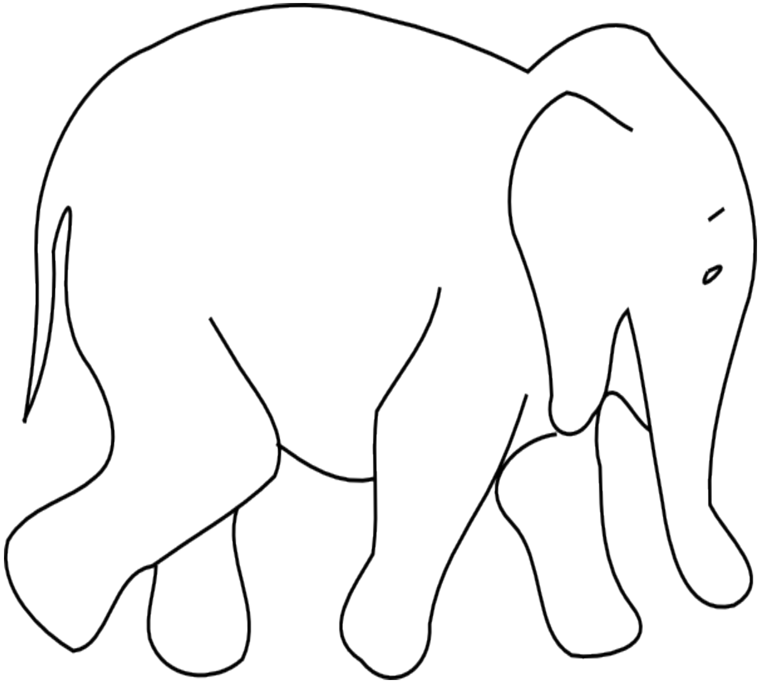
Attribute Transformation for Attribute-Based Access Control

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

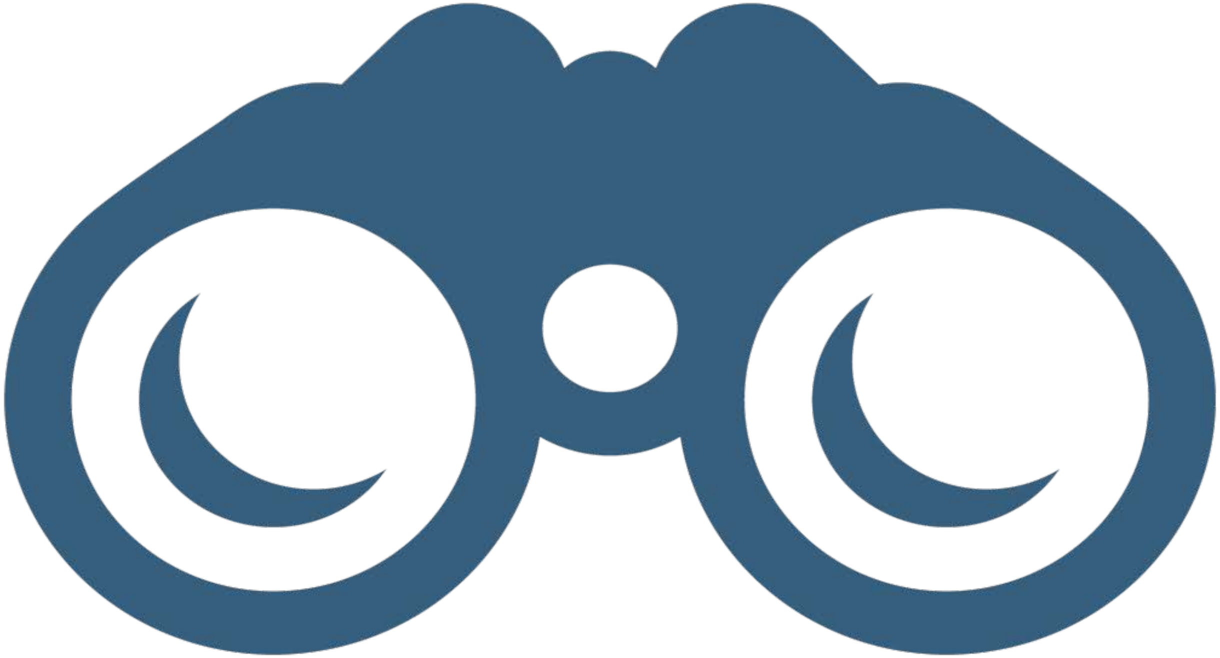
Outline

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



Summary

We have presented a concept of attribute transformation and specify two types of transformation---attribute reduction and attribute expansion.



Motivation

Attribute explosion!

