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

Ravi Sandhu
Institute for Cyber Security
University of Texas at San Antonio
San Antonio, Texas
ravi.sandhu@utsa.edu

ABSTRACT

The ongoing authorization leap from rights to attributes offers numerous compelling benefits. Decisions about user, subject, object and context attributes can be made relatively independently and with suitable decentralization appropriate for each attribute. Policies can be formulated by security architects to translate from attributes to rights. Dynamic elements can be built into these policies so the outcomes of access control decisions automatically adapt to changing local and global circumstances. On the benefits side this leap is a maturation of authorization matching the needs of emerging cyber technologies and systems. On the risks side devolving attribute management may lead to attributes of questionable provenance and value, with attendant possibility of new channels for social engineering and malware attacks. We argue that the potential benefits will lead to pervasive deployment of attribute-based access control (ABAC), and more generally attribute-based security. The cyber security research community has a responsibility to develop models, theories and systems which enable safe and chaos-free deployment of ABAC. This is the current grand challenge for access control researchers.

Categories and Subject Descriptors

D.4.6 [**Operating Systems**]: Security and Protection—Access controls; K.6.5 [**Management of Computing and Information Systems**]: Security and Protection

General Terms

Security, Privacy

Keywords

Authorization, Rights, Attributes

1. INTRODUCTION

Access control has been a central component of cyber security for over four decades, and will remain so for decades. Access control seeks answers to fundamental questions of cyber security: Who is authorized to access specific objects and in what mode (e.g., read, write)? Who determines overall policy for this purpose? In whose interests is such policy deployed? Where and how is this policy articulated? How do we comprehend and manage interactions between various policy components? Who controls and manages details of access by specific users to specific objects? How does access control policy evolve and adapt? How do we enforce access controls, especially in large distributed systems? How do we achieve adequate assurance regarding enforcement?

Researchers have developed dozens of access control models to address such questions. Only three have received meaningful practical traction: Discretionary Access Control (DAC) [5, 6], Lattice-Based Access Control (LBAC[1]) [1, 2] and Role-Based Access Control (RBAC) [3, 8]. Numerous others have been proposed and studied (too many to cite even a small sample) providing fundamental insights and theoretical understanding, and articulating interpretations and requirements of access control in new domains such as workflow systems, geospatial systems, digital rights management and social media. Nonetheless DAC, LBAC and RBAC remain the dominant paradigms in practice so far.

DAC, LBAC and RBAC have strong mathematical and intuitive foundations. The intuition underlying DAC is that the owner of a resource should control who can access that resource. LBAC seeks to enforce one-directional information flow in a lattice of security labels. The intuitive concept of RBAC is that access should be determined by function via the role abstraction, rather than by identity or clearance. RBAC is fundamentally different from DAC and LBAC in its deliberate lack of built-in policy. The overriding concept rather is mechanistic in requiring the primacy of a role as the unit that enables authorization. The actual function or purpose of each role is left unspecified.

DAC and LBAC emerged almost concurrently with the development of multi-user computers in the late 1960s and dominated access control for a quarter century. Although, nascent notion of roles had been used in commercial applications and access control products since the early 1970s, RBAC remained an amorphous concept and did not gain significant traction amongst researchers and practitioners until publication of the RBAC96 family of core RBAC models [8]. Since RBAC's emergence in the early to mid 1990s with solid conceptual and formal foundations, it has become the dominant form of access control in commercial systems.

2. FROM RBAC TO ABAC

Even though RBAC has been enthusiastically received and practised there has been an undercurrent of dissatisfaction

[1]Equivalently known as Mandatory Access Control (MAC), Multi-Level Security (MLS) or BLP (Bell-LaPadula).

SACMAT'12, June 20–22, 2012, Newark, New Jersey, USA.
ACM 978-1-4503-1295-0/12/06.

beginning almost contemporaneously with its success. Anecdotal evidence from the author's practitioner contacts indicates the common feeling, "We are using RBAC because there is nothing better at the moment." Researchers have been acutely aware of RBAC's shortcomings and have proposed a variety of incremental improvements. The major issues with RBAC include the following: role granularity is inadequate for fine-grained authorization leading to role explosion, explicit user-role and permission-role assignment by administrators is cumbersome, role design and engineering is difficult and expensive, adjustment based on local/global situational factors is difficult, and there is a proliferation of extensions to the core RBAC models.

