The Authorization Leap from Rights to Attributes: Maturation or Chaos?

Prof. Ravi Sandhu
Executive Director and Endowed Chair

SACMAT
June 21, 2012

ravi.sandhu@utsa.edu
www.profsandhu.com
www.ics.utsa.edu
Dozens of models proposed and studied. Only three winners (meaningful practical traction)
- DAC: Discretionary Access Control, 1970
- MAC: Mandatory Access Control, 1970
- RBAC: Role-Based Access Control, 1995

RBAC emerged at an inflection point due to
dissatisfaction with the then dominant DAC and MAC
- We are currently at another inflection point due to
dissatisfaction with the now dominant RBAC
- ABAC (Attribute-Based Access Control) has emerged as the prime candidate to be the next dominant paradigm
NO!! Never!!

Is ABAC the right word for the moment?
- Certainly a strong candidate
- Already too late?
  - ReBAC (relationship-based access control) not ABAC
  - Big Data, Analytics and AI will take care of everything

ABAC is exponentially more complex than anything that has been an Access Control winner so far (DAC, MAC, RBAC)
- We need the complexity, but need to manage it
- If Google can index the web, we can do ABAC!!
Attribute-Based Access Control (ABAC)

- Attributes are name:value pairs
  - possibly chained
- Associated with
  - users
  - subjects
  - objects
  - contexts
    - device, connection, location, environment, system ...
- Converted by policies into rights just in time
  - policies specified by security architects
  - attributes maintained by security administrators
  - ordinary users morph into architects and administrators
Authorization Leap

Rights to attributes
- Rights
- Labels
- Roles
- Attributes

Maturation  ??  Chaos

Benefits
- Decentralized
- Dynamic
- Contextual
- Consolidated

Risks
- Complexity
- Confusion
- Attribute trust
- Policy trust

© Ravi Sandhu  World-Leading Research with Real-World Impact!
Cyber technologies and systems trends will drive pervasive adoption of ABAC
  - RBAC is simply not good enough

ABAC deployment is going to be messy but need not be chaotic

Researchers can facilitate ABAC adoption and reduce chaos by developing
  - Models
  - Theories
  - Systems
Authorization Challenges

Policy Specification

Policy Enforcement

Policy Reality

Policy Administration

© Ravi Sandhu

World-Leading Research with Real-World Impact!
- Analog Hole
- Inference
- Covert Channels
- Side Channels
- Spoofing
- Attack Asymmetry
- Compatibility
- ....
The RBAC Story

Proposed Standard

Standard Adopted

Amount of Publications

Year of Publication

Pre-RBAC

Early RBAC

2nd expansion phase

© Ravi Sandhu

World-Leading Research with Real-World Impact!
ABAC Status

1990?

ABAC still in pre/early phase

© Ravi Sandhu

World-Leading Research with Real-World Impact!
Role Hierarchy (RH)

Constraints
RBAC Policy Configuration Points

Role Hierarchy (RH)

Security Administrator

Security Architect

User Sessions

Session Roles

Constraints
NIST model limits constraints to Static and Dynamic Separation of Duties
ABAC Status

1990?

ABAC still in pre/early phase


Year of Publication

Amount of Publications

Proposed Standard

Standard Adopted

1990?

2012

1990?

Pre-RBAC
Early RBAC
2nd expansion phase

© Ravi Sandhu

World-Leading Research with Real-World Impact!
ABAC Prior Work Includes

- X.509, SPKI Attribute Certificates (1999 onwards)
  - IETF RFCs and drafts
  - Tightly coupled with PKI (Public-Key Infrastructure)
- XACML (2003 onwards)
  - OASIS standard
  - Narrowly focused on particular policy combination issues
  - Fails to accommodate the ANSI-NIST RBAC standard model
  - Fails to address user subject mapping
- Usage Control or UCON (Park-Sandhu 2004)
  - Fails to address user subject mapping
  - Focus is on extended features
    - Mutable attributes
    - Continuous enforcement
    - Obligations
    - Conditions
RBAC Overall Assessment

- Role granularity is not adequate leading to role explosion
  - Researchers have suggested several extensions such as parameterized privileges, role templates, parameterized roles (1997-)
- Role design and engineering is difficult and expensive
  - Substantial research on role engineering top down or bottom up (1996-), and on role mining (2003-)
- Assignment of users/permissions to roles is cumbersome
  - Researchers have investigated decentralized administration (1997-), attribute-based implicit user-role assignment (2002-), role-delegation (2000-), role-based trust management (2003-), attribute-based implicit permission-role assignment (2012-)
- Adjustment based on local/global situational factors is difficult
  - Temporal (2001-) and spatial (2005-) extensions to RBAC proposed
- RBAC does not offer an extension framework
  - Every shortcoming seems to need a custom extension
  - Can ABAC unify these extensions in a common open-ended framework?
ABAC Research Agenda

1. Foundational Principles and Theory

2. Core ABAC Models

3. Administrative ABAC Models

4. Extended ABAC Models

5. ABAC Policy Languages

6. ABAC Enforcement Architectures

7. ABAC Design and Engineering
7. Design and Engineering:


1. Foundational Principles and Theory


NOTE: Only a small sampling of the RBAC literature is cited in this diagram
ABAC Research Agenda

7. ABAC Design and Engineering

5. ABAC Policy Languages

3. Administrative ABAC Models

4. Extended ABAC Models

2. Core ABAC Models

6. ABAC Enforcement Architectures

1. Foundational Principles and Theory
ABAC Research Agenda

1. Foundational Principles and Theory

2. Core ABAC Models
   Initial Results

3. Administrative ABAC Models

4. Extended ABAC Models

5. ABAC Policy Languages

6. ABAC Enforcement Architectures

7. ABAC Design and Engineering

© Ravi Sandhu

World-Leading Research with Real-World Impact!
Approach this challenge from several perspectives
Initial results on a bottom-up approach

