

# A Perspective on Graphs and Access Control Models

Ravi Sandhu

George Mason University and NSD Security  
ISE Department, MS4A4  
George Mason University  
Fairfax, VA 22030, USA  
[sandhu@gmu.edu](mailto:sandhu@gmu.edu)  
<http://www.list.gmu.edu>

**Abstract.** There would seem to be a natural connection between graphs and information security. This is particularly so in the arena of access control and authorization. Research on applying graph theory to access control problems goes back almost three decades. Nevertheless it is yet to make its way into the mainstream of access control research and practice. Much of this prior research is based on first principles, although more recently there have been significant efforts to build upon existing graph theory results and approaches. This paper gives a perspective on some of the connections between graphs and their transformations and access control models, particularly with respect to the safety problem and dynamic role hierarchies.

## 1 Introduction

In concept there appears to be a strong potential for graphs and their transformations to be applied to information security problems. In practice, however, this potential largely remains to be realized. Applications of graph theory in the security domain go back almost three decades and there has been a steady trickle of papers exploring this potential. Nonetheless graph theory has yet to make its way into the mainstream of security research and practice. In part this may be due to the relative youth of the security discipline and the particular focus of the research community in the early years. Because of the versatility of graph representations and graph theory techniques perhaps it is only a matter of time before a strong and compelling connection is found.

Information security is a broad field and offers multiple avenues for application of graph theory. To pick just two examples, in recent years we have seen application of graph theory in penetration testing and vulnerability analysis [2, 7, 17, 20, 29] and in authentication metrics [21]. It is beyond the scope of this paper to consider the vast landscape of information security. Rather we will focus on the specific area of access control and authorization.

We begin with a brief review of access control and access control models, and then identify two specific problems of access control where graph theory has been

employed in the past. These are the so-called safety problem and the problem of dynamic hierarchies. The rest of the paper explores past work in these two problem areas in some detail and concludes with a brief discussion of possible future research.

## Access Control

Access control is concerned with the question of who can do what in a computer system. Clearly the same object (such as a file) may be accessible by different users in different ways. Some users may be able to read and write the file, others to just read it and still others who have no access to the file. Strictly speaking users do not manipulate files directly but rather do so via programs (such as a text editor or word processor). A program executing on behalf of a user is called a subject, so access control is concerned with enforcing authorized access of subjects to objects. This basic idea was introduced by Lampson in a classic paper [14] and continues to be the central abstraction of access control. Authorization in Lampson's access matrix model is determined by access rights (such as r for read and w for write) in the cells of an access matrix. An example of an access matrix is shown in figure 1. Here subject U can read and write file F but only read file G. Subject V can read and write file G but has no access to file F. A review of the essential concepts of access control is available in [25].

	F	G		
U	r w	r		
V		r w		

**Fig. 1.** Example of an Access Matrix.

The access matrix of figure 1 can be easily depicted as a directed graph with labelled edges as shown in figure 2. Thereby the intuitive feeling that there is a strong connection between graphs and access control. For convenience, we will henceforth talk of the access matrix and access graph as equivalent notions.

## Access Control Models

A static access graph is not very interesting. Real computer systems are highly dynamic in that the access rights of subjects to objects change over time and new subjects and objects (and thereby new rights) are created and existing ones

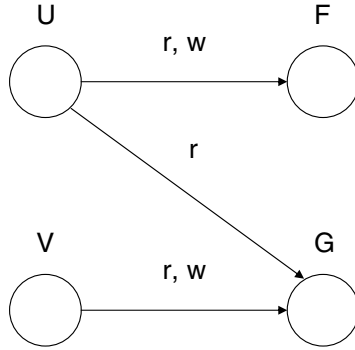


Fig. 2. Example of an Access Graph.

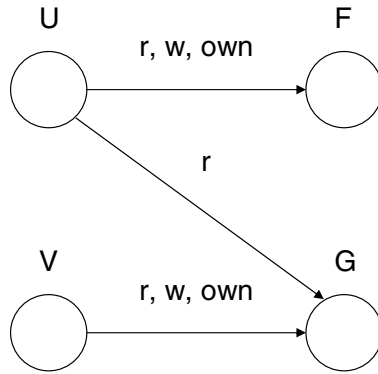


Fig. 3. Owner-Based Discretionary Access Control.

deleted. In terms of the access matrix this means that not only can contents of existing cells be changed but new rows and columns can be created and existing ones destroyed. In terms of the access graph, in addition to edge adding and deleting operations new nodes can be created and existing ones deleted.

