Trust Negotiation

22/01/2004
Daniel Olmedilla

Learning Lab Lower Saxony and University of Hanover
Outline

- Motivation
- What is Trust Negotiation
- How does it work
- Conclusions & Further Work
- References
Outline

- Introduction
- What is Trust Negotiation
- How does it work
- Conclusions & Further Work
- References
Typical client/server authentication

- A company provides access to its clients through a web site
- Each of the client has an account with a login and password associated
- Each time a client tries to access the private network he submits the login and password
- Each time a new client starts to work with the company a new account must be created
Coalition

- Several course providers decide to join in order to improve their offer and catalogue

- All of them are interconnected and they offer a web site for the coalition where a user can search for courses

- Each of the providers requires a user to have an account to consume the course

- The user must register at the web site to access the resources and submit her login and password
Buying in Internet

- Bob wants to access an electronic AI book at “E-Book Store” (a web site he found while surfing in Internet)

- Previously, E-Book requires Bob to register providing full name, age, complete address and e-mail

- Bob does not mind to give his full name and age but he does not like to provide his complete address and e-mail. However, he does not have any other option so he does it.

- E-Book sells that book so now it asks Bob to provide his credit card information. Bob would not mind to buy the book because it is not too expensive and he is really interested in reading it. However, he has never heard about E-Book so he decides to not buy it
New Network Requirements

- Traditional distributed environments
  - Providers and requesters are known each other
    - Close environments
  - Server has to trust the client
    - Unidirectional access control: take-it-or-leave-it

- WWW, P2P, GRID, etc...
  - Dynamic networks
    - Nodes are usually not known in advance
  - Trust between strangers is needed
  - Bi-directional access control required
User Requirements

- Users do not want to register at any site (tedious task)

- Users want control over what information they disclose and set levels of privacy
  - E.g. My first name has not the same level than my credit card number

- Users do not want to give irrelevant information
  - E.g. Give my e-mail if I buy a mp3 song in a music portal

- Users want assurance about what other nodes will do with their information
  - Policies can be used at the user side to give this control
Outline

- Introduction
- What is Trust Negotiation
- How does it work
- Conclusions & Further Work
- References
Trust Negotiation

- Goal → protect resources from unauthorized access
- New approach to establishing trust between strangers
  - Initial trusting among nodes is not necessary
  - No need for registration (or even registration automatically)
- Use and interchange of credentials: online analogue to the paper credentials in real life
- Negotiation according to policies
  - Access control policies can be used in both sides (requester and provider)
- Delegation
- Trust Negotiation → trust is established gradually through an iterative exchange of digital credentials
Credentials

- Describe one or more attributes of the owner asserted by the issuer.

- As credentials contain sensitive information, they are not shown until the other part demonstrates that it is qualified to have such sensitive information.

- Possibility of credentials at the attribute level (no irrelevant information is disclosed)

- Typically based on public/private key cryptography
Control Access Policies

- Protect a resource or a credential
  - A policy must be fulfilled in order to access to the credential

- Focus the negotiation on those credentials actually needed to advance the negotiation
  - Specify credentials that the other negotiation participant must provide

- Several policies can be involved during the negotiation.

- Several policies for the same resource or credential.

- Policies over policies: protection of policies.
Extra Features

- Inference engine
  - Prolog based
  - Currently Minerva
  - XSB under development

- Adding Semantics
  - RDF import facility
  - Use of metadata information into policies

- Prototype implemented in an applet
  - Browser based
  - No extra software required
Outline

- Introduction
- What is Trust Negotiation
- How does it work
- Conclusions & Further Work
- References
Example scenario

