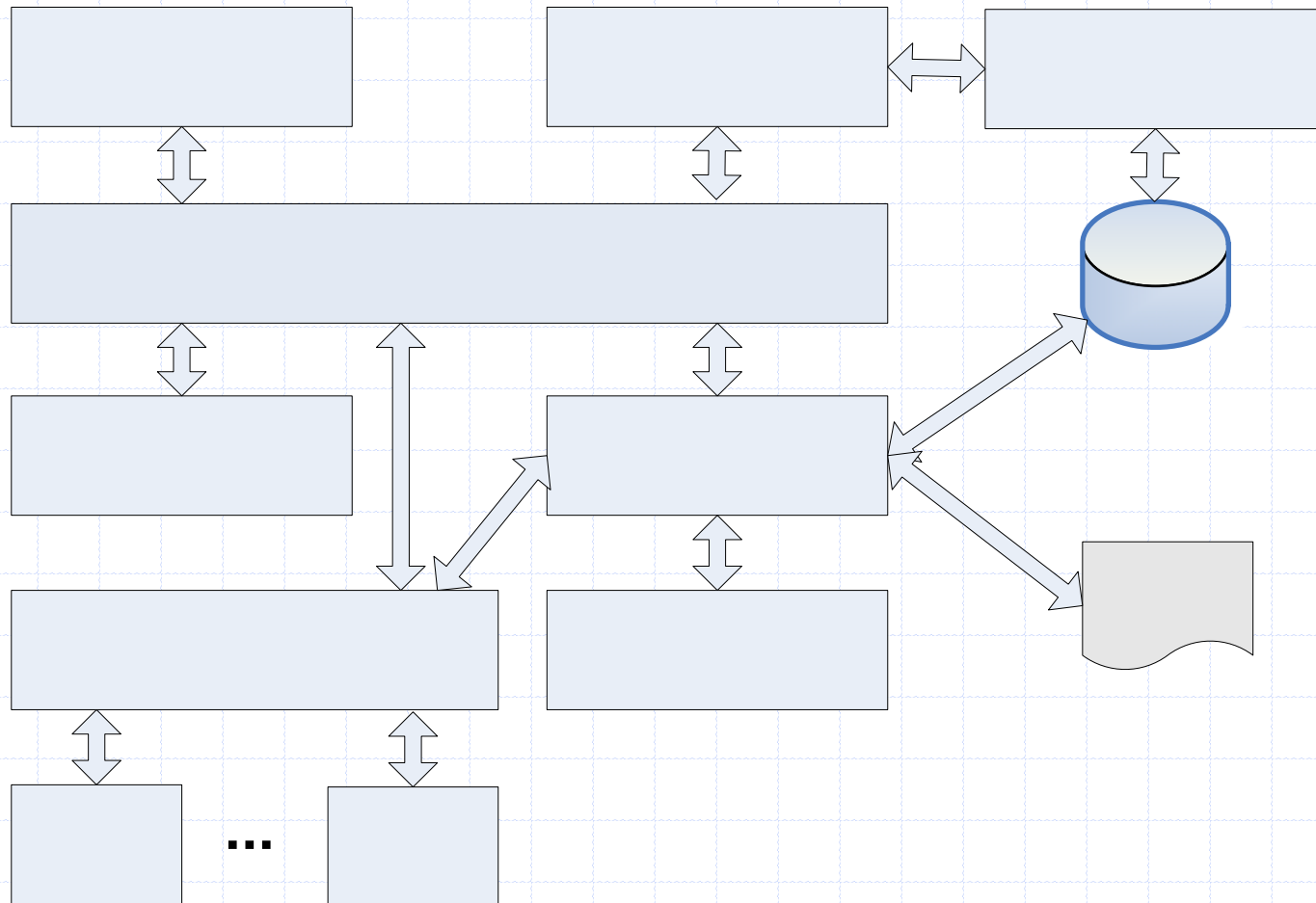
System

- ◆ Implemented in Java
- ◆ Supports any number of applications and event types
 - Advertisements read at start-up
- ◆ External attributes

- ◆ Event schema
 - List of attributes
 - All attributes - String
[LOC_INFO: (user, building, room)]