An access control model specifies the operations by which the access graph can be changed. These operations are typically authorized by existing rights in the access graph itself. A common example of this is the “own” right shown in figure 3. The owner of a file has the own right for it and can add and delete rights for that file at the owner’s free discretion. Thus subjects U and V control the rights of all subjects to files F and G respectively, i.e., U and V control the addition and deletion of edges labelled r or w terminating in F and G respectively.

The policy of owner-based discretionary access control is certainly reasonable but researchers quickly realized that there are many other policies of practical interest. For example, can the “own” right itself be granted? Some systems do not allow this. The creator of a file becomes its owner and remains its owner thereafter. Other systems allow ownership to be propagated from one subject to

another. Some of these allow multiple simultaneous ownership, while other allow only one owner at a time. There does not appear to be any single policy that will be universally applicable. Hence the need for flexible access control models in this regard.

### The Safety Problem

In a seminal paper Harrison, Ruzzo and Ullman [8] proposed a simple language for stating the rules by which an access graph can be changed. The resulting model is often called HRU. They then posed the safety problem as follows<sup>1</sup>.

Given an initial access graph and a fixed set of rules for making authorized changes to it, is it possible to reach an access graph in which subject X has  $\alpha$  right to object Y (i.e., there is an edge labelled  $\alpha$  from X to Y)?

It turns out that safety is undecidable in the HRU model. Surprisingly the quest to find useful models with efficiently decidable safety proved to be quite challenging. Although significant positive results have appeared over the years, a appropriate balance between safety and flexibility remains a challenge for access control models. The role of graph theory in progress on the safety problem is discussed in section 2.

### Dynamic Hierarchies

Most practical access control systems go beyond the simple access graph we have discussed to provide a role (or group) construct. Thus subjects not only get the rights that they individually are granted but also acquire rights granted to roles (or groups) that they are a member of. For example, figure 4 shows an access graph in which subject U is a member of role G which in turn has the rights r and w for file F. Thereby U is authorized to read and write file F<sup>2</sup>.

Roles are a powerful concept in aggregating permissions and simplifying their administration [5, 26]. Modern access control systems are typically role-based because of the power and flexibility of this approach. Roles are often organized into hierarchies as shown, for example, in figure 5. This is a Hasse diagram of a partial order where senior roles are shown towards the top and junior ones towards the bottom. The Supervising Engineer role inherits all permissions of its junior roles, thus this role can do everything that the junior roles can do plus more. Conversely, a user who is a member of a senior role is also considered to be a member of the junior roles. In other words permissions are inherited upwards in the hierarchy and membership is inherited downwards. In practice role hierarchies need to change and evolve over time. How to do this effectively is a challenging problem for role administration. The application of graph transformations in progress on this issue is discussed in section 3.

---

<sup>1</sup> The original HRU formulation is in terms of the access matrix but is easily restated as done here in terms of the access graph.

<sup>2</sup> This can be shown in the access graph by a “temporary” edge labelled r, w directed from U to F.

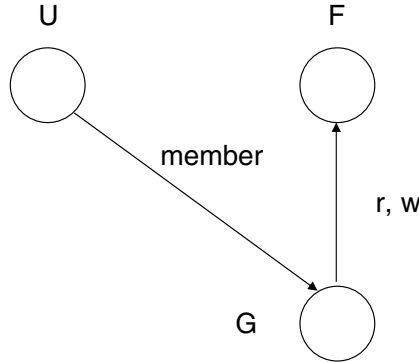


Fig. 4. Roles in Access Control.

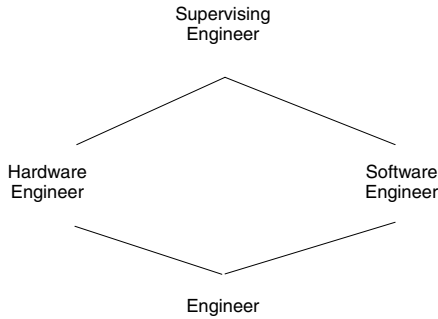
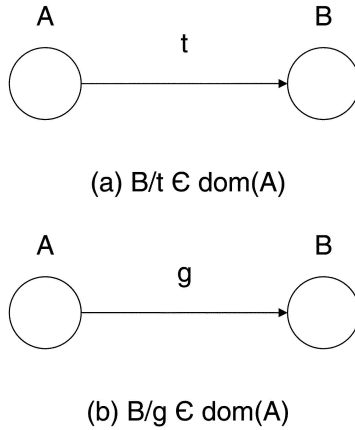


Fig. 5. An Example of Hierarchical Roles.

