

# Access Control and Mobile Agents

Guillermo Navarro

[gnavarro@ccd.uab.es](mailto:gnavarro@ccd.uab.es)

Combinatorics and Digital Communication Group (CCD)

Department of Computer Science

Universitat Autònoma de Barcelona (UAB)

*September 2003*

# Contents

- ▶ Objectives.
- ▶ Access Control.
- ▶ Trust Management and Access Control.
- ▶ Sea-of-Data applications and Mobile Agents ⇒ MARISM-A.
- ▶ Access control for MARISM-A.
- ▶ Conclusions.

# Objectives

## (I) Access Control:

- ▶ *Traditional* access control.
- ▶ Novel approaches for access control.

## (II) Mobile Agents and Sea-of-Data applications:

- ▶ Access control and mobile agents.
- ▶ Design of an access control system for MARISM-A.

# Access control definition

**Access control:** control every access to a system and ensure that all and only authorized accesses can take place.

# Mandatory Access Control

## Mandatory Access Control (**MAC**)

- ▶ Central authority determines the access control rules.
- ▶ Bell-LaPadula model (1973): Control information flow between security levels.

# Mandatory Access Control

## Mandatory Access Control (**MAC**)

- ▶ Central authority determines the access control rules.
- ▶ Bell-LaPadula model (1973): Control information flow between security levels.
  
- ▶ Applicable in military-like environments.

# Discretionary Access Control

## Discretionary Access Control (**DAC**)

- ▶ The owner of a resources determines the access control rules of the resource.
- ▶ Access Matrix (Lampson 1974)  $\implies$  Access Control List (ACL).

# Discretionary Access Control

## Discretionary Access Control (**DAC**)

- ▶ The owner of a resources determines the access control rules of the resource.
- ▶ Access Matrix (Lampson 1974)  $\implies$  Access Control List (ACL).
- ▶ DAC-based policies currently the most used in corporate environments.

# Role-based Access Control (RBAC)

*Ferraiolo and Kuhn (NIST 1992)*

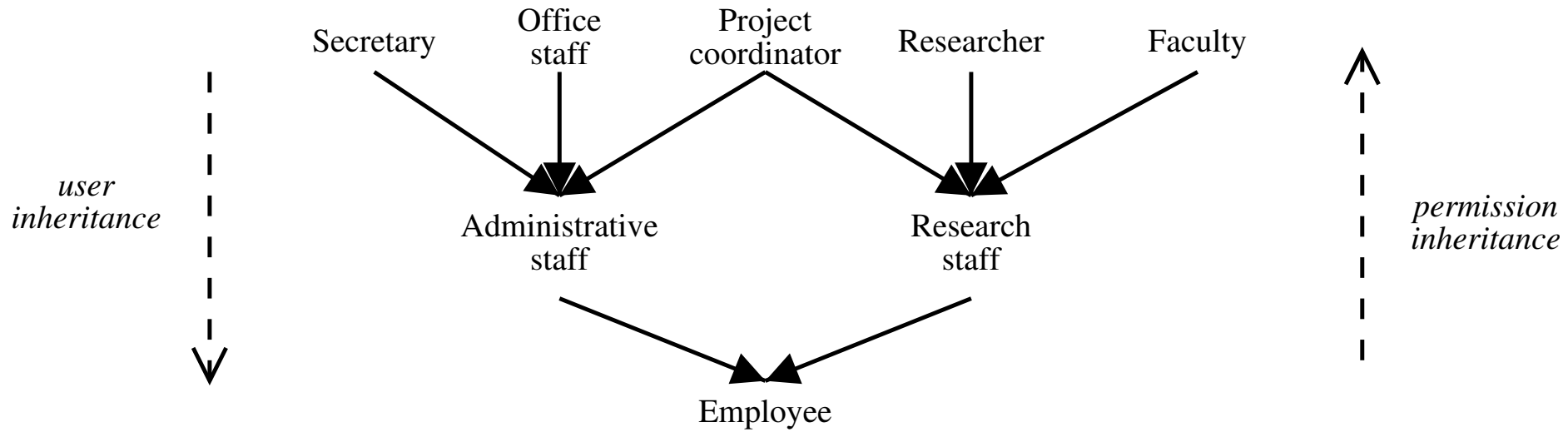
- ▶ **Users** associated with **roles**.
- ▶ **Roles** associated with **permissions**.

# Role-based Access Control (RBAC)

*Ferraiolo and Kuhn (NIST 1992)*

- ▶ **Users** associated with **roles**.
  - ▶ **Roles** associated with **permissions**.
- 
- ▶ **Role hierarchies**.
  - ▶ Easy administration of access control.
  - ▶ Reduces cost and errors of administration.
  - ▶ RBAC proposed as a NIST standard (2003).

# RBAC Hierarchies example



# Current efforts in Access Control

Current research focused on the standardization of access control in distributed systems.