ABACα model (DBSEC 2012)
- Just sufficient to cover the core of DAC, MAC and RBAC
- No extraneous features (however attractive and desirable)

ABACβ model (in progress)
- Grow ABACα to accommodate additional models, including numerous RBAC extensions and RBAC-related models (e.g. RT)
### ABACα Requirements

<table>
<thead>
<tr>
<th>Subject attribute values constrained by creating user?</th>
<th>Object attribute values constrained by creating subject?</th>
<th>Attribute range ordered?</th>
<th>Attribute functions return set value?</th>
<th>Object attributes modification?</th>
<th>Subject attribute modification by creating user?</th>
</tr>
</thead>
<tbody>
<tr>
<td>DAC</td>
<td>YES</td>
<td>NO</td>
<td>YES</td>
<td>YES</td>
<td>NO</td>
</tr>
<tr>
<td>MAC</td>
<td>YES</td>
<td>YES</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
</tr>
<tr>
<td>RBAC₀</td>
<td>YES</td>
<td>NA</td>
<td>NO</td>
<td>YES</td>
<td>NA</td>
</tr>
<tr>
<td>RBAC₁</td>
<td>YES</td>
<td>NA</td>
<td>YES</td>
<td>YES</td>
<td>YES</td>
</tr>
<tr>
<td>ABACα</td>
<td>YES</td>
<td>YES</td>
<td>YES</td>
<td>YES</td>
<td>YES</td>
</tr>
</tbody>
</table>
ABACα Model Structure

1. Constraints on subject attributes at creation and modification time.
2. Constraints on object attributes at creation and modification time.
3. Authorization policy

© Ravi Sandhu
World-Leading Research with Real-World Impact!
U, S and O represent finite sets of existing users, subjects and objects respectively.

UA, SA and OA represent finite sets of user, subject and object attribute functions respectively. (Henceforth referred to as simply attributes.)

P represents a finite set of permissions.

For each att in UA \cup SA \cup OA, Range(att) represents the attribute’s range, a finite set of atomic values.

SubCreator: S \rightarrow U. For each subject SubCreator gives its creator.

attType: UA \cup SA \cup OA \rightarrow \{set, atomic\}. Specifies attributes as set or atomic valued.

Each attribute function maps elements in U, S and O to atomic or set values.

\forall ua \in UA. ua : U \rightarrow \begin{cases} 
\text{Range}(ua) & \text{if attType}(ua) = \text{atomic} \\
2^{\text{Range}(ua)} & \text{if attType}(ua) = \text{set}
\end{cases}

\forall sa \in SA. sa : S \rightarrow \begin{cases} 
\text{Range}(sa) & \text{if attType}(sa) = \text{atomic} \\
2^{\text{Range}(sa)} & \text{if attType}(sa) = \text{set}
\end{cases}

\forall oa \in OA. oa : O \rightarrow \begin{cases} 
\text{Range}(oa) & \text{if attType}(oa) = \text{atomic} \\
2^{\text{Range}(oa)} & \text{if attType}(oa) = \text{set}
\end{cases}
ABACα Additional Components

- **Administrative Functions**
  - AddUser (u:NAME, uaset:UASET)
  - DeleteUser (u:NAME)
  - ModifyUserAtt (u:NAME, uaset:UASET)

- **System Functions**
  - CreateSubject (u; s:NAME, saset:SASET)
  - DeleteSubject (u; s:NAME)
  - ModifySubjectAtt (u; s:NAME, saset:SASET)

- **Review Functions**
  - UserAttributes (u:NAME)
  - UserOperationsOnObject (u, o: NAME)
  - AssignedUser(ua: NAME, value: Range(ua))
  - UserPermissions(u: NAME)
  - SubjectPermissions(s: NAME)

- **Policy Configuration Languages**
ABACβ Model Structure

Policy Configuration Points
Same as ABACα

Enrich other ABACα Components
## ABACβ Examples

<table>
<thead>
<tr>
<th>Model</th>
<th>Description</th>
<th>Extension-Specific</th>
<th>Extension-General</th>
</tr>
</thead>
<tbody>
<tr>
<td>Attribute based User-Role Assignment [1]</td>
<td>Roles are computed from user attribute based on predefined rules</td>
<td>Subject attribute constrained by user attribute</td>
<td>Extensions to languages for specifying subject attribute constraints.</td>
</tr>
<tr>
<td>Role based Trust Management [2]</td>
<td>Each entity has role. Role is managed by the owned entity.</td>
<td>Chained attribute, distinguished attribute intentional vs extensional representation.</td>
<td>Extensions to nature of attribute.</td>
</tr>
<tr>
<td>Role and Organization based Access Control [3]</td>
<td>User is assigned to role and organization pair.</td>
<td>Attribute should be able to represent pair-value (org, role).</td>
<td>Extensions to nature of attribute.</td>
</tr>
</tbody>
</table>

---

[3]. Zhixiong Zhang, Xinwen Zhang and Ravi Sandhu, ROBAC: Scalable Role and Organization Based Access Control Models, TrustCol 2006
ABAC Research Agenda

1. Foundational Principles and Theory

2. Core ABAC Models
   Initial Results

3. Administrative ABAC Models

4. Extended ABAC Models

5. ABAC Policy Languages

6. ABAC Enforcement Architectures

7. ABAC Design and Engineering

World-Leading Research with Real-World Impact!