E-Learn offers free Spanish course to police officers of California

1. Alice requests to access E-Learn's Spanish course at no charge
2. E-Learn asks Alice to show a police badge issued by the California State Police to prove that she is a police officer, and her driver's license to prove that she is living in the state of California
3. Alice is willing to disclose her driver's license to anyone, so she sends it to E-Learn. However, she considers her police badge to contain sensitive information. She tells E-Learn that in order to see her police badge, E-Learn must prove that it belongs to the Better Business Bureau.
4. Fortunately, E-Learn does have a BBB membership card. The card contains no sensitive information, so E-Learn discloses it to Alice
5. Alice now believes that she can trust E-Learn and discloses her police badge to E-Learn
6. After verifying that the badge is valid and that Alice owns it and the driver's license, E-Learn gives Alice the free registration for this course
Policy Examples

E-Learn:
freeEnroll(Course, Requester) $ Requester →
  policeOfficer(Requester) @ csp $ Requester, 
  rdfType(Course, ‘http://…/elena#Course’),
  dcLanguage(Course, ‘es’),
  creditUnits(Course, X),
  X <= 1.

Alice:
policeOfficer(alice) @ csp $ Requester →
  member(Requester) @ bbb $ Requester
| signedBy [csp].
Scenario Network Diagram

0a
PeerTrust Inc. gives a signed applet to E-Learn Web

0b
PeerTrust Inc. gives an app to Alice

1a
Alice requests the Spanish course and downloads the applet

1b
Alice starts the app and requests the Spanish course

2
The applet/app loads the local policies

3
Secure communication is established

4-5
Negotiation

6
E-Learn grants Alice access to the course

7
E-Learn creates a temporary account at the corresponding LMS

8
Alice accesses the Spanish course

9
InstitutionB Security (Negotiation Server)

Distributed Repository

E-Learn Web (Web Server)

E-Learn Security (Negotiation Server)

InstitutionA Security (Negotiation Server)
Scenario Interaction Diagram

Alice

freeEnroll(spanish101,alice)

driversLicense(alice) @ caDMV ?

policeOfficer(alice) @ csp ?

member(eLearn) @ bbb ?

policeOfficer(X) @ csp <- policeOfficer(X) @ chp , < policeOfficer(alice) @ chp 

access granted

E-Learn
Outline

- Introduction
- What is Trust Negotiation
- How does it work
- Conclusions & Further Work
- References
Conclusions (I)

<table>
<thead>
<tr>
<th></th>
<th>Centralized: (LDAP, Kerberos)</th>
<th>Federated (e.g. Liberty Alliance)</th>
<th>Trust Negotiation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>Centralized</td>
<td>Decentralized</td>
<td>Decentralized</td>
</tr>
<tr>
<td>Initial trusting required</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Strategies</td>
<td>No strategy</td>
<td>Unique strategy</td>
<td>Flexible: several different strategies</td>
</tr>
<tr>
<td>User interactivity</td>
<td>No user interactivity: only server policies</td>
<td>No user interactivity: only server policies</td>
<td>Node (User and Server) policies</td>
</tr>
<tr>
<td>Registration</td>
<td>Required in each node manually</td>
<td>Required or replicated automatically</td>
<td>No required (possible automatically)</td>
</tr>
</tbody>
</table>
Conclusions (& II)

- E-commerce needs a scalable approach that allows automatic on-line pre-registration, or does away entirely away with the need of pre-registration.

- More suitable and valuable for Distributed Environments (decentralized) and even required for Open Distributed Environments

- Horn Logic (e.g. PROLOG) is ideal to create policies easily and with flexibility

- Semantic metadata can be used in policy definition

- Easy integration of different strategies (ad-hoc strategies) for negotiation
Further work

- More standard approach
  - XML Signatures
  - SAML for message interchange
  - RuleML for policy encoding

- Integrate XSB Prolog inference engine

- Loop detection during negotiation

- Extension to Semantic Web Services

- Cash of credentials
Outline

- Introduction
- What is Trust Negotiation
- How does it work
- Conclusions
- References
References

- PeerTrust project page: http://www.learninglab.de/deutsch/projekte/peertrust.html