Over the past decade ABAC has slowly but surely emerged as a strong candidate to supplant or supplement RBAC. Intuitively, an attribute is a property usually expressed as a name:value pair which can be associated with any entity in the system, including users, subjects, objects and contexts. Suitably defined attributes can represent security labels, clearances and classifications (LBAC), identities and access control lists (DAC) and roles (RBAC). Thereby ABAC supplements and subsumes rather than supplants these currently dominant models. Moreover any number of additional attributes such as location, time of day, strength of authentication, departmental affiliation, qualification, frequent flyer status, and so on, can be brought into play within the same extensible framework of attributes. Thus the proliferation of RBAC extensions might be unified by adding appropriate attributes within a common framework, solving many of these shortcomings of core RBAC. At the same time we should recognize that ABAC with its flexibility may further confound the problem of role design and engineering. Attribute engineering is likely to be a more complex activity, and a price we may need to pay for added flexibility.

Much as nascent RBAC concepts were around for decades before their formalization in 1996 [4], nascent ABAC notions have been around for a while. X.500, X.509, LDAP and XACML are familiar practitioner standards. In academic research ABAC models, such as [7, 9], have been proposed in specific contexts. The situation with ABAC today is analogous to that of RBAC in the early 1990s, as a promising but amorphous concept without authoritative conceptual and formal foundations. The persistence of ABAC notions, even in absence of such foundations, indicates its native appeal and suggests that ABAC will be with us for a very long time.

The core compelling value of ABAC is divide and conquer. Decisions about user, subject, object and context attributes can be made relatively independently and with suitable decentralization appropriate for each attribute. Policies can be formulated by security architects to translate from attributes to rights. Dynamic elements can be built into these policies so the outcomes of access control decisions automatically adapt to changing local and global circumstances. This is very much in line with the needs of emerging cyber systems and technologies. The core risk of ABAC lies in the potential chaos that can result from assembling multiple independent, and possibly conflicting, decisions predictably into a coherent whole. Moreover, attributes may have questionable provenance and value due to malfeasance by malicious users. Even well-meaning and diligent users can be led astray by phishing, social engineering and surreptitious malware. Surely ABAC will introduce new channels for such attacks. Is ABAC then a recipe for chaos?

3. THE GRAND CHALLENGE

In a nutshell, the grand challenge is how to garner the promised benefits of ABAC without engendering chaos? We believe this can happen only if we are able to develop rich and usable models and architectures for ABAC with strong conceptual and formal foundations. We need to do for ABAC what the RBAC96 model [8], the NIST standard [3] and their numerous extensions and enhancements did for RBAC.

Is there a guarantee that such ABAC models and architectures will be found? In science the only guarantee is in the finding. However, there are several reasons to believe that formulation of such ABAC models is well within our grasp. In particular, the innovative research and practical deployments inspired by the RBAC96 model over the past fifteen plus years give us a promising road map for ABAC. ABAC research can be more systematic and organized than was possible with RBAC, given our accumulated understanding of access control since the birth of RBAC. We are confident that an aggressive and coordinated research thrust in ABAC will achieve for ABAC at least what was achieved for RBAC via a rather ad hoc research agenda in its early days (mid to late 1990s).

We believe that practitioners will move to ABAC regardless of what the research community does, because the benefits are too compelling to bypass. Ad hoc efforts by practitioners to build ABAC systems are likely to lead to chaos rather than maturation. Hence the grand challenge!

Acknowledgments The author's work is partially supported by grants from AFOSR, NSF and the State of Texas.

4. REFERENCES

[1] D. Bell and L. LaPadula. Secure computer systems: Unified exposition and Multics interpretation. Technical report, Mitre, 1975.

[2] D. Denning. A lattice model of secure information flow. *Communications of the ACM*, 19(5):236–243, 1976.

[3] D. F. Ferraiolo, R. Sandhu, S. Gavrila, D. R. Kuhn, and R. Chandramouli. Proposed NIST standard for role-based access control. *ACM TISSEC*, 4(3):224–274, August 2001.

[4] L. Fuchs, G. Pernul, and R. Sandhu. Roles in information security: A survey and classification of the research area. *Comp. and Sec.*, 30(8):748 – 769, 2011.

[5] G. Graham and P. Denning. Protection – principles and practice. In *AFIPS Spring Joint Computer Conference*, pages 40:417–429, 1972.

[6] B. Lampson. Protection. In *5th Princeton Symposium on Information Science and Systems*, pages 437–443, 1971. Reprinted in *ACM Operating Systems Review* 8(1):18–24, 1974.

[7] J. Park and R. Sandhu. The UCON_{ABC} usage control model. *ACM TISSEC*, 7(1):128–174, February 2004.

[8] R. Sandhu, E. Coyne, H. Feinstein, and C. Youman. Role-Based Access Control Models. *IEEE Computer*, pages 38–47, 1996.

[9] L. Wang, D. Wijesekera, and S. Jajodia. A logic-based framework for attribute based access control. In *ACM FMSE Workshop*, pages 45–55, 2004.