Modeling Trust in Distributed Systems

By,
Lalana Kagal, Scott Cost,
Timothy Finin, Yun Peng

University of Maryland Baltimore County

Presented at the Second Workshop on Norms and Institutions in MAS, Autonomous Agents, Montreal, May 29, 2001

Outline

- ✓ Problems
- ✓ What is Distributed Trust
- ✓ Background
- ✓ Limitations
- ✓ Overview of our system
- ✓ Design
- ✓ Ontology
- ✓ How it works
- ✓ Processing
- ✓ Ongoing Work
- ✓ Summary

Problem

There are 2 scenarios that we are trying to address

1. Supply Chain Management System
   - ✓ Focus of the paper
2. Dynamic Wireless Environment
   - ✓ Ongoing work

Supply Chain Management

- ✓ Inter company information access
- ✓ Sharing/accessing information, and performing actions across (or within) organizations
- ✓ have to observe organizational policies for security and authorization.

Supply Chain Management

- ✓ We need to:
  - grant authorization and rights
  - delegate authorization and rights
  - request certificates proving authorizations
  - request actions, attaching certificates as necessary
- ✓ Implemented a system for CIIMPLEX EECOMS project

CIIMPLEX EECOMS

CIIMPLEX: Consortium for Integrated Intelligent Manufacturing Planning and Execution

EECOMS : Extended Enterprise Coalition for Integrated Collaborative Manufacturing Systems

Funder: National Institute of Standards and Technology / Advanced Technology Program Technologies for the Integration of Manufacturing Applications (TIMA)

Goal: Plug and Play framework of business objectives and integration-enabling tools allowing a suite of solutions that can be implemented "out-of-the-box" at small and midsized manufacturing and process sites including MES, ERP, Finite Scheduling, and Capacity Analysis/Decision Support

Objectives: interoperability, configurability, adaptability, extensibility, plug and play.

Focus: supply chain management

CIIMPLEX Participants

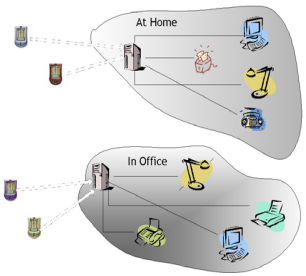
- ✓ IBM Corp
- ✓ University of Maryland Baltimore County
- ✓ University of Florida
- ✓ University of North Carolina at Charlotte
- ✓ Berclain USA Ltd.
- ✓ Boeing

CIIMPLEX

- ✓ QAD Inc
- ✓ GSE Systems
- ✓ Lucent Technologies
- ✓ Ingersoll-Rand Co.
  - Demand Solutions
  - DLoG Remex Inc.
- ✓ Intercim
- ✓ EnvisionIt Software
- ✓ The Haley Corporation

Dynamic Wireless Environments

- ✓ Unknown entities
- ✓ Wireless devices are resource poor
- ✓ Authenticate other wireless devices
- ✓ Need to communicate and sometimes use other devices



Dynamic Wireless Environments

- We need to
  - specify 'public' policies
  - Manage authorization and delegation
  - Implementation should be 'light', yet effective

Distributed Trust

- **Issues**
  - No central authority
  - logging in is not possible
  - Access control for entities never encountered before
- We use Distributed Trust to solve the above issues
- **Distributed Trust**
  - trust = policies + credentials + determining if credentials fulfill policies + delegating trust to third parties

Background

- PGP
  - Secure emails
  - web of trust
- SPKI
  - used for access control
  - A PKC consists of key,name, authorization
  - includes notion of delegation + permission to delegate further
  - authorization certificates
  - Depth of delegation is boolean or integer

Background (cont.)

- Role based Access Control
- Trust Establishment
  - based on role based Access Control
  - Policy Language is defined in XML
  - Certificate Collector collects missing certificates
  - supports negative certificates
- PolicyMaker
  - binds public keys to predicates
  - no mapping between keys and personal id
  - simple language to express trust info
  - Given policy, answers queries about trust

Background (cont.)

- ✓ Delegation Logic
  - Language for specifying trust information
  - ability to manage non-monotonic reasoning
  - expresses delegation depth explicitly

Limitations

- ✓ PGP and X.509 certificates
  - provide authorization
  - no delegation
- ✓ SPKI
  - provides authorization
  - limited delegation
  - no constraints on delegations
- ✓ Role based
  - Difficult in inter company communication as roles are different across domains

Limitations

- ✓ Trust Establishment
  - specifies certificate types that makes interoperability with other TE systems difficult
  - used for mapping between entities and roles
  - limited support for delegation
- ✓ PolicyMaker
  - Policy is complex, fully programmable
  - Hard to understand for non programmers

Overview of our System

- ✓ provides authorization
- ✓ representation for credentials and trust relationships
- ✓ flexibility in describing policies
- ✓ constraints on execution
- ✓ constraints on further delegation