- ▶ **Trust Management** (KeyNote, SPKI/SDSI).
- ▶ X.509 Attribute Certificates.
- ▶ Secure Assertion Markup Language (SAML).
- ▶ eXtensible Access Control Markup Language (XACML).
- ▶ eXtensible rights Markup Language (XrML).
- ▶ ... *and many more* ...

# Trust Management definition

“... unified approach for specifying and interpreting security policies, credentials, and relationships.” (*Blaze et al. 1996*)

# Trust Management definition

“... unified approach for specifying and interpreting security policies, credentials, and relationships.” (*Blaze et al. 1996*)

Trust management system:

- ▶ **Actions:** operations with security consequences.

# Trust Management definition

“... unified approach for specifying and interpreting security policies, credentials, and relationships.” (*Blaze et al. 1996*)

Trust management system:

- ▶ **Actions**
- ▶ **Principals**: entities that can be authorized to request actions.

# Trust Management definition

“... unified approach for specifying and interpreting security policies, credentials, and relationships.” (*Blaze et al. 1996*)

Trust management system:

- ▶ **Actions**
- ▶ **Principals**
- ▶ **Policies**: govern the actions that principals are authorized for.

# Trust Management definition

“... unified approach for specifying and interpreting security policies, credentials, and relationships.” (*Blaze et al. 1996*)

Trust management system:

- ▶ **Actions**
- ▶ **Principals**
- ▶ **Policies**
- ▶ **Credentials**: allow principals to delegate authorizations.

# Trust Management definition

“... unified approach for specifying and interpreting security policies, credentials, and relationships.” (*Blaze et al. 1996*)

Trust management system:

- ▶ **Actions**
- ▶ **Principals**
- ▶ **Policies**
- ▶ **Credentials**
- ▶ **Compliance Checker:** service that determines whether a requested action should be allowed, based on policy and a set of credentials.

# Trust Management definition

“... unified approach for specifying and interpreting security policies, credentials, and relationships.” (*Blaze et al. 1996*)

Trust management system:

- ▶ **Actions**
- ▶ **Principals**
- ▶ **Policies**
- ▶ **Credentials**
- ▶ **Compliance Checker**

PolicyMaker, KeyNote, SPKI/SDSI\*, ...

# Trust Management for Access Control

Main goal, answer the question:

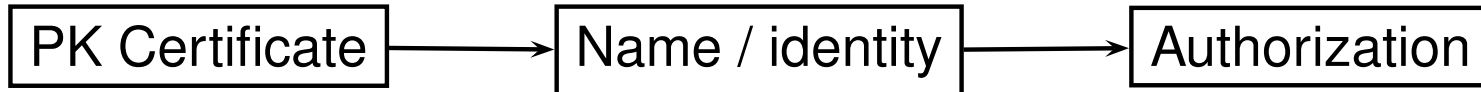
Does the set  $C$  of **credentials** prove that the **requester  $r$  complies** with the security **policy  $P$** ?

# Trust Management for Access Control

Main goal, answer the question:

Does the set  $C$  of **credentials** prove that the **requester  $r$  complies** with the security **policy  $P$** ?

Traditional approach: identity based PKI (*the big lie*)

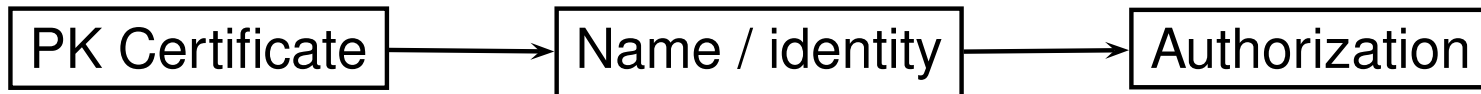


# Trust Management for Access Control

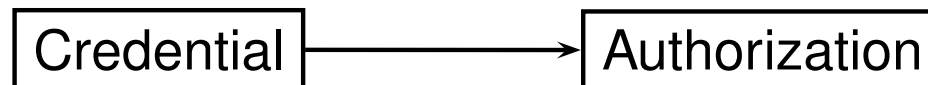
Main goal, answer the question:

Does the set  $C$  of **credentials** prove that the **requester  $r$  complies** with the security **policy  $P$** ?