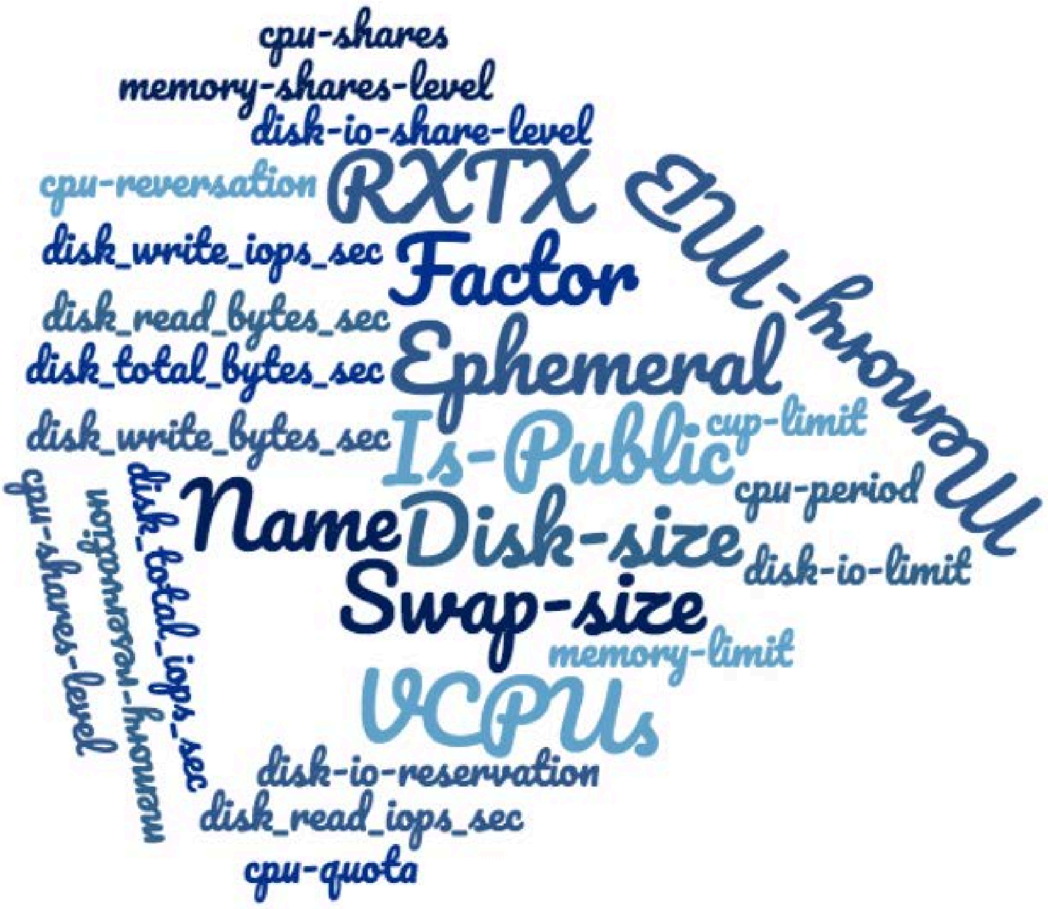
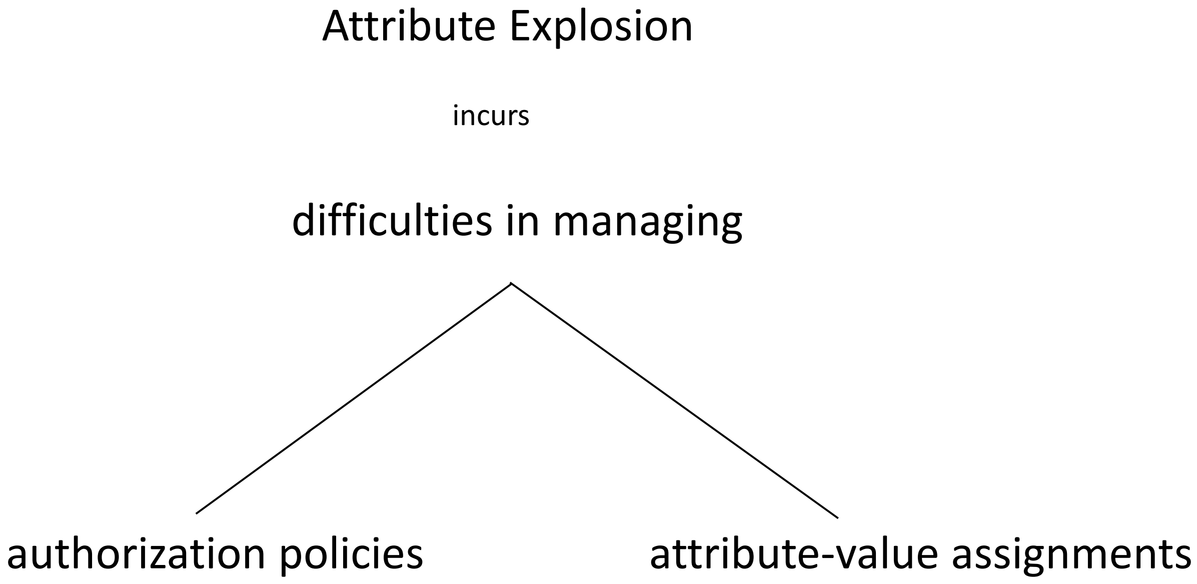


Figure 1: Attributes defined for OpenStack Virtual Machines

Motivation (continuing)



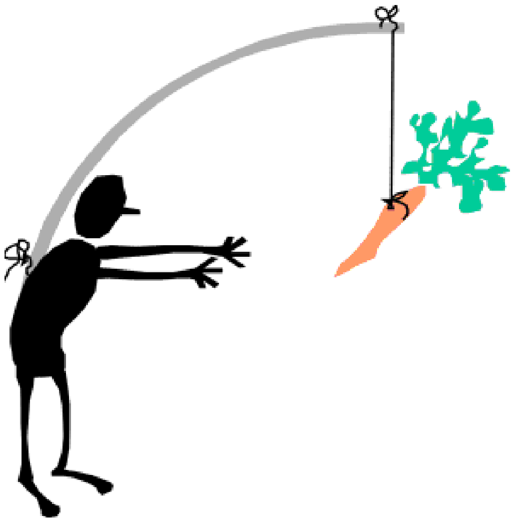
Attribute Explosion

incurs

difficulties in managing

authorization policies

attribute-value assignments



Motivation (continuing)