Overview of our System

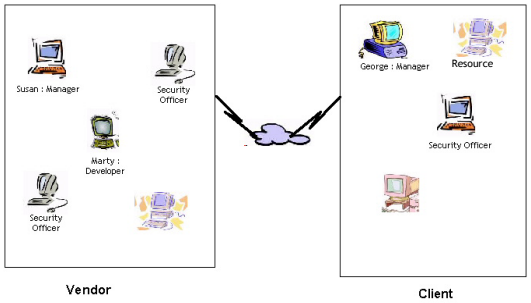
- ✓ Security and authorization policy represented as rules and constraints
- ✓ Principals make signed statements
- ✓ Principals can be humans (via a suitable interface) or software agents acting on the behalf of humans
- ✓ Agents reason about the policy and statements to derive (prove) authorizations

Design

Supply Chain Management System

- ✓ Companies have security policies
- ✓ Policy enforced by a number of 'security officers'
- ✓ Each agent in the system has an ID certificate, X.509
- ✓ All communication via signed messages
- ✓ Trust info encoded in Prolog

Design



Ontology

- ✓ Agents
  Entities in the system
  Associated with roles
- ✓ Actions
  application specific action
- ✓ Propositions
  - Permissions
    Actions that an agent can perform
    permission(Agent,Action,Constraint)

Ontology

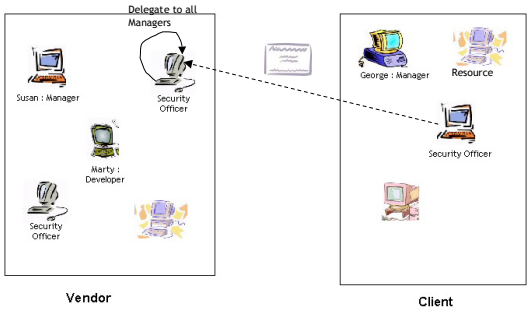
- Delegations
  Abilities that are deferred
  delegate(Issue,Start,End,From,To,Permission,
  Constraint on redelegation, Flag)
  Delegation is an ability

✓ requests
  request(From,To,Action)
  requestCert(From,To,Action)

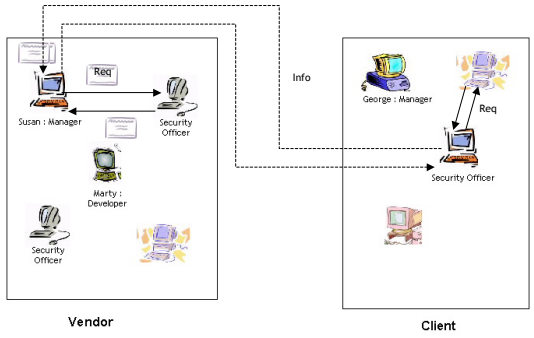
Policy

✓ Security Policy
- Authorization policies
  - Specifies rules for checking credentials
- Delegation policies
  - Rules for deferring of permissions
- Basic Permissions
  - Role based access rights for entities
  - Access rights for an agent
  - Certain basic rights

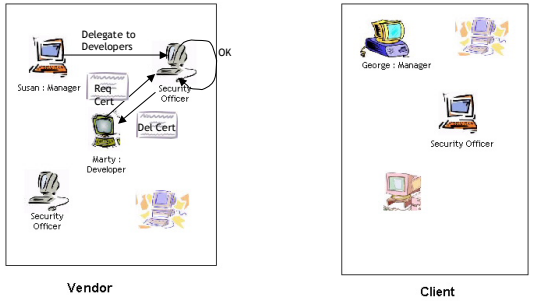
How it works : Initial delegation



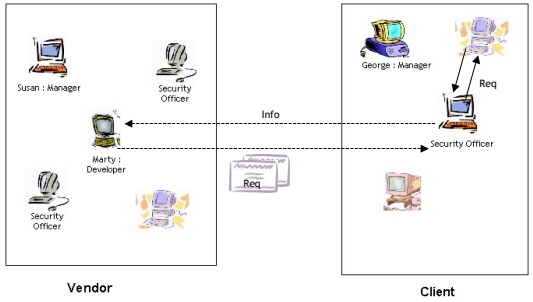
How it works : Request



How it works : Delegation



How it works : Request



Processing

- ✓ Prolog is used to specify policies, delegations and queries
- ✓ An action is allowed if
  - the agent has the ability to perform the action or
  - has been delegated the ability by someone, who has the ability to delegate

Processing

- ✓ An agent has the right to delegate if
  - it is an ability
  - or someone had delegated to it the right and the delegator had the *right to delegate*

Ongoing Work

- ✓ Use a semantic language based on XML (RDF/DAML) for representation of trust information
- ✓ Use XML Signatures
- ✓ Trust in dynamic wireless environment
- ✓ Specifying ontology for permissions, obligations, entitlements, prohibitions in RDF/DAML

Ongoing Work

- ✓ Reputation mechanism
- ✓ Using smart cards for authentication
- ✓ Delegating of obligations, entitlements, prohibitions
- ✓ Short lived Propositions
  - reduces processing time
  - easy handling of revocations

Summary

- ✓ We have developed an infrastructure for distributed trust
- ✓ Designed a representation for trust info, credentials and policies
- ✓ Showed its feasibility through implementation
- ✓ Discussed some of our future research directions

Questions

?