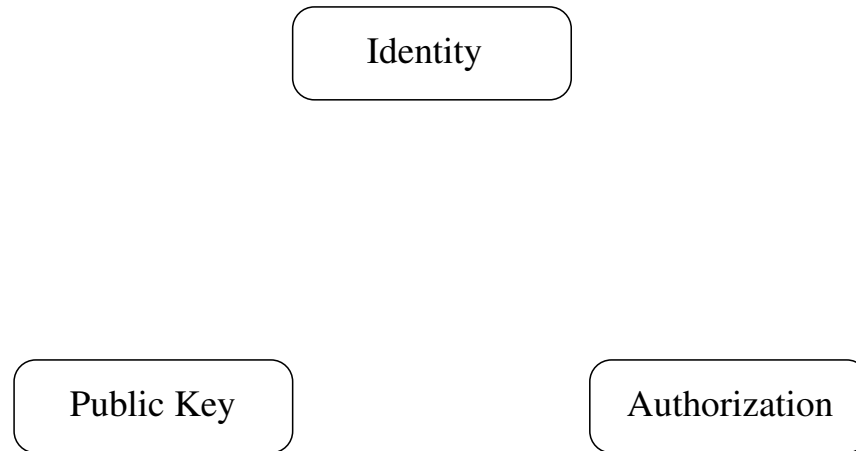
Traditional approach: identity based PKI (*the big lie*)



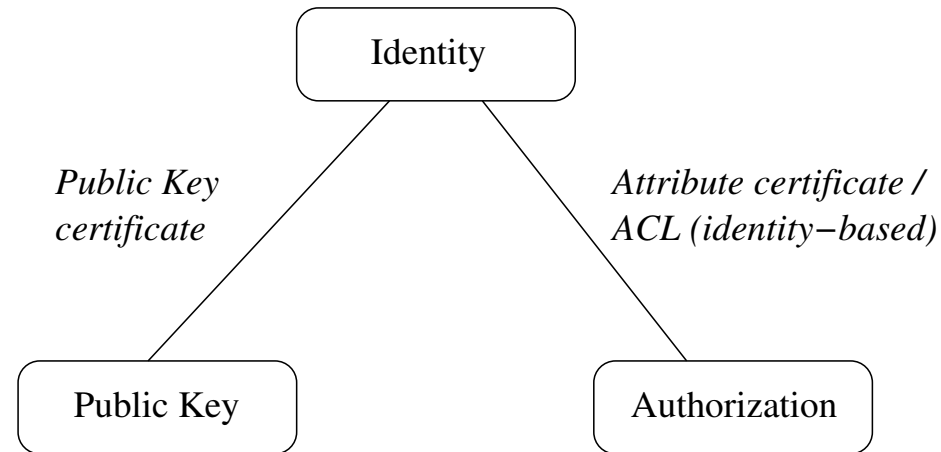
Trust management approach



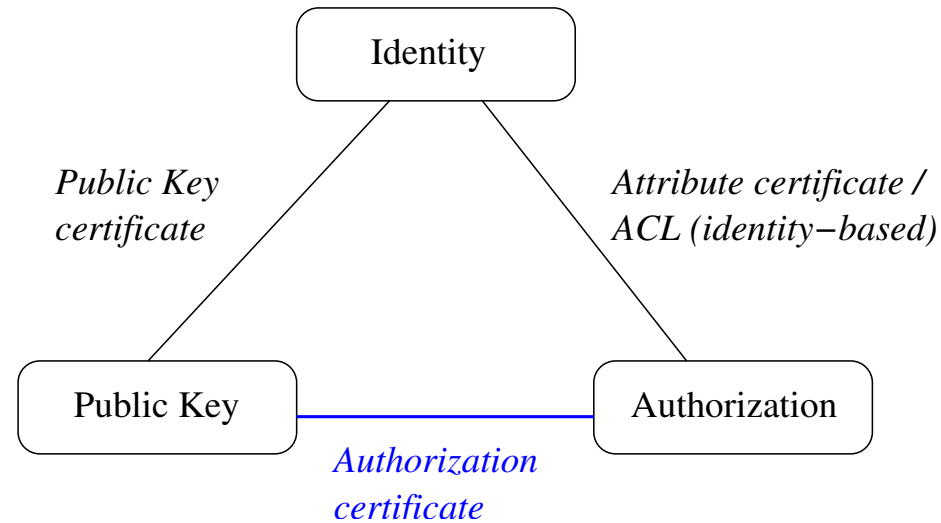
# Authorizations, identities, and public keys



