OM-AM and PEI

Prof. Ravi Sandhu
THE OM-AM WAY

What?

Objectives
Model
Architecture
Mechanism

How?
LAYERS AND LAYERS

- Multics rings
- Layered abstractions
- Waterfall model
- Network protocol stacks
- Napolean layers
- RoFi layers
- OM-AM
- etcetera
OM-AM AND MANDATORY ACCESS CONTROL (MAC)

What?

No information leakage
Lattices (Bell-LaPadula)
Security kernel
Security labels

How?
OM-AM AND DISCRETIONARY ACCESS CONTROL (DAC)

What?

Owner-based discretion

numerous

numerous

ACLs, Capabilities, etc

How?

Assurance
OM-AM AND ROLE-BASED ACCESS CONTROL (RBAC)

What?

Objective neutral

RBAC96, ARBAC97, etc.

user-pull, server-pull, etc.

certificates, tickets, PACs, etc.

How?
SERVER-PULL

Client

Server

User-role
Authorization
Server
USER-PULL

Client

Server

User-role
Authorization
Server
PROXY-BASED

Client → Proxy Server → Server

User-role Authorization Server
## THE OM-AM WAY

<table>
<thead>
<tr>
<th>What?</th>
<th>How?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Objectives</td>
<td>Architecture</td>
</tr>
<tr>
<td>Model</td>
<td>Mechanism</td>
</tr>
</tbody>
</table>

© Ravi Sandhu
Security and system goals (objectives/policy)

- Necessarily informal
- Specified using users, subjects, objects, admins, labels, roles, groups, etc. in an ideal setting.
- Security analysis (objectives, properties, etc.).

Policy models

- Approximated policy realized using system architecture with trusted servers, protocols, etc.
- Enforcement level security analysis (e.g. stale information due to network latency, protocol proofs, etc.).
- Technologies such as Cloud Computing, Trusted Computing, etc.
- Implementation level security analysis (e.g. vulnerability analysis, penetration testing, etc.)
- Software and Hardware

Enforcement models

Implementation models

Concrete System