Delegation of Responsibility

Bruno Crispo
University of Cambridge

23 February 1999
Changing Environment

- 2-party system → 3-party system
- Public Available → Public Offered
- Computer system security → Infrastructure policy + Service provider policy + Customer policy
- Trust Assumptions
Problem

President delegates the power to sign certain documents on her behalf to her secretary.
The president announces the sacking of the secretary because of a mistake in a very important document. The secretary has signed the document in place of the president but she has not made a mistake.

How she can defend herself?
How she can build evidence to corroborate her innocence?

23/02/1999
B. Crispo: Delegation of Responsibility
Notation

- Principal: generic entity of the system.
- Grantor: principal that delegates.
- Grantee: principal that has been delegated.
- End-Point: principal where delegation is used.
- A → B → C → D —→ End-Point, B and C are intermediaries.
Delegation of Responsibility

**Delegation of Rights:** process whereby a principal authorises an agent to act on her behalf, by transferring a set of rights to the agent, possibly for a specific period of time

Grantor is trusted

**Delegation of Responsibility:** process whereby a principal authorises an agent to act on her behalf, possibly for a specific period of time, during which it is always possible to distinguish whether a particular delegated task was performed by the principal or by the agent acting on her behalf.

Grantor is not trusted
# Delegation of Responsibility (cont’d)

<table>
<thead>
<tr>
<th>Delegation of Rights</th>
<th>Delegation of Responsibility</th>
</tr>
</thead>
<tbody>
<tr>
<td>↓</td>
<td>↓</td>
</tr>
<tr>
<td>Trust</td>
<td>No Trust</td>
</tr>
<tr>
<td>↓</td>
<td>↓</td>
</tr>
<tr>
<td>shared responsibility</td>
<td>responsibility</td>
</tr>
<tr>
<td>only rights</td>
<td>and rights</td>
</tr>
<tr>
<td>↓</td>
<td>↓</td>
</tr>
<tr>
<td>no auditing</td>
<td>auditing</td>
</tr>
</tbody>
</table>

23/02/1999

B. Crispo: Delegation of Responsibility
Other Semantics of Delegation
Necessary Conditions to Delegate

• Right to Delegate.

• Freedom to choose the principal that will act as grantee.
# Taxonomy

<table>
<thead>
<tr>
<th>Type of delegation</th>
<th>Grantor before</th>
<th>Grantor after</th>
<th>Grantee before</th>
<th>Grantee after</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rights G</td>
<td>G</td>
<td>G</td>
<td>∅</td>
<td>G</td>
</tr>
<tr>
<td>Partial Right G'</td>
<td>G'</td>
<td>G'</td>
<td>G''</td>
<td>G''=G'∪G''</td>
</tr>
<tr>
<td>Responsibility R</td>
<td>R</td>
<td>∅</td>
<td>∅ or G</td>
<td>R or R + G</td>
</tr>
<tr>
<td>Responsibility and Rights R+G</td>
<td>R + G</td>
<td>∅ or G</td>
<td>∅</td>
<td>R + G</td>
</tr>
<tr>
<td>Responsibility and Partial Rights R + G'</td>
<td>R + G'</td>
<td>∅ or G'</td>
<td>G''</td>
<td>R + G'' = G'' ∪ G''</td>
</tr>
</tbody>
</table>
Capability (1)

- **Free Propagation:** capability can be freely propagated in the system through principals.

- **Free Access:** whoever possess the capability can use the rights bound to this capability.
Capability

- SCAP [Karger 1988]
- Amoeba [Mullender 1985]
- ICAP [Gong 1990]

- Limitations:
  - Extended TCB and no use of cryptography.
  - Client-Server-Client instead of Client-Client.
  - Security policy dictated and enforced by the infrastructure.
Delegation Token  (1)

[Sollins “Cascaded Authentication” IEEE SSP 1988]
[Neuman “Proxy-Based Authorization and Accounting for Distributed System” CDS 1993]

\{Grantor, Grantee , Rights, Validity, Others\}_{K_{Grantor}}

Delegation key
Delegation Token (2)

- Problems:
  - Trustworthy intermediaries.
  - Chains of delegation.
  - Grantors are trusted:
    - not to abuse their power to delegate.
    - not to abuse their knowledge of delegation keys.
  - Grantees are trusted:
    - not to abuse the delegated rights.
Untrusted Grantee

[Abadi, Burrows, Kaufman, Lampson “Authentication and Delegation with Smart-Cards” TR. 67 1992]

[Abadi, Burrows, Lamspn, Plotkin “A Calculus for Access Control in Distributed Systems” ACM ToPLaS 1993]


• Auditing: detecting grantee’s misbehaviors.

A → B

A

B

(B for A)
Principle of Consent

[Abadi, Burrows, Kaufman, Lampson “Authentication and Delegation with Smart-Cards” TR. 67 1992]
[Abadi, Burrows, Lamson, Plotkin “A Calculus for Access Control in Distributed Systems” ACM ToPLaS 1993]

- PoC: Delegated rights must always be explicitly accepted by the grantee.
- Grantor and grantee share responsibilities for the delegated rights.
Delegation of Responsibility

• Self-Authenticating Proxy
  [Low, Christianson “Self Authenticating Proxies” IEE EE 1994]
  [Low, Christianson “A Technique for Authentication, Access Control and Resource Management in Open Distributed Systems” IEE EE 1994]

• Cryptographic Solution
  [Kim et al. “Proxy Signatures Revisited” ICICS 1997]
Delegation Protocol

\[ M = \text{“G wishes to delegate to g } \Omega \text{ using } K_{G_R}^{+} \text{”} \]
1. G → g: M, \( SIG(M, K_{G}^{-}) \)

\[ M' = \text{“g accepts } \Omega \text{ using } K_{g}^{+} \text{ and exercise } \Omega \text{ using } K_{g_R}^{+} \text{”} \]
2. g → G: M', \( SIG(M', K_{g}^{-}) \)

\[ M'' = \text{“g, G, } \Omega , K_{G_R}^{+}, K_{g_R}^{+}, K_{g_R}^{+} \text{”} \]
3. G → g: T = M'', \( SIG(M'', K_{G_R}^{-}) \)

4. g then signs T producing the delegation token: T, \( SIG(T, K_{g_R}^{-}) \)
Logic of Delegation

• Need of formalism to analyse delegation protocols.

[Abadi, Burrows,Lamspon,Plotkin “A Calculus for Access Control in Distributed Systems” ACM ToPLaS 1993]

• Cannot express the difference between the two semantics
• Not general
• No distinction between active and passive entities
Future Work

- Interactions between these semantics of delegation with those defined in other areas (e.g., Object-oriented)
- Formal approach
- Implementation of auditing mechanisms
- Principle of the least trust