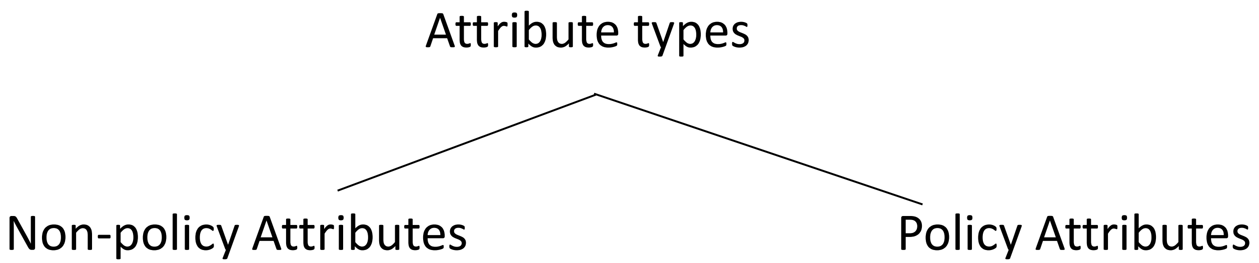
We cannot get rid of attributes we need.

But we can manage

with

Attribute Transformation

Attribute Transformation (assumptions)



Attribute types

Non-policy Attributes

Policy Attributes

Assumptions:

Non-policy Attributes ∩ Policy Attributes = ϕ
Non-policy Attributes >> Policy Attributes

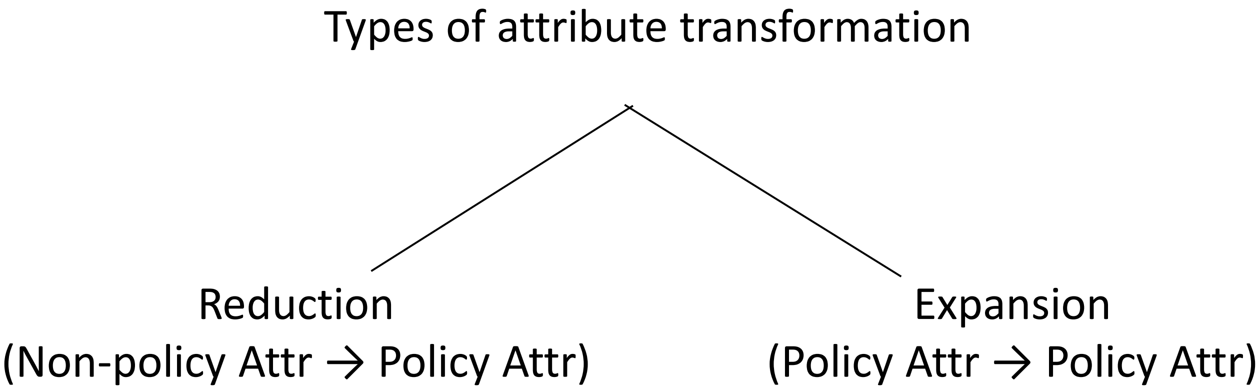
Examples:

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

Object attributes (Policy):
sensitivity, security-label

Attribute Transformation

Attribute Transformation is the process of transforming one set of attribute-value assignments into another set of assignments.