## 2 Graphs and the Safety Problem

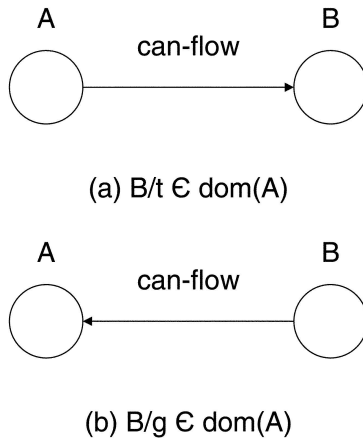
The take-grant model of Lipton and Snyder [15] is among the earliest applications of graph theory to access control. This model was developed in reaction to the undecidable safety results of the HRU model. It takes its name from the two rights it introduces,  $t$  for take and  $g$  for grant. The depiction of these two rights in the access graph is shown in figure 6. The notation  $B/t \in \text{dom}(A)$  denotes the possession of the  $B/t$  capability in the  $A$ 's domain, and is equivalent to stating that  $t \in [A, B]$  cell of the access matrix. Similarly for  $B/g \in \text{dom}(A)$ . The take right in figure 6(a) enables any right that  $B$  has to be copied to  $A$ . That is any edge originating at  $B$  can be duplicated with the same label and termination node but originating at  $A$ . The grant right in figure 6(b) conversely enables any right that  $A$  has to be copied to  $B$ . That is any edge originating at  $A$  can be duplicated with the same label and termination node but originating at  $B$ .

Somewhat surprisingly it turns out that the flow of rights in the take-grant model is symmetric. This allows for efficient safety analysis in the model but severely limits its expressive power. The original formulation of the take-grant model depicted the take and grant rights in the access graph as shown in figure 6.



**Fig. 6.** Transport of Rights in the Take-Grant Model: The Original Access Graph View.

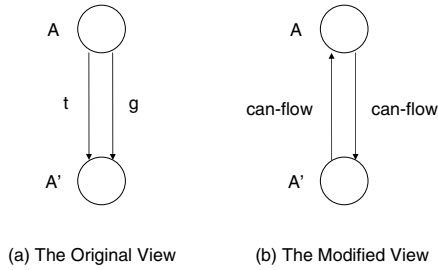
Lockman and Minsky [16] observed that a slightly different graph representation would demonstrate the symmetry of take-grant much more easily. They proposed to represent the ability for rights to flow from A to B by a directed edge from A to B labelled can-flow. The two situations of figure 6 are respectively shown in figure 7 in this modified representation. The focus of this representation is on the flow of rights rather than on the underlying right that enables the flow.



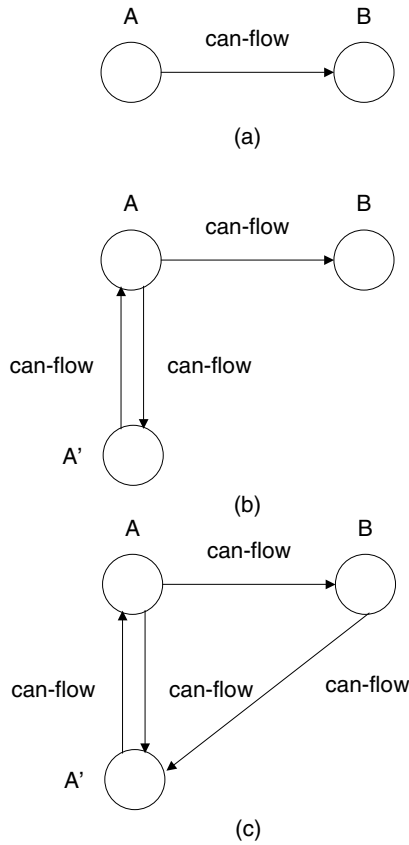
**Fig. 7.** Transport of Rights in the Take-Grant Model: The Modified Can-Flow View.

