Attribute Transformation for Attribute-Based Access Control

Prosunjit Biswas, Ravi Sandhu and Ram Krishnan
Department of Computer Science
Department of Electrical and Computer Engineering
University of Texas, San Antonio

ABAC’17, March 24, 2017, Scottsdale, AZ, USA
Outline

- Summary
- Motivation
- Attribute Transformation
- Attribute Reduction
- Attribute Expansion
- Conclusion
- Q/A

World-Leading Research with Real-World Impact!
We have presented a concept of attribute transformation and specify two types of transformation---attribute reduction and attribute expansion.
Attribute explosion!

Figure 1: Attributes defined for OpenStack Virtual Machines

World-Leading Research with Real-World Impact!
Motivation (continuing)

Attribute Explosion

incurs

difficulties in managing

authorization policies  attribute-value assignments
Motivation (continuing)

We cannot get rid of attributes we need.

But we can manage

with

Attribute Transformation
Attribute Transformation (assumptions)

Attribute types

Non-policy Attributes

Policy Attributes

Assumptions:
Non-policy Attributes \( \cap \) Policy Attributes = \( \phi \)
Non-policy Attributes >> Policy Attributes

Examples:
Object attributes (Non-policy):
size, created_by, shared, location

Object attributes (Policy):
sensitivity, security-label
Attribute Transformation is the process of transforming one set of attribute-value assignments into another set of assignments.

Types of attribute transformation

- **Reduction** (Non-policy Attr → Policy Attr)
- **Expansion** (Policy Attr → Policy Attr)
Attribute Reduction

The process of transforming non-policy attribute-value assignments into policy attributes-value assignments.

Non-policy attributes

- size(f1) = 100MB
- created-by(f1) = system-d
- shared(f1) = false
- location(f1) = /log/system-log

Policy attributes

- security-label(f) = sensitive

Effective assignments

- security-label(f) = sensitive

Derived assignments

- Derived assignments

Attribute reduction
Motivation from literature:

1. Attribute-Based User-Role Assignment [1]

Attribute Reduction (usefulness)

Useful for

- Abstraction
- Modular design
- Hierarchical policy
Authorization policy with Policy attributes:

Can-read $\equiv$ security-label$(o)$ = sensitive $\land$ role$(u)$=manager

Mapping rules with Non-policy Attributes:

VM-mapping $\equiv$ resource-type$(o)$ = VM $\land$ image-type$(o)$ = corporate $\rightarrow$
security-label$(o)$ = sensitive

Firewall-mapping $\equiv$ resource-type$(o)$ = firewall $\land$ protocol$(o)$ = UDP $\land$
network$(o)$ = internal $\rightarrow$ security-label$(o)$ = sensitive
Example of mapping rule:

file-length(f) = 100 MB \land created-by(f) = system-d \land is-shared(f) = false \rightarrow security-label(f) = sensitive
Conflicts resulting from multiple mappings

- resource-type(o) = VM
  - encryption(o) = plain
  - security-label(o) = regular

- resource-type(o) = VM
  - image-type(o) = corporate
  - security-label(o) = sensitive
Attribute Reduction (issues)

Conflicts resulting from assigned and derived values

resource-type(o) = VM

encryption(o) = plain

security-label(o) = regular

security-label(o) = sensitive

mapping1

Derived value

Explicitly assigned value
The process of transforming policy-attribute-value assignments into a different set of policy-attributes-value assignments.

**Policy attributes**

- is-a-veteran(u) = True

**Derived assignments**

- benefits(u) = \{b1, b2\}
- skills(u) = \{skill1, skill2\}

**Derived assignments**

**Resulting assignments**

- is-a-veteran(u) = True
- benefits(u) = \{b1, b2\}
- skills(u) = \{skill1, skill2\}
Motivation from literature:

1. Hierarchical Group and Attribute-Based Access Control (HGABAC) [3]
What next?

- Other forms of Attribute Transformation
- Chain of Attribute Transformation
- Fitting Attribute Transformation in ABAC models