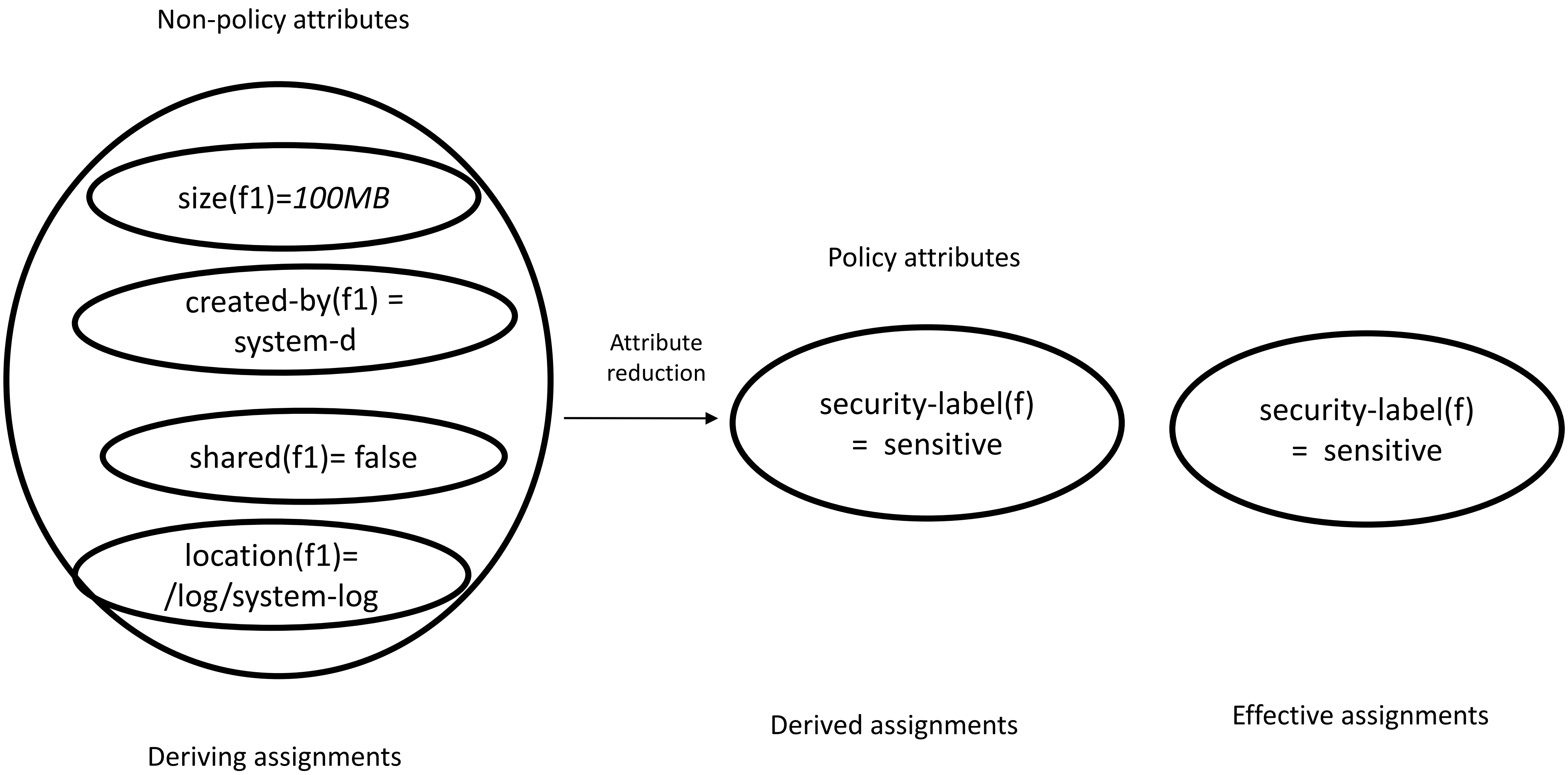
Types of attribute transformation

Reduction
(Non-policy Attr → Policy Attr)

Expansion
(Policy Attr → Policy Attr)

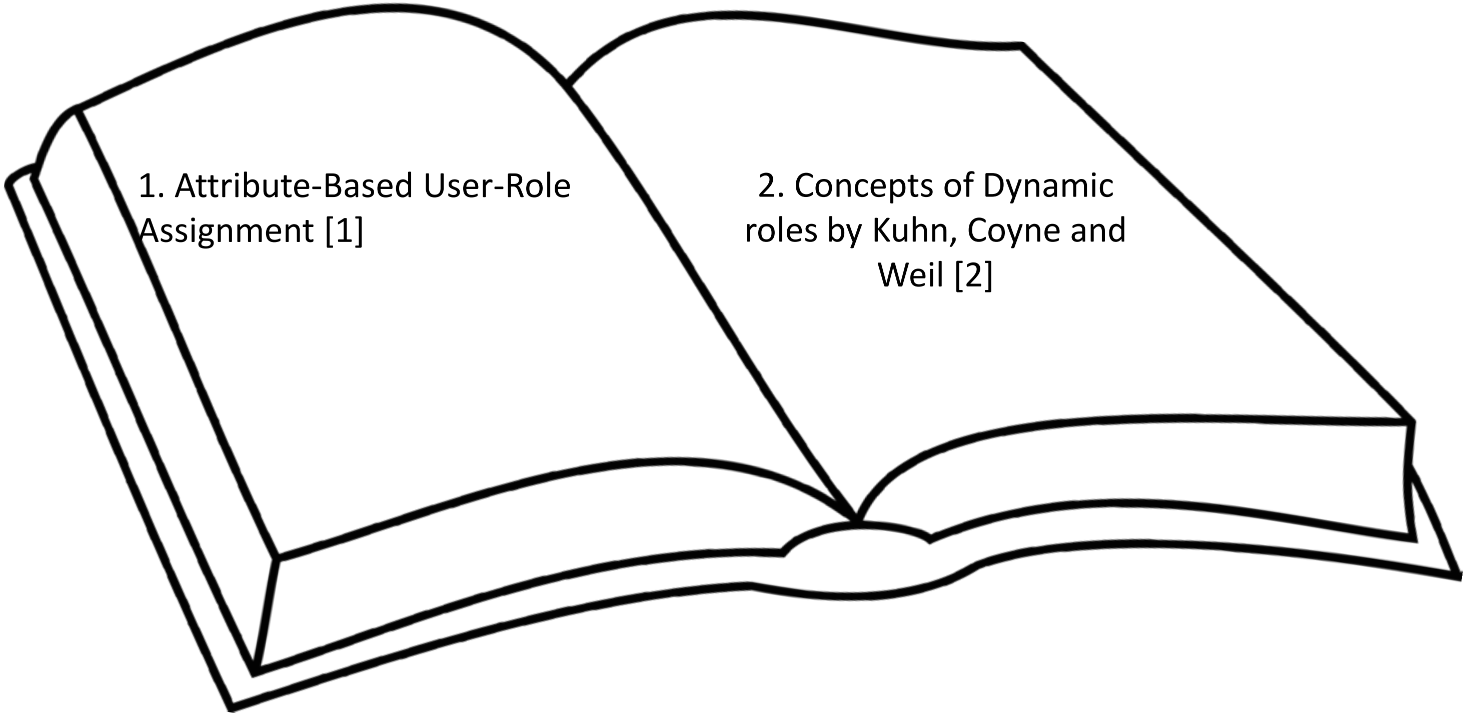
Attribute Reduction

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



Attribute Reduction (motivation)

