Policy Based Framework for Trust Management and Evolution of Peer to Peer Groups.

Madhumita Chatterjee
(04429802)

Advisors:
Prof M. Bernard &
Prof G. Sivakumar.

1/16/2008
Outlay of Presentation

- Context – Motivation
- Status of Work
- Future work plan

- Trust Metrics
- Meta Policies and Context Policies
- Simulation

6 slides
Context

- Increase of web-based collaborative applications
- Popularity of peer to peer groups
- E.g. File sharing Napster, Gnutella, Kaaza
- Diverse security requirements.
- Need for admission control, access control, key management.
Cont...

- Groups can have multiple levels
- Access control policy needs to be dynamic - multi-level
- Secure protocols for join, leave, level elevation, integrated with key management.
Motivation

- Evolving self-organising groups like wikipedia, P2P collaborations like f/oss, & project discussion archives motivates new security requirements.
- Quantum of functionality to be assigned to peers.
  - Overhead of all peers storing certificates.
  - Functionality of peers needs to be dynamic
Cont....

- Need for multiple levels and dynamic access from one level to another
- In dynamic P2P communities peers are unknown to each other and uncertain about each other's reputation
  - necessary to develop strategies for establishing trust among peers.
Trust Engine

Self Trust
Feedback by Direct interaction
Indirect Recommendation
Incentives for rating

Weighted Context Specific Parameter
Peer Credibility

Trust Computation

Context based policy

E-commerce Domain
Policy 1
Open Source Software Development
Policy 2
Online gaming
Policy 3

Global meta polices for multilevel access control

P2P Groups Integrated Framework
Authentication Access Control Key Mgmt.

Evolution of P2P Groups
Dynamic Policies
• Adaptive Trust
• Malicious Peers

Peersim Simulator

Role 1
Role 2
Role 3
Trust Engine

Self Trust

Feedback by Direct interaction

Indirect Recommendation

Incentives for rating

Trust Computation

Weighted Context Specific Parameter

Peer Credibility

Context based policy

E-commerce Domain

Open Source Software Development

Online gaming

Policy 1

Policy 2

Policy 3
Roles and Levels

- Peer roles......
  - Member Peer
  - Admission Control Peer
  - Maximal Role Peer.

- Group Levels
  - Hierarchical levels
Global meta-level policy

- Policy initiated by the creator of the group
- Rules like
  - For a new peer to join at least 50% of existing peers must vote positive.
  - To update level a peer and should be an authenticated member at the lower level and must have an accumulated trust value >x.
  - For a new policy to be framed at least 75% of existing peers with trust value >some threshold x must collaborate.
Cont.....

- The group policy can also dynamically prioritize requests for join.
- Low level requests for join could be postponed in a group which already has a large number of members performing the lower roles.
- Peers at the highest level can introduce a new level into the group if the situation so demands.
  - For a new level to be introduced at least $y$ no. of peers must be present in the current highest level and all 100% should collaborate.
Context based policy

• Rules for
  ■ Number of member level peers to be permitted to complete a specific task
  ■ Number of maximal role peers required
    ■ E.g policy rules for accepting a critical software in an open source software development project
  ■ Dynamic policies need to be framed based on current context and behavior of peers
    ■ E.g University exam policy to decide in number of grace marks to be given because of unanticipated issues.
Examples

- In a workflow management system resources for a particular activity may only be available when pre-requisite tasks have been completed
  - Access rights to individuals may need to be dynamically adjusted to allow/disallow access to restricted/authorized resources

- Military Operation System
  - Access control policies to define privilege of each user could be in accordance with overall mission and doctrine objectives which could change......
Admission constraints

- For e.g a maximal role peer in a software development workflow context could have the following admission constraints:
  - \#members(Maximal_role) > 2 & < 5
  - &member(thisUser,Moderator)
  - &!member(thisUser,developer)
  - &\#members(admission_control_peer)>0
Modelling our policies

- Use X-GTRBAC Trust enhanced XML based generalised temporal role based access control model
  - XML User Sheet (XUS). Declares the users and their authorization credentials.
  - XML Role Sheet. Declares the roles, their attributes, role hierarchy.
  - XML Permission sheet (XPS). Declares the available permissions.
  - XML User-to-Role-Assignment Sheet (XURAS). Declares rules for assignment of users to roles
  - XML Permission-to-Role Assignment Sheet (XPRAS). Defines rules for assignment of permission to roles
Cont...

- Join Request or Service request is defined as a tuple
  \(<\text{role}, \text{srv}, \text{contextset}>\) where
- \(\text{Role} \in \{\text{Role Set}\}\)
- \(\text{Srv} \in \{\text{Service set}\}\)
- Context set = \{dynamic context parameters\}
Service access request

Rule 1:
If request = JOIN_REQ then
if role = member ∧ level = 1 ∧ srv = msg.receive then
if rating > 1 ∧ voting minbound > 50 then permission = "grant"

Rule 2:
If request = JOIN_REQ then
if role = member ∧ level = 2 ∧ srv = msg.receive, msg.send then
if rating < 4 then permission = "deny"
Rule3:
If request = $UPDATE\_REQ$ then
if role = admission_peer $\land$ level = 2 $\land$ srv = msg_receive then
if rating $>$ 6 then permission = "grant"