# Authorizations, identities, and public keys



# Authorizations, identities, and public keys



# Delegation of authorization (or trust) for AC

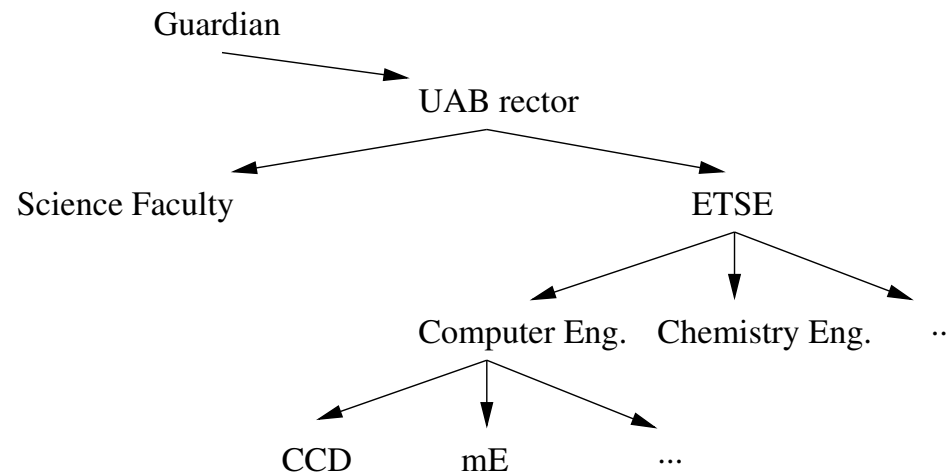
⇒ **Decentralized access control management.**

- ▶ Benefits: flexible, natural and intuitive way to distribute management.

# Delegation of authorization (or trust) for AC

⇒ **Decentralized access control management.**

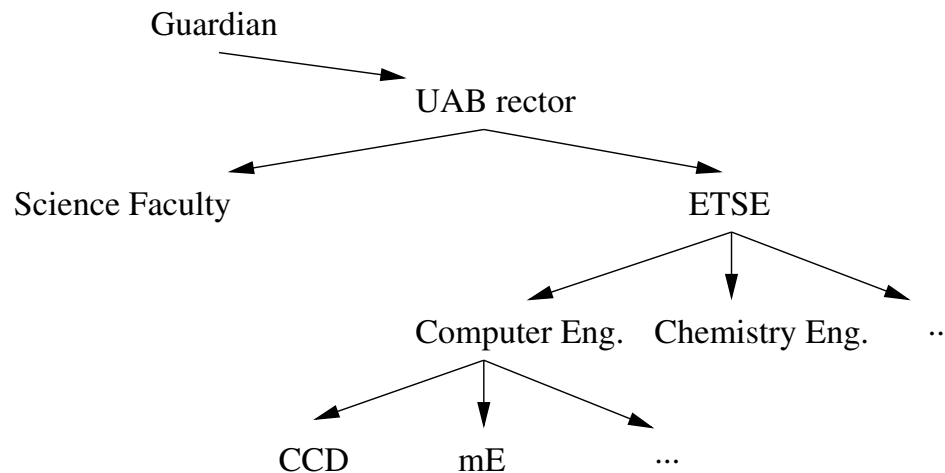
- ▶ Benefits: flexible, natural and intuitive way to distribute management.



# Delegation of authorization (or trust) for AC

⇒ **Decentralized access control management.**

- ▶ Benefits: flexible, natural and intuitive way to distribute management.



- ▶ Drawbacks: increases the complexity of the resolution algorithms.

# SPKI/SDSI vs. KeyNote

- ▶ Both present the same approach, and similar characteristics.

# SPKI/SDSI vs. KeyNote

- ▶ Both present the same approach, and similar characteristics.
- ▶ KeyNote main limitation: **monotonic principle.**  
(SPKI/SDSI supports revocation of certificates)

# SPKI/SDSI vs. KeyNote

- ▶ Both present the same approach, and similar characteristics.
- ▶ KeyNote main limitation: **monotonic principle.**  
(SPKI/SDSI supports revocation of certificates)
- ▶ SPKI/SDSI advantage: **local name system.**  
powerful mechanism for managing groups (o roles) in a distributed environment.

# Sea-of-Data (SoD) Applications

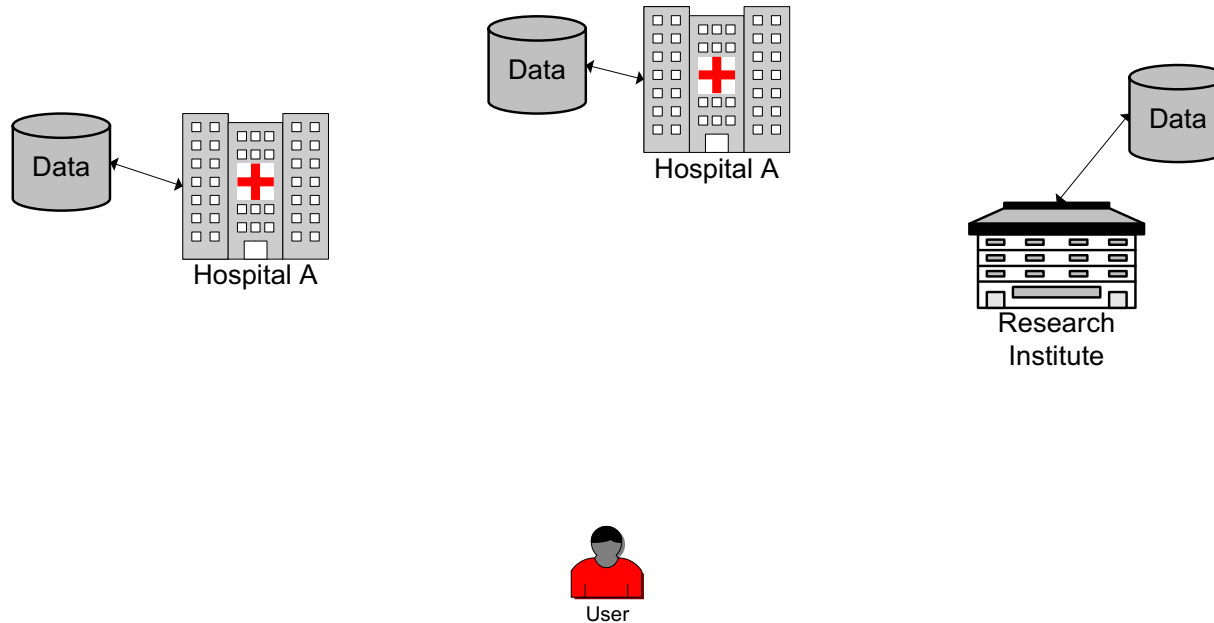
Processing huge quantities of distributed data.