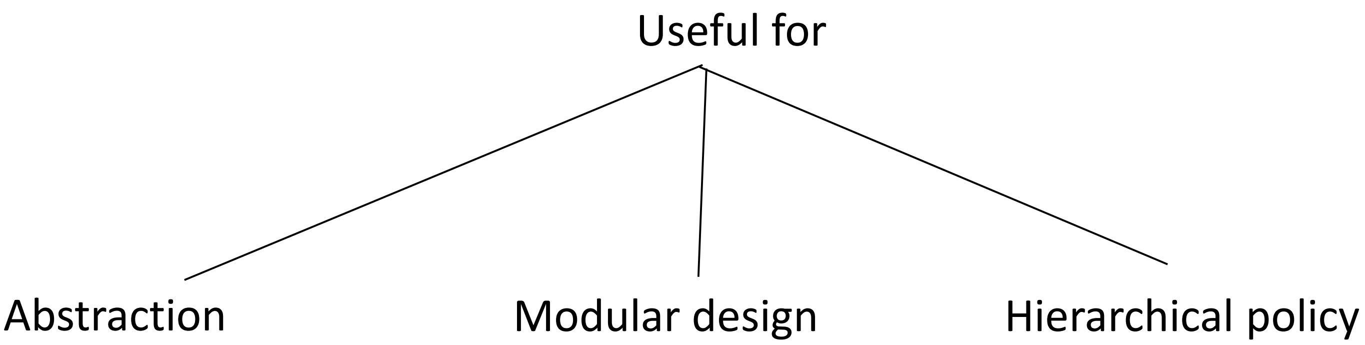
Motivation from literature:

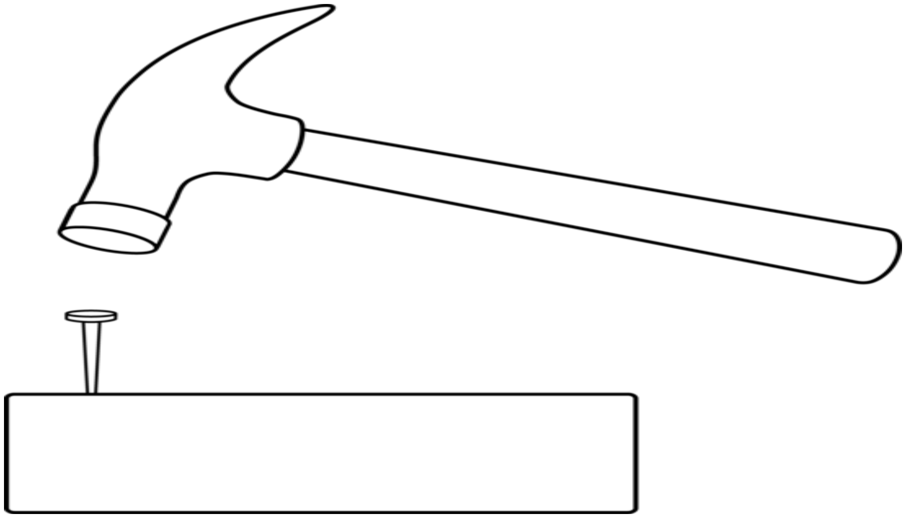


1. Attribute-Based User-Role Assignment [1]

2. Concepts of Dynamic roles by Kuhn, Coyne and Weil [2]

Attribute Reduction (usefulness)





Attribute Reduction (usefulness)

Authorization policy with Policy attributes:

Can-read ≡ security-label(o) = sensitive ∧ role(u)=manager

Mapping rules with Non-policy Attributes:

VM-mapping ≡ resource-type(o) = VM ∧ image-type(o) = corporate → security-label(o) = sensitive

Firewall-mapping ≡ resource-type(o) = firewall ∧ protocol(o) = UDP ∧ network(o) = internal → security-label(o) = sensitive

Attribute Reduction (mapping rules)

Example of mapping rule:

file-length(f) = 100 MB ∧ created-by(f) = system-d ∧ is-shared(f) = false → security-label(f) = sensitive

