MULTI-DOMAIN TRUST MANAGEMENT IN VARIABLE-THREAT ENVIRONMENTS USING ROLLBACK-ACCESS

Mike Burmester
Florida State University
Tallahassee, FL

Prasanta Das, Martin Edwards
The MITRE Corporation
McLean, VA

Alec Yasinsac
University of South Alabama
Mobile, AL

ABSTRACT

Trust Management systems are trust infrastructures that support authorization for security-critical actions in decentralized environments. In this paper we describe an extension suitable for multi-domain applications in variable-threat environments that allows for temporary adjustments of trust levels in response to elevated threat levels, and which can be reversed without compromising actions that took place during such periods—we term this, rollback-access. We argue that a rollback-access capability is an essential feature for security-critical applications, and propose a working prototype for an agent based implementation.

1. INTRODUCTION

In support of implementing Coalition Information Sharing (CIS) capabilities, the need exists to establish trust among members of the network. These trust relationships may be ad hoc and dependent on validation of identity of one member by others within a trust community. To meet this need requires the U.S. to establish trust relationships with coalition partners.

The paper will investigate how a new capability, that we term rollback-access (RA), can manage trust for increased or decreased functionality across the Global Information Grid in support of multinational information sharing in a net-centric environment. The goal is to establish dynamic and flexible trust establishment mechanisms for complex coalition environments.

The approach is a fundamental change in the nature of trust. In most existing trust systems, trust is binary and static; an entity is either trusted or untrusted. Multi-level trust systems extend the binary model, but reflect a simple extension of a two valued trust system to a discrete sized, static trust value set, providing only minor additional functionality. Moreover, they retain all the limitations of static, two-valued trust systems, such as the difficulty of dealing effectively with conflicting trust information. Coalition trust environments have inherent complexities that cannot be accurately captured in fixed value trust systems.

A mobile code-based trust management system based on a variable trust valued model more nearly reflects the myriad of subtleties that characterize modern coalitions and consortiums (e.g., trust, mistrust, malicious hosts, insider attacks, passive adversaries, sleeper cells, etc.) This paper will present an approach to develop a working prototype of a dynamic trust system with RA functionality for distributed systems that support mobile code and several operational scenarios and demonstration environments that highlight the selected trust system features.

Background. There is extensive work in the literature on modeling access control and trust management (TM) systems. Access control can be discretionary, mandatory or role-based [3, 4, 13, 20]. Early work on TM systems involved analyzing their structures (e.g., [15, 1, 16]), while later work focused on decentralized trust management [6, 7], and on designing flexible systems appropriate for open network applications (e.g., [21, 2]). Recently several papers address implementation issues (e.g., [12, 14, 19, 8]).

TM systems such as KeyNote [5] and SPKI/SDSI [11, 10] use credentials to delegate permissions. Role-based TM's such as RT [18, 17] combine the flexibility of role-based access control (RBAC) with the strength of TM.

Our contribution. In today's environment of asymmetric warfare and homeland security, the formation of coalition partnerships among governmental and non-governmental organizations within United States as well as U.S. collaboration with international partners is essential. The premise of such information sharing is the need-to-share security critical information among ad hoc domains. This sharing is based on trust policies that determine the internal or external trust value sets which enable the two sides to establish trust so they can interact with each other.

The trust relations among the ad hoc coalition partners introduce additional security uncertainty based on changing external and internal threat levels. Since the coalition

networks may contain multiple domains the ability to share information across these domains is paramount. The information trustworthiness will depend on the security implementations within each domain. The dynamic trust model is planned to be able to adjudicate between these varying domains and broker the appropriate access based on the perceived and calculated threat levels. Therefore, a dynamic trust model based on a static set of criteria is less applicable and prone to security vulnerability. We present a dynamic trust model, i.e., temporary adjustment or denial of information with rollback access capability not based on a pre-defined static role-based access control or attribute-based access control mechanisms, but adding to these types of access control the parameter of threat levels and location (for an illustration see Fig. 2).

2. SCENARIOS

To motivate our methodology and the rollback-access functionality we consider two scenarios. The first scenario stresses the need for a dynamic, open-door functionality, while the second underlines the intricacies of managing rollback-access.

2.1. Scenario A

The U.S. President through the Department of Homeland Security (DHS) and the Office of the Director of National Intelligence (DNI) has established a policy where counterterrorism information is to be shared to the greatest extent possible. The point of the policy is to ensure all participants in the national counterterrorism effort are provided the most accurate and current information available.

Operational environment. Key to the implementation of this policy is the reality that establishing a single network that all the responsible agencies within federal, state, local and tribal organizations is too expensive and will take too long. Therefore the implementation guidance stresses the need for all data/information producers and owners to instantiate "open door" capabilities to their networks and data stores. The policy does not call for wholesale exposure of data and information, but for open visibility of data and information to those needing the information to execute their day-to-day mission.

Trust Model. The intelligence community is ready to embrace this policy. It is, however, looking for