To complete description of the take-grant model we show the create operation shown in figure 8 using both styles of representation<sup>3</sup>. This diagram shows the

<sup>3</sup> The take-grant model also includes revoke and destroy operations. We omit their definition since they are not relevant here.



**Fig. 8.** Creation in the Take-Grant Model.



**Fig. 9.** Reversal of Flow in the Take-Grant Model.

result of A creating a new subject  $A'$ . A gets the  $A'/t$  and  $A'/g$  rights thus enabling can-flow in both directions.

With the modified can-flow representation symmetry of flow of rights in the take-grant model is easily demonstrated in figure 9. Figure 9(a) shows the initial situation with can-flow from A to B. This flow can be authorized by  $B/g \in \text{dom}(A)$

or by  $A/t \in \text{dom}(B)$ . The specific authorization is not material. The first step is for  $A$  to create  $A'$  as shown in figure 9(b) with the resulting can-flow edges from  $A$  to  $A'$  and vice versa. These edges are authorized by  $A'/t \in \text{dom}(A)$  and  $A'/g \in \text{dom}(A)$ . In particular,  $A'/g$  can be moved from  $\text{dom}(A)$  to  $\text{dom}(B)$  by virtue of the can-flow from  $A$  to  $B$ . This gives us the situation with can-flow from  $B$  to  $A'$  shown in figure 9(c). In conjunction with the can-flow from  $A'$  to  $A$  there is now a flow from  $B$  to  $A$ , thus reversing the original flow from  $A$  to  $B^4$ .

I came across these constructions during my doctoral research. I believe they demonstrate a fundamental truth. Graph representations are flexible and it is important to capture the correct properties in the edges and nodes of the graph. My subsequent work on the safety problem resulted in a number of models [1, 22, 24]. Some elements of graph theory are used in the analysis of these models but there is a probably a stronger tie with existing graph theory results. So there is potential for exploring a deeper connection here. Based on the discussion above much will depend upon a suitable representation of the access control problem in graph edges and nodes. Jaeger and Tidswell [6] have used graph notation to capture constraints and argue that this approach may lead to practical safety results. Nyanchama and Osborn have also discussed the representation of conflict of interest policies in their role-graph model [18]. Also recently Koch et al [9–12] have developed safety results directly based on the theory of graph transformations. Reconciliation of these results with the known safety results for access control models would be a step forward in understanding the insights that graphs and their transformations can provide in this domain.

### 3 Graphs and Dynamic Hierarchies

The use of role (or group) hierarchies in access control has a long history and considerable motivation for simplifying management of rights. The current commercial success of role-based access control products that support hierarchies is testimony to this fact. Mathematically a hierarchy is a partial order, that is a reflexive, transitive and anti-symmetric binary relation on roles. In this section we briefly look at two lines of research dealing with dynamic hierarchies for access control.

A particular kind of hierarchy called an ntree was introduced by this author [23]. The ntree has some very appealing properties for access control including the fact that it is a dimension 2 partial order, so it can be represented as the intersection of two linear orders. This allows us to label each node  $n$  with a pair of natural numbers  $l(n)$  and  $r(n)$ , such that  $u \leq v$  if and only if  $l(u) \leq l(v)$  and  $r(u) \leq r(v)$ . The ntree also has a recursive definition based on refining an existing node into another ntree, with the base case being a forest of trees and inverted trees. There are efficient algorithms for recognizing whether or not a given hierarchy is an ntree. One of the open questions regarding ntrees is how

---

<sup>4</sup> Lockman and Minsky [16] went on to consider the grant-only and take-only models with the former having the symmetric flow property of take-grant but the latter allowing asymmetric flow.



to recognize hierarchies that are close to being ntrees and could benefit from a ntree representation augmented with some additional information. More generally research on hierarchies with special properties that are attractive for access control can be pursued.

The need for dynamic role hierarchies as part of the administration of role-based access control (RBAC) is motivated in the ARBAC97 model [27]. The fundamental question is how to allow localized evolution of a hierarchy without disrupting larger global relationships. The authors of ARBAC97 suggest the notion of an encapsulated range as the basis for determining a suitable unit for local modifications. Crampton and Loizou [3, 4] point out some problems with this notion and propose a mathematically better founded notion of administrative scope. Koch et al [13] discuss administrative scope in their graph-based approach to access control and provide an operational semantics for it. The issue of evolving hierarchies and reconciling pre-existing hierarchies is likely to grow in importance as enterprises deploy RBAC across multiple business units and business partners.