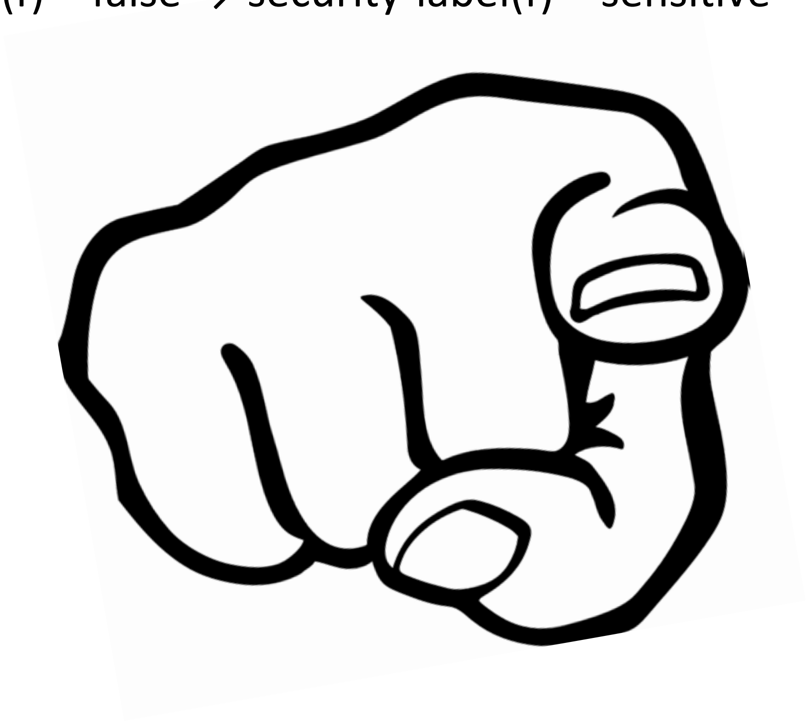
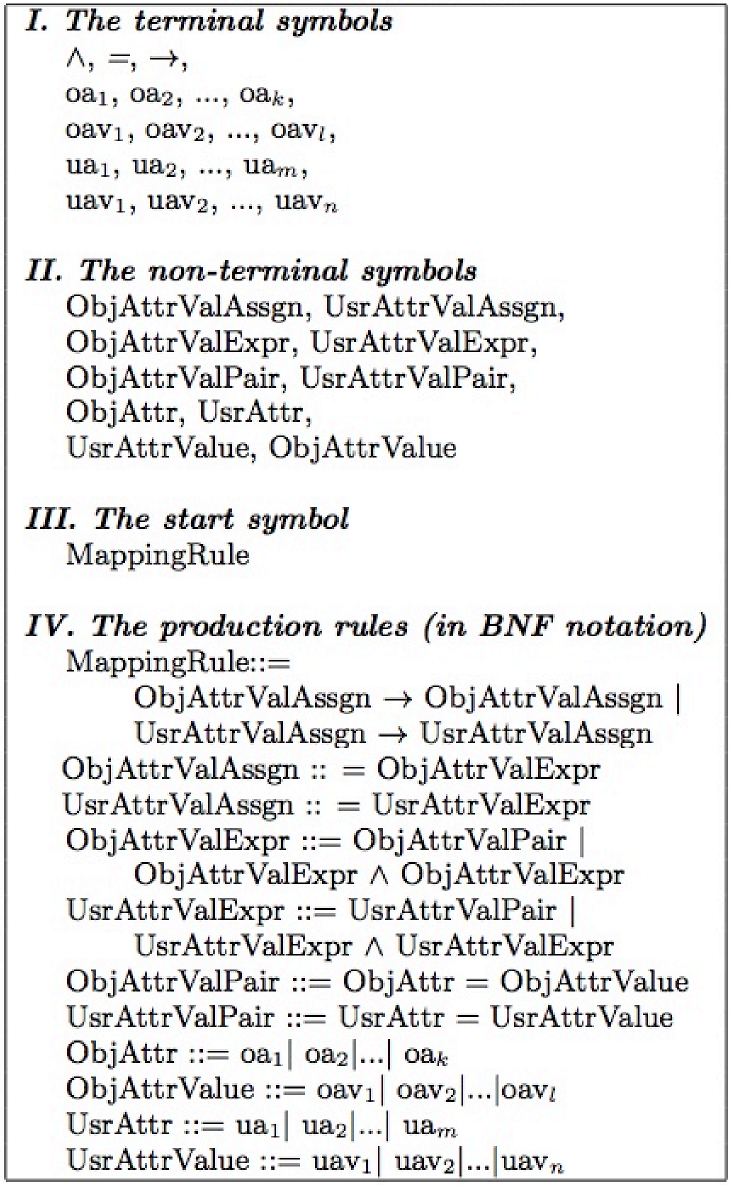


Table 1: Mapping rules



I. The terminal symbols

$\wedge$, $=$, $\rightarrow$,

oa_1, oa_2, ..., oa_k,

oav_1, oav_2, ..., oav_l,

ua_1, ua_2, ..., ua_m,

uav_1, uav_2, ..., uav_n

II. The non-terminal symbols

ObjAttrValAssgn, UsrAttrValAssgn,
ObjAttrValExpr, UsrAttrValExpr,
ObjAttrValPair, UsrAttrValPair,
ObjAttr, UsrAttr,
UsrAttrValue, ObjAttrValue

