AUTHORIZATION, TRUST AND RISK

- Information security is fundamentally about managing
  - authorization and
  - trust
  so as to manage risk
THE OM-AM WAY

What?

Model

Architecture

Mechanism

Assurance

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

What?

Objective neutral
RBAC96, ARBAC97, etc.
user-pull, server-pull, etc.
certificates, tickets, PACs, etc.

How?

RBAC96

ROLE HIERARCHIES
USER-ROLE ASSIGNMENT
PERMISSIONS-ROLE ASSIGNMENT

USERS
ROLES
PERMISSIONS
SESSIONS
CONSTRAINTS
EXAMPLE ROLE HIERARCHY

Director (DIR)

Project Lead 1 (PL1)
  Production 1 (P1)
    Engineer 1 (E1)

Project Lead 2 (PL2)
  Production 2 (P2)
    Engineer 2 (E2)

Quality 1 (Q1)

Quality 2 (Q2)

Engineering Department (ED)

Employee (E)

PROJECT 1

PROJECT 2
EXAMPLE ROLE HIERARCHY

Director (DIR)

Project Lead 1 (PL1)
  Production 1 (P1)
  Engineer 1 (E1)
  Quality 1 (Q1)

Project Lead 2 (PL2)
  Production 2 (P2)
  Engineer 2 (E2)
  Quality 2 (Q2)

PROJECT 1

PROJECT 2

EXAMPLE ROLE HIERARCHY

Project Lead 1 (PL1)
  Production 1 (P1)
  Engineer 1 (E1)
  Quality 1 (Q1)

Project Lead 2 (PL2)
  Production 2 (P2)
  Engineer 2 (E2)
  Quality 2 (Q2)

PROJECT 1

PROJECT 2
LBAC: LIBERAL *-PROPERTY

```
H
M1
L
M2
```

<table>
<thead>
<tr>
<th></th>
<th>Read</th>
<th>Write</th>
</tr>
</thead>
<tbody>
<tr>
<td>M1</td>
<td>-</td>
<td>+</td>
</tr>
<tr>
<td>M2</td>
<td>+</td>
<td>-</td>
</tr>
</tbody>
</table>

RBAC96: LIBERAL *-PROPERTY

```
HR
M1R
LR
M2R
```

```
LW
M1W
HW
M2W
```

<table>
<thead>
<tr>
<th></th>
<th>Read</th>
<th>Write</th>
</tr>
</thead>
<tbody>
<tr>
<td>M1R</td>
<td>-</td>
<td>+</td>
</tr>
<tr>
<td>M2R</td>
<td>+</td>
<td>-</td>
</tr>
<tr>
<td>M1W</td>
<td>-</td>
<td>+</td>
</tr>
<tr>
<td>M2W</td>
<td>+</td>
<td>-</td>
</tr>
</tbody>
</table>
**RBAC96: LIBERAL *-PROPERTY**

- user $\in xR$, user has clearance $x$
- user $\in LW$, independent of clearance
- Need constraints
  - session $\in xR$ iff session $\in xW$
  - read can be assigned only to $xR$ roles
  - write can be assigned only to $xW$ roles
  - $(O,\text{read})$ assigned to $xR$ iff
    $(O,\text{write})$ assigned to $xW$

**LBAC: STRICT *-PROPERTY**

- M1
- M2
- L
- H
- Read
- Write
- $+$
- $-$
RBAC96: STRICT *-PROPERTY

Variations of DAC

- Strict DAC
- Liberal DAC
Revocation

- Grant-Independent Revocation.
- Grant-Dependent Revocation.

SERVER MIRROR
PROXY-BASED

Client <-> Proxy Server <-> Server

User-role Authorization Server

MECHANISMS

- SSL
- X.509 certificates
- Secure cookies
- LDAP
- Etc