- ▶ e.g. medical image processing:

# Sea-of-Data (SoD) Applications

Processing huge quantities of distributed data.

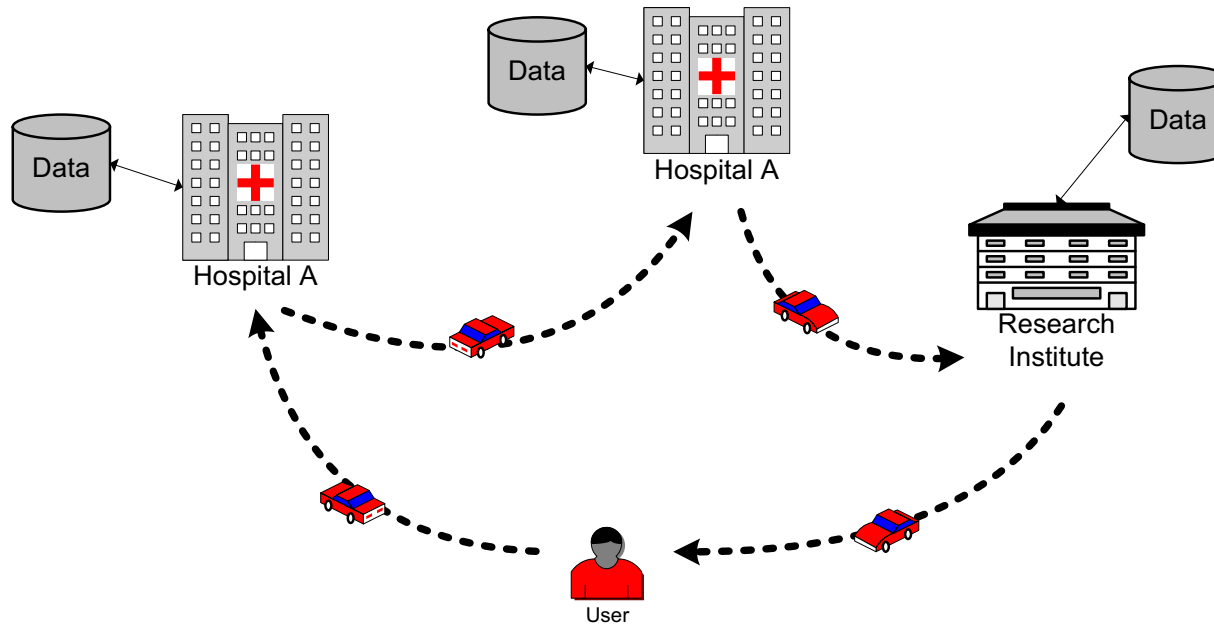
- ▶ e.g. medical image processing:



# Sea-of-Data (SoD) Applications

Processing huge quantities of distributed data.

- ▶ e.g. medical image processing:



# SoD applications and mobile agents

What MA provide SoD applications:

- ▶ **the code is executed where data is located,**
- ▶ initial launching platform (user) does not need to be on-line,
- ▶ parallelization of the execution.

# SoD applications and mobile agents

What MA provide SoD applications:

- ▶ **the code is executed where data is located,**
- ▶ initial launching platform (user) does not need to be on-line,
- ▶ parallelization of the execution.

While MA appear to be the most feasible solution for SoD applications, they introduce important security problems.

SoD applications require a secure **resource access control**.

## An **A**rchitecture for **M**obile **A**gents with **R**ecursive Itineraries and **S**ecure **M**igration

- ▶ Secure mobile agent platform based on JADE.
- ▶ FIPA compliant.
- ▶ Itinerary and code protection.
- ▶ Support for several agent types and architectures:  
static or mobile, encrypted or plain, explicit or nested itinerary, . . .
- ▶ PKI for general security services: communication  
between agencies, . . .
- ▶ . . .