III. The start symbol

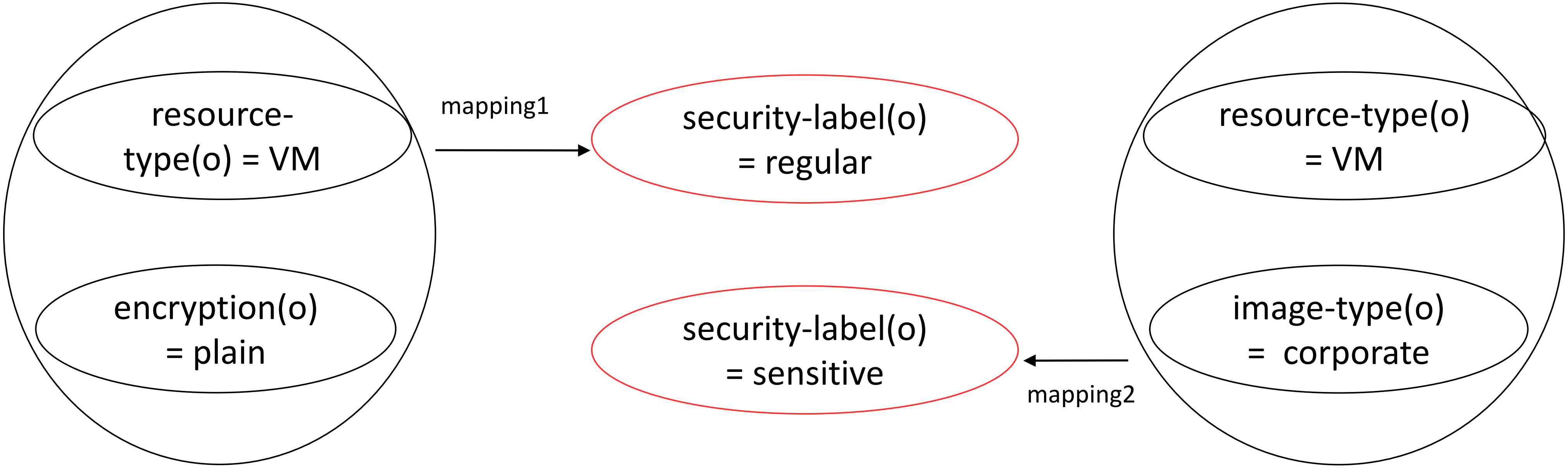
MappingRule

IV. The production rules (in BNF notation)

```
MappingRule::=
        ObjAttrValAssgn → ObjAttrValAssgn |
        UsrAttrValAssgn → UsrAttrValAssgn
ObjAttrValAssgn :: = ObjAttrValExpr
UsrAttrValAssgn :: = UsrAttrValExpr
ObjAttrValExpr ::= ObjAttrValPair |
        ObjAttrValExpr ∧ ObjAttrValExpr
UsrAttrValExpr ::= UsrAttrValPair |
        UsrAttrValExpr ∧ UsrAttrValExpr
ObjAttrValPair ::= ObjAttr = ObjAttrValue
UsrAttrValPair ::= UsrAttr = UsrAttrValue
ObjAttr ::= oa1| oa2|...| oak
ObjAttrValue ::= oav1| oav2|...|oavl
UsrAttr ::= ua1| ua2|...| uam
UsrAttrValue ::= uav1| uav2|...|uavn
```

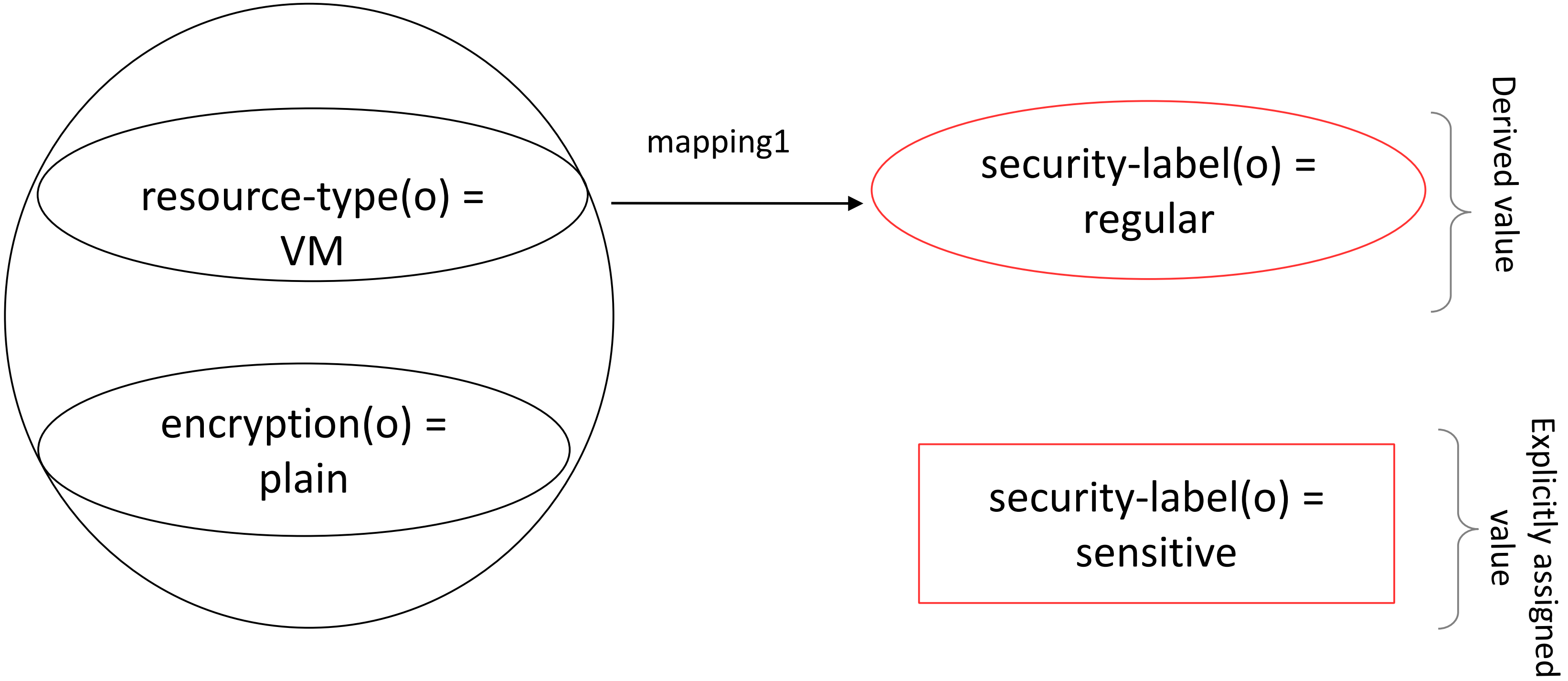
Attribute Reduction (issues)