## 4 Conclusion

In this paper we have briefly explored the connection between graphs and access control models, focusing on the safety problem and on dynamic role hierarchies. There is a long history of attempts to apply graph theory to these problems. Much of the earlier work is based on first principles. In recent years we have seen a more direct application of graph theory results. There is strong potential in further exploring this connection.

The area of access control and authorization has had a resurgence of interest in recent years. Although the access matrix model has served as a reasonable foundation for access control research and practice it has become considerably dated. With the Internet explosion many new forms of access control are being deployed in various e-commerce scenarios. There is increasing realization that the foundations of access control need a deeper and richer model. A number of authors have proposed various extensions to traditional access control. Park and Sandhu [19, 28] recently proposed a unified model for next generation access control called usage control or UCON. Initial efforts to formalize this model have taken a logic-based approach [30]. It would be interesting to see how graphs and their transformations can be applied to UCON models. The framework of UCON is very rich so there is likely to be some aspect of UCON that can benefit from a graph-based formal foundation.

## Acknowledgement

This material is based upon work supported by the National Science Foundation under Grant No. 0310776. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author and do not necessarily reflect the views of the National Science Foundation (NSF).

## References

1. Paul Ammann, Richard Lipton and Ravi Sandhu. The Expressive Power of Multi-Parent Creation in Monotonic Access Control Models. *Proc. IEEE Computer Security Foundations Workshop V*, Franconia, New Hampshire, June 1992, pages 148-156.
2. Paul Ammann, Duminda Wijesekera, and Saket Kaushik. Scalable, Graph-Based Network Vulnerability Analysis. *Proceedings CCS 2002: 9th ACM Conference on Computer and Communications Security*, pages 217-224, Washington, DC, November 2002.
3. Jason Crampton. Administrative Scope and Role Hierarchy Operations. *Proceedings of the Seventh ACM Symposium on Access Control Models and Technologies*, Monterey, California, 2002, pages 145-154.
4. Jason Crampton and George Loizou. Administrative scope: A Foundation for Role-Based Administrative Models. *ACM Trans. Inf. Syst. Secur.*, Volume 6, Number 2, pages 201-231, 2003.
5. David F. Ferraiolo, Ravi Sandhu, Serban Gavrila, D. Richard Kuhn and Ramaswamy Chandramouli. Proposed NIST Standard for Role-Based Access Control. *ACM Transactions on Information and System Security*, Volume 4, Number 3, August 2001, pages 224-274.
6. T. Jaeger and J. E. Tidswell. Practical safety in flexible access control models. *ACM Trans. on Info. and System Security*, 4(2), pages 158-190, 2001.
7. S. Jha, O. Sheyner and J. Wing. Two Formal Analyses of Attack Graphs. *Proceedings of the 15th IEEE Computer Security Foundations Workshop*, p.49-63, June 24-26, 2002.
8. Michael A. Harrison, Walter L. Ruzzo and Jeffrey D. Ullman. Protection in Operating Systems. *Commun. ACM*, 19:8, 461-471, 1976.
9. M. Koch, L.V. Mancini, and F. Parisi-Presicce. A Formal Model for Role-Based Access Control using Graph Transformation. In F.Cuppens, Y.Deswarte, D.Gollmann, and M.Waidner, editors, *Proc. of the 6th European Symposium on Research in Computer Security (ESORICS 2000)*, Lect. Notes in Comp. Sci. 1895, pages 122-139. Springer, 2000.
10. M. Koch, L.V. Mancini, and F. Parisi-Presicce. Decidability of Safety in Graph-Based Models of Access Control. In D.Gollmann, G.Karjoth, and M.Waidner, editors, *Proc. of the 7th European Symposium on Research in Computer Security (ESORICS 2002)*, Lect. Notes in Comp. Sci. 2502, pages 229-243. Springer, 2002.
11. M. Koch, L.V. Mancini, and F. Parisi-Presicce. A Graph Based Formalism for RBAC. *ACM Trans. on Info. and System Security*, 5(3):332-365, 2002.
12. M. Koch and F. Parisi-Presicce. Describing Policies with Graph Constraints and Rules. In A. Corradini, H. Ehrig, H.-J. Kreowski, and G. Rozenberg, editors, *Int. Conference on Graph Transformations*, Lect. Notes in Comp. Sci. 2505, pages 223-238. Springer, 2002.
13. M. Koch, L. V. Mancini and F. Parisi-Presicce. Administrative Scope in the Graph-Based Framework. *Proceedings of the ninth ACM Symposium on Access control Models and Technologies*, Yorktown Heights, New York, pages 97-104, 2004.
14. Lampson, B.W. Protection. *5th Princeton Symposium on Information Science and Systems*, pages 437-443, 1971. Reprinted in *ACM Operating Systems Review* 8(1):18-24, 1974.
15. R.J. Lipton and L. Snyder. A Linear Time Algorithm for Deciding Subject Security. *Journal of the ACM*, Volume 24, Number 3, pages 455-464, 1977.