# RBAC + SPKI/SDSI for MARISM-A

- ▶ Requirements and limitations:
  - ▷ Mobile agents cannot store private keys and perform cryptographic operations.
  - ▷ Agencies registered in a CA.
  - ▷ Users (clients) not registered in the CA.
    - ⇒ each user is an SPKI/SDSI principal

# RBAC + SPKI/SDSI for MARISM-A

- ▶ Requirements and limitations:
  - ▷ Mobile agents cannot store private keys and perform cryptographic operations.
  - ▷ Agencies registered in a CA.
  - ▷ Users (clients) not registered in the CA.
    - ⇒ each user is an SPKI/SDSI principal
- ▶ Roles implemented using SPKI/SDSI local names.
- ▶ Access control framework divided in 4 modules, each one implemented as a static agent.
- ▶ Communication: FIPA ACL.

# Access Control Modules

- ▶ Role Manager (**RM**)

users → roles

# Access Control Modules

- ▶ Role Manager (**RM**)  
users → roles
- ▶ Authorization Manager (**AM**)  
authorizations → roles

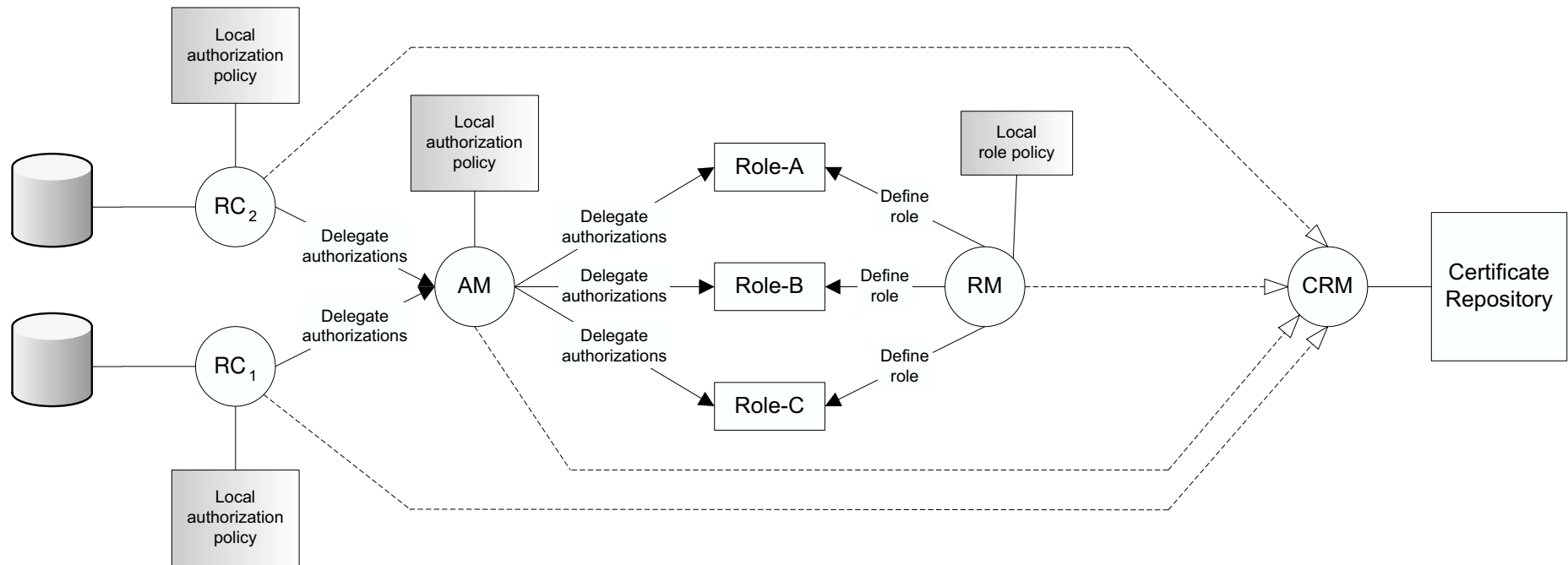
# Access Control Modules

- ▶ Role Manager (**RM**)  
users → roles
- ▶ Authorization Manager (**AM**)  
authorizations → roles
- ▶ Certificate Repository Manager (**CRM**)  
**manage** an SPKI/SDSI certificate repository

# Access Control Modules

- ▶ Role Manager (**RM**)  
users → roles
- ▶ Authorization Manager (**AM**)  
authorizations → roles
- ▶ Certificate Repository Manager (**CRM**)  
**manage** an SPKI/SDSI certificate repository
- ▶ Resource Controller (**RC**)  
access request verifier

# Access Control modules explained



# Establishing mobile agents role membership

- ▶ A **hash** of the agent code as the subject of an SPKI/SDSI certificate.