Conflicts resulting from multiple mappings


resource-type(o) = VM
encryption(o) = plain
mapping1
security-label(o) = regular
security-label(o) = sensitive
mapping2
resource-type(o) = VM
image-type(o) = corporate

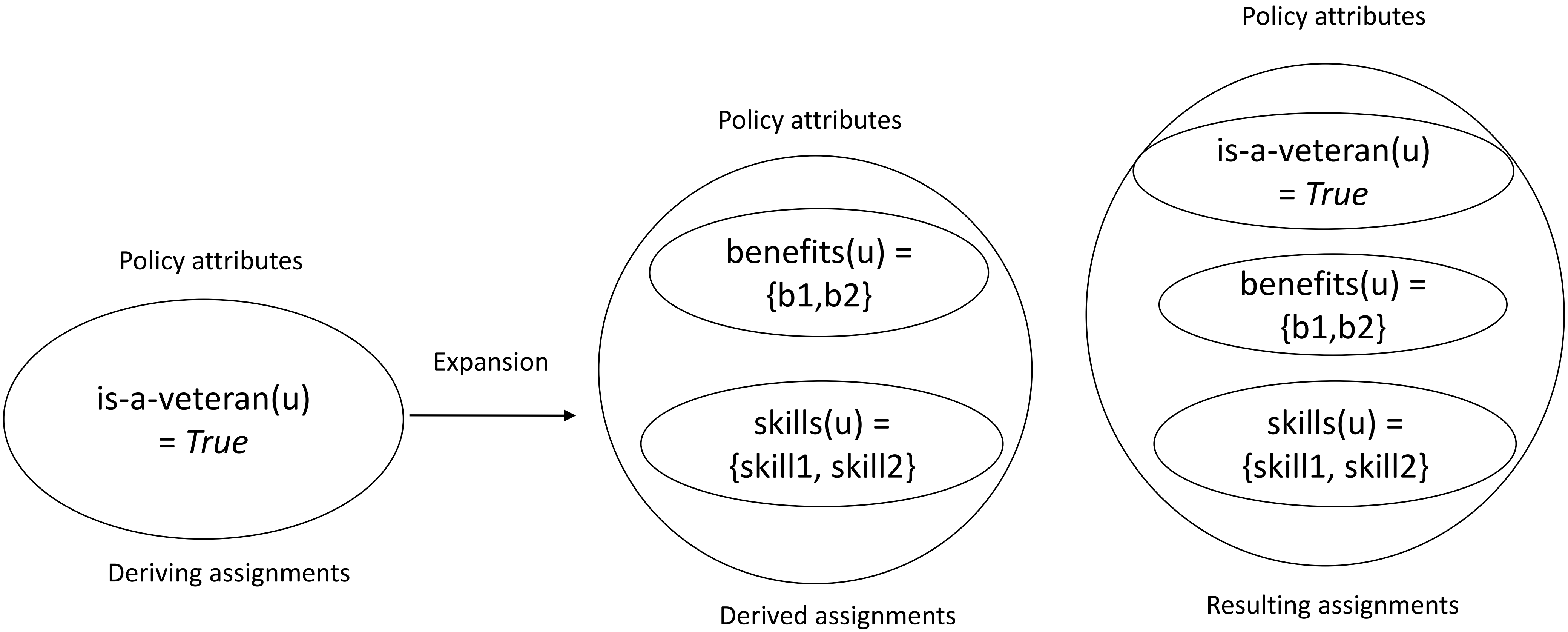
Attribute Reduction (issues)

Conflicts resulting from assigned and derived values


resource-type(o) = VM
encryption(o) = plain
mapping1
security-label(o) = regular
Derived value
security-label(o) = sensitive
Explicitly assigned value

Attribute Expansion

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



Attribute Expansion (motivation)

Motivation from literature:

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

Conclusion

What next?

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

References

1. Servos, Daniel, and Sylvia L. Osborn. "HGABAC: Towards a formal model of hierarchical attribute-based access control." International Symposium on Foundations and Practice of Security. Springer International Publishing, 2014.

2. Kuhn, D. Richard, Edward J. Coyne, and Timothy R. Weil. "Adding attributes to role-based access control." Computer 43.6 (2010): 79-81.

3. Servos, Daniel, and Sylvia L. Osborn. "HGABAC: Towards a formal model of hierarchical attribute-based access control." International Symposium on Foundations and Practice of Security. Springer International Publishing, 2014.