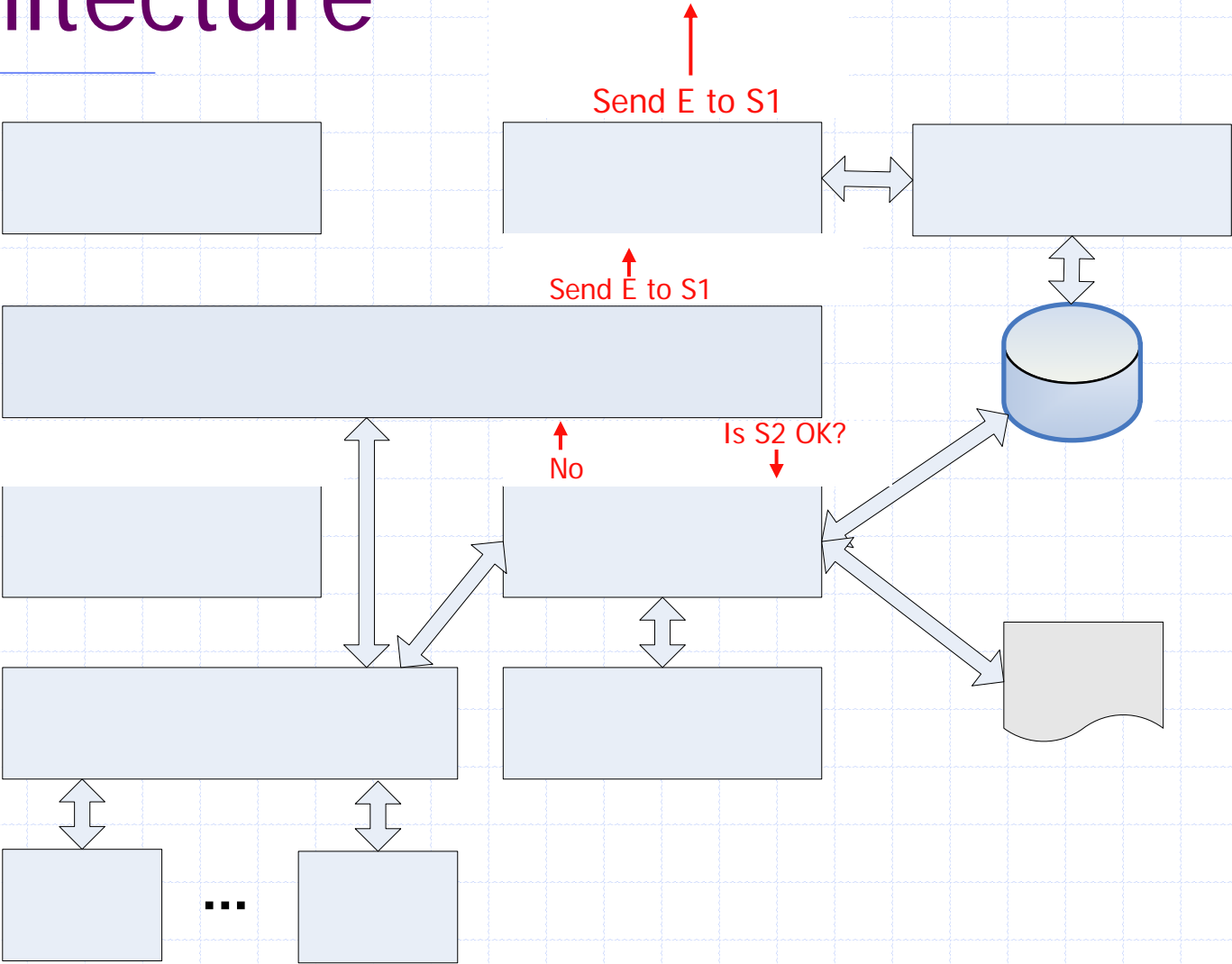
- ◆ Subscriptions
 - Only equality implemented (others trivial to add)
(user == "alice" && building == EECS && room == "*")