# Establishing mobile agents role membership

- ▶ A **hash** of the agent code as the subject of an SPKI/SDSI certificate.

How:

# Establishing mobile agents role membership

- ▶ A **hash** of the agent code as the subject of an SPKI/SDSI certificate.

How:

1. **User-managed role:** trusted user

# Establishing mobile agents role membership

- ▶ A **hash** of the agent code as the subject of an SPKI/SDSI certificate.

How:

1. **User–managed role**: trusted user
2. **RM–managed role**: not so trusted user

# Considerations of the access control system

- ▶ Flexible, scalable, lightweight, ...

# Considerations of the access control system

- ▶ Flexible, scalable, lightweight, . . .
- ▶ Can be used in all FIPA compliant environments.

# Considerations of the access control system

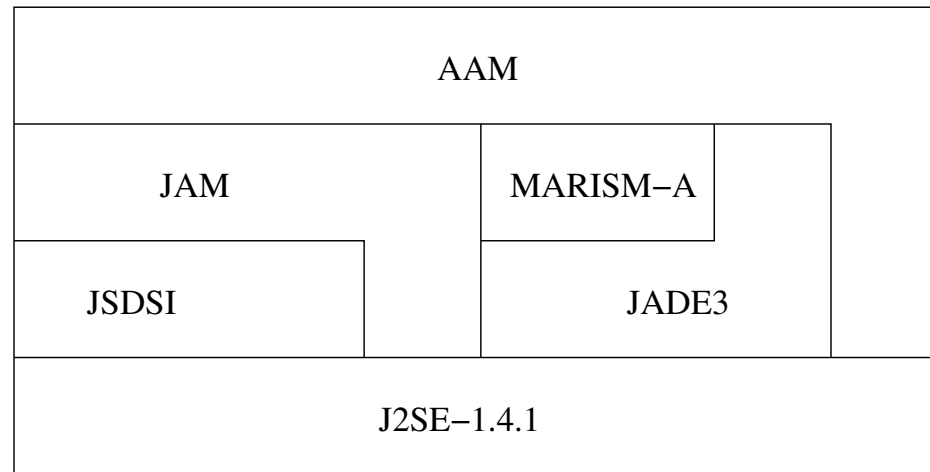
- ▶ Flexible, scalable, lightweight, ...
- ▶ Can be used in all FIPA compliant environments.
- ▶ Platform independent (dependent on the resource!).

# Considerations of the access control system

- ▶ Flexible, scalable, lightweight, ...
- ▶ Can be used in all FIPA compliant environments.
- ▶ Platform independent (dependent on the resource!).
- ▶ Distribution:
  - ▷ management (RM and AM)  $\implies$  OK,
  - ▷ verifier (RC)  $\implies$  OK,
  - ▷ repository (CRM)  $\implies$  possible but complex.

