Security as a Architectural Concern

Reid Holmes
NFP: Performance

- Throughput: Measure of the amount of work an application must perform in a unit of time
- Response time: Measure of the latency an application exhibits
- Deadlines: Work must be done by a specific time
- Definition of work unit and time is important
  - Average loads or peak loads?
  - Large tasks are small tasks?
NFP: Security

- Security: “The protection afforded a system to preserve its **integrity**, **availability**, and **confidentiality** if its resources.”

- Confidentiality

- Integrity

- Availability
Security arch. principles

- Least privilege:

- Fail-safe defaults
  - Deny access if explicit permission is absent.

- Economy of mechanism

- Open design
  - Secrecy != security.
Security arch. principles

- Separation of privilege
  - Introduce multiple parties to avoid exploitation of privileges.

- Least common mechanism
  -

- Psychological acceptability
  - Make security mechanisms usable.

- Defence in depth
  -
IIS Example
Access control

- Decide whether access should be granted.
  - Discretionary:
    - Based on the accessor’s identity, the resources, and whether the accessor has permissions.
  - Mandatory:
    - Policy based. (e.g., dominating labels)
    - Cross-cutting concern that should be investigated at an architectural level.
## Discretionary access control

<table>
<thead>
<tr>
<th></th>
<th>DB</th>
<th>Component</th>
<th>Interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alice</td>
<td>Read-write; always</td>
<td>Bend</td>
<td>Y</td>
</tr>
<tr>
<td>Bob</td>
<td>Read-write; Between 9-5</td>
<td>Fold</td>
<td>N</td>
</tr>
<tr>
<td>Charles</td>
<td>No access</td>
<td>Spindle</td>
<td>N</td>
</tr>
<tr>
<td>Dave</td>
<td>No access</td>
<td>Mutilate</td>
<td>Y</td>
</tr>
<tr>
<td>Eve</td>
<td>Read-only; Always</td>
<td>Non</td>
<td>N</td>
</tr>
</tbody>
</table>
Mandatory access control
Trust management

- Trust is a subjective probability with which one agent assesses another agent's will perform some specific action within a specific context.
- Reputation is the expectation of an agent’s behaviour based on their past behaviours.
- Trust cannot be isolated to individual components.
  - Dominant concern in decentralized applications.
  - Architecture provides a foundation for reasoning about trust-related issues.
Activity

- Create an architecture for iRoadTrip.
  - Components:
    - GPS
    - Timer
    - UI (Create / Join / View / Configure)
    - Geolocation (e.g., Google maps)
    - App Engine
    - Persistence Facade
    - Client Marshaller
    - Server Marshaller
    - Client Storage
iRoadTrip: statechart
iRoadTrip: New trip
iRoadTrip: Location update