Architecture



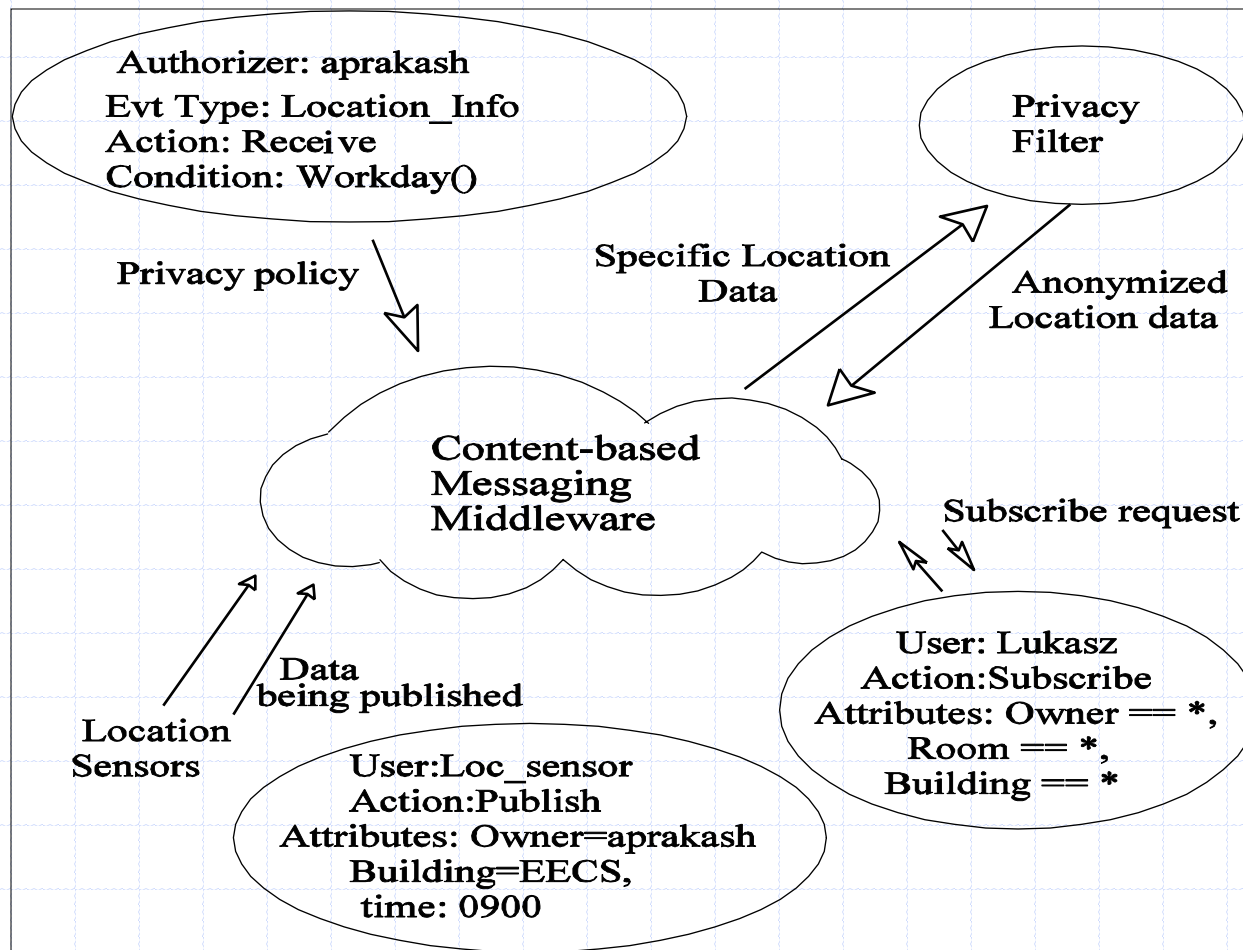
**Broker-to-Broker
Handler**

Architecture



**Broker-to-Broker
Handler**

Location-Tracking Application



Location-Tracking Application

◆ Event schema:

[LOC_INFO: (user, building, room)]

◆ Sensors

- planned - RFID
- currently – event generator

◆ Privacy policies

- users own event about them
- allow others to receive your events
- based on event attributes and external attributes

Eve authorizes everybody to receive her events but only when Eve and the subscriber are in the same room.

Authorizer: POLICY

Conditions: (app domain == "LOC_APP") && (evtType == "LOC_INFO") && (owner == "Eve") && (action == "RECEIVE") && (building == extBuilding) && (room == extRoom) -> "true";

Authorizer: PO

locat

: (ap

locat

Conditions: (ap

(acti

Authorizer: loca

Licensee: locat

Conclusion and Future Work

- ◆ Flexible support for complex privacy policies
- ◆ Services (such as privacy filters)
 - Publisher/subscriber
- ◆ Restricting delegation
- ◆ Support for contract signing
- ◆ Support for archived events

Questions?

opyrchal@muohio.edu

aprakash@eecs.umich.edu

csu2103@cse.iitd.ernet.in