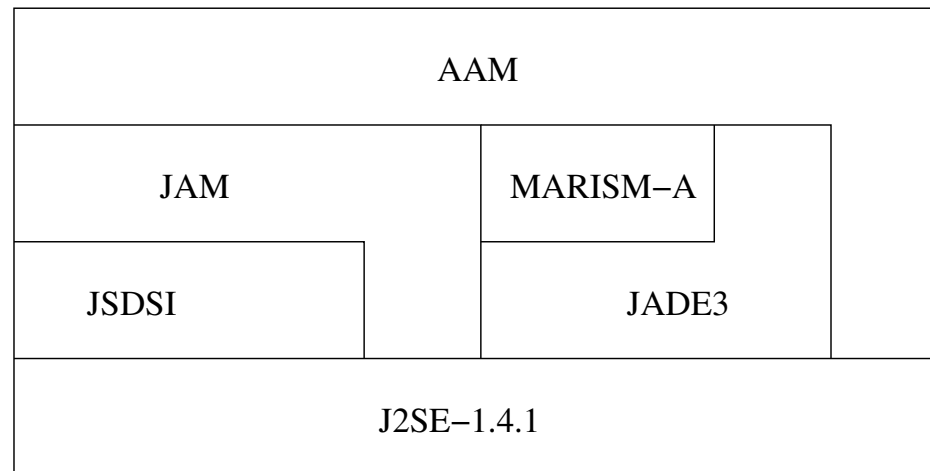
# Implementation

## Authorization Framework for MARISM-A



# Implementation

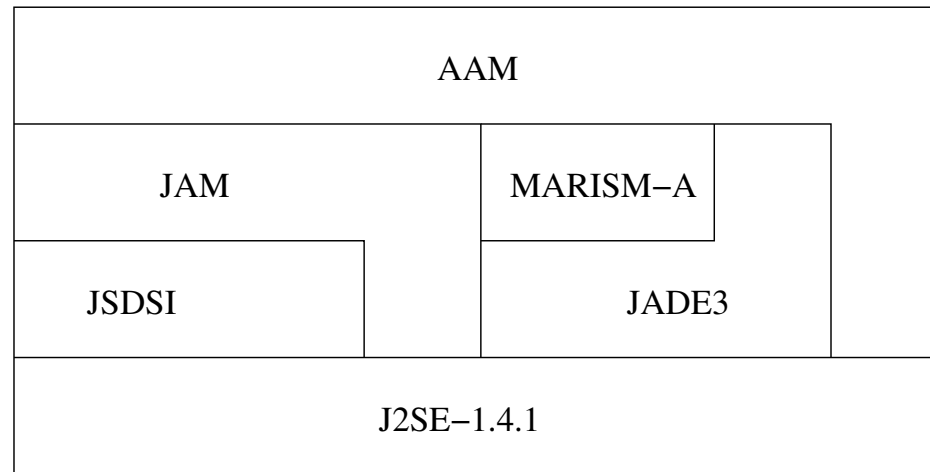
## Authorization Framework for MARISM-A



- ▶ **JSDSI**: Java implementation of the SPKI/SDSI.

# Implementation

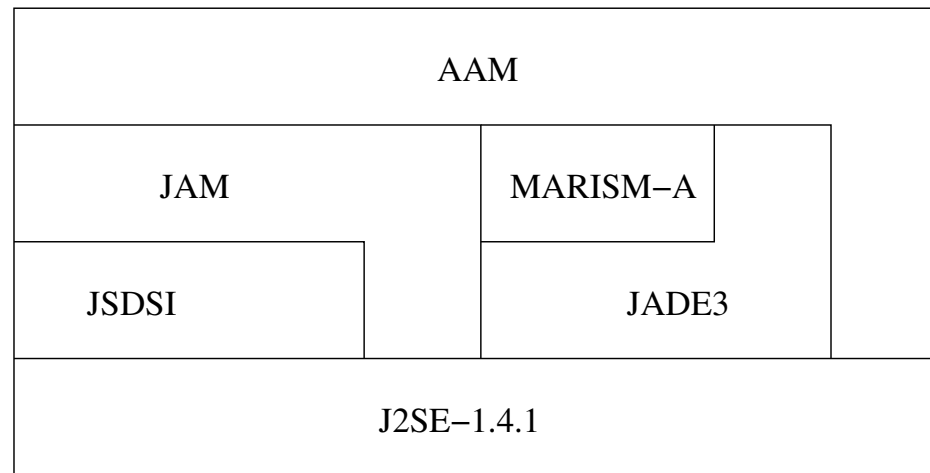
## Authorization Framework for MARISM-A



- ▶ **JSDSI**: Java implementation of the SPKI/SDSI.
- ▶ **JAM**: Java **A**uthorization API for **M**ARISM-A.

# Implementation

## Authorization Framework for MARISM-A



- ▶ **JSDSI**: Java implementation of the SPKI/SDSI.
- ▶ **JAM**: Java **A**uthorization API for **MARISM-A**.
- ▶ **AAM**: **A**gent-based **A**uthorization system for **MARISM-A**.

# Implementation considerations

- ▶ Early prototype.
- ▶ Main problem: JSDSI! (lack of good Java implementations for trust management).
- ▶ Supporting tools (mainly GUI) for the end-user.
- ▶ Important issue not discussed here: how to actually access the resource.
- ▶ IOHO novel and interesting solution.

# Conclusions

- ▶ Study of access control theory, and current approaches for distributed systems.  
RBAC + Trust Management (SPKI/SDSI).

# Conclusions

- ▶ Study of access control theory, and current approaches for distributed systems.  
RBAC + Trust Management (SPKI/SDSI).
- ▶ Access control for SoD and Mobile Agents.  
Design and prototype implementation for MARISM-A

# Future research lines

- ▶ **Delegation.**

# Future research lines

- ▶ **Delegation.**
- ▶ **Trust Negotiation.**

# Future research lines

- ▶ **Delegation.**
- ▶ **Trust Negotiation.**
- ▶ **Certificate discovery (X.509, SAML, ...), and Policy Distribution.**

# Future research lines

- ▶ **Delegation.**
- ▶ **Trust Negotiation.**
- ▶ **Certificate discovery** (X.509, SAML, . . . ), and **Policy Distribution.**
- ▶ **Authorization** (or Trust) **Management** for highly distributed systems (Grid, P2P).

# Access Control and Mobile Agents

Guillermo Navarro

[gnavarro@ccd.uab.es](mailto:gnavarro@ccd.uab.es)

Combinatorics and Digital Communication Group (CCD)

Department of Computer Science

Universitat Autònoma de Barcelona (UAB)

*September 2003*