Rule4:
If request = $UPDATE\_REQ$ then
if role = maximal_peer $\land$ level = 3 $\land$ srv = msg_receive then
if rating $>$ 8 then permission = "grant"
Trust and Reputation

- Trust is the firm belief in the competence of an entity to act independently, securely and reliably within a specified context (Marsh)
- Reputation is an expectation about an individual’s behavior based on information about or observations of its past behavior
- Reputation-based trust management allows each peer to manage its own trust.
Decentralized Trust Management

- Main challenges
  - How to cope with potential malicious peers
  - How to cope with indirect trust given input of direct trust valuations
  - How to continuously revise trustworthiness of a peer at runtime
  - How can trust evaluations be done when peers have unequal roles in the group
Different Models proposed

- Complaint based Model
- Voting based reputation Model
- Voter’s co-relation model
- Ebay Trust Management system
- PeerTrust a trust model for e-commerce communities.
Incentive based Context Sensitive Trust Model

- Feedback in terms of reputation of a peer as computed by peers with whom it had direct interactions.
- Reputation computed on the basis of indirect recommendations given by unknown peers.
- Total number of transactions and cost of transaction in e-commerce scenario.
- Credibility of peer giving the feedback.
Cont...

- Context specific attributes associated with the transactions and current trust value of peer u as calculated by peer v based on these attributes
- Trust value of peer u based on past history
- Weightage to be given to recent interactions
Trust Engine

- Self Trust
- Feedback by Direct interaction
- Indirect Recommendation
- Incentives for rating

Weighted Context Specific Parameter
Peer Credibility

Trust Computation
Weights and attributes

• We assign weights with different attributes depending on the type of application. Let $x$ be the set consisting of the different attributes.

• $x = \{x_1, x_2, \ldots, x_n\}$ Relative importance assigned to each attribute can be modelled as weight $w_{x_i}$ such that $\sum w_{x_i} = 1$.

• Let $I_u$ denote the total number of transactions performed by peer $u$ and $P$ denote the set of peers.
• R\textsubscript{IU} is the global rating of peer u.
• R\textsubscript{IV} is the rating of peer v.
• T\textsubscript{uuv} is the trust value of u as computed by v
• C(u,i) is the cost associated with the \textsuperscript{i\textsubscript{th}} transaction of peer u
• F\textsubscript{u} is the total number of feedback ratings given by peer u.
Basic Trust Metric

\[ RI_u = \frac{\sum_{v \in P, v \neq u} T_{uv} \cdot RI_v}{I_u} + \sum T_{uk} \cdot RI_k \]

\[ \sum T_{uv} = \sum R_{xi} \cdot w_{xi} \]
For e-commerce domain

- attribute set \( x_i = \{P, D, Q, S, R, T\} \) where
  - \( P \rightarrow \) price of product.
  - \( D \rightarrow \) Delivery time
  - \( R \rightarrow \) Reliability of payment
  - \( S \rightarrow \) prompt shipment
  - \( Q \rightarrow \) Quality of product
  - \( T \rightarrow \) quantum of trust in unknown
Fine-tuning parameters

- To account for high value transactions

\[
RI_u = \frac{\sum_{v \in P, v \neq u} T_{uv} \cdot RI_v \cdot \alpha \cdot C(u, i)}{I_u} + \beta \cdot \sum T_{uk} \cdot RI_k
\]
F/OSS Domain

- The attribute set for this domain would consist of parameters like...
  - P → price of software.
  - D → Delivery time
  - L → Lines of Code
  - Q → Quality
Adding incentives for rating

\[ RI_u = \frac{\sum_{v \in P, v \neq u} T_{uv} \times RI_v \times \alpha \times C(u, i)}{I_u} + \beta \times \sum T_{uk} \times RI_k + \gamma \times \frac{F_u}{I_u} \]
Unsolved Issues

- Behavioral changes
- Overcoming Whitewashers
- Overcoming Colluding malicious peers
- Overcoming Sybil attack
- Reducing storage overhead and communication cost
Simulation

- **Peersim Simulator**
  - a library written in the Java language which consists of different components or classes which help in constructing and initializing the underlying network, which can handle different protocols, can control and modify the network.
General Functioning

![Peersim architecture diagram]

1/16/2008
Latency of peers joining
Varying policies in presence of malicious peers
Varying % of malicious peers for different policies
Effect of different policies on same number of malicious peers
Future Plan

- Evolution of P2P Groups
- Dynamic Policies
  - Adaptive Trust
  - Malicious Peers
- Peersim Simulator
Future Plan (A)– Evolution of P2P groups

- Testing the flexible policy framework for evolution of P2P groups in different application contexts.
  - How group dynamics (roles) affects performance and reliability
  - Evaluate meta policies to see how they enable us to capture behavior of group in different contexts.
Future Plan (B) Adaptive Trust Module

- Explore pros and cons of different trust settings in different contexts.
  - Effect of varying weights to attribute parameters.
- How can trust decisions be made adaptive to changes in trust among peers
  - Behavioral and temporal changes
  - No prior knowledge
- Explore and analyze the interplay of different policies based on adaptive trust mechanism.
Other.....

- Need to make the approach more robust against malicious peers.
- Effect of degree of maliciousness and frequency of malicious peers affecting the group
Contributions....

- “An Integrated Framework for Authentication and Access Control in Peer to Peer Groups”
Acknowledgements

- Prof G. Sivakumar
- Prof M. Bernard
References