16. Lockman, A. and Minsky, N. Unidirectional Transport of Rights and Take-Grant Control. *IEEE TSE*, Volume SE-8, Number 6, pages 597–604, 1982.
17. J. P. McDermott. Attack Net Penetration Testing. *Proceedings of the 2000 workshop on New Security Paradigms*, Ballycotton, County Cork, Ireland, pages 15–21, 2000, ACM Press.
18. M. Nyanchama and S.L. Osborn. The Role Graph Model and Conflict of Interest. *ACM Trans. on Info. and System Security*, 1(2):3–33, 1999.
19. Jaehong Park and Ravi Sandhu. The UCON<sub>ABC</sub> Usage Control Model. *ACM Transactions on Information and System Security*, Volume 7, Number 1, February 2004, pages 128–174.
20. C. Phillips and L. Swiler. A graph-based system for network vulnerability analysis. *ACM New Security Paradigms Workshop*, pages 71–79, 1998.
21. Michael K. Reiter and Stuart G. Stubblebine. Authentication Metric Analysis and Design. *ACM Trans. Inf. Syst. Secur.*, Volume 2, Number 2, pages 138–158, 1999.
22. Ravi Sandhu. The Schematic Protection Model: Its Definition and Analysis for Acyclic Attenuating Schemes. *Journal of the ACM*, Volume 35, Number 2, April 1988, pages 404–432.
23. Ravi Sandhu. The NTree: A Two Dimension Partial Order for Protection Groups. *ACM Transactions on Computer Systems*, Volume 6, Number 2, May 1988, pages 197–222.
24. Ravi Sandhu. The Typed Access Matrix Model. *Proc. IEEE Symposium on Research in Security and Privacy*, Oakland, California, May 1992, pages 122–136.
25. Ravi Sandhu and Pierangela Samarati. Access Control: Principles and Practice. *IEEE Communications*, Volume 32, Number 9, September 1994, pages 40–48.
26. Ravi Sandhu, Edward Coyne, Hal Feinstein and Charles Youman,. Role-Based Access Control Models. *IEEE Computer*, Volume 29, Number 2, February 1996, pages 38–47.
27. Ravi Sandhu, Venkata Bhamidipati and Qamar Munawer. The ARBAC97 Model for Role-Based Administration of Roles. *ACM Transactions on Information and System Security*, Volume 2, Number 1, February 1999, pages 105–135.
28. Ravi Sandhu and Jaehong Park. Usage Control: A Vision for Next Generation Access Control. *Proc. Computer Network Security: Second International Workshop on Mathematical Methods, Models, and Architectures for Computer Network Security, MMM-ACNS*, Saint Petersburg, Russia, September 21–23, 2003, Springer-Verlag Lecture Notes in Computer Science 2776, pages 17–31.
29. Oleg Sheyner, Joshua Haines, Somesh Jha, Richard Lippmann and Jeannette M. Wing. Automated Generation and Analysis of Attack Graphs. *Proceedings of the 2002 IEEE Symposium on Security and Privacy*, p. 254–265, May 12–15, 2002.
30. Xinwen Zhang, Jaehong Park, Francesco Parisi-Presicce and Ravi Sandhu. A Logical Specification for Usage Control. *Proc. 9th ACM Symposium on Access Control Models and Technologies (SACMAT)*, New York, June 2–4, 2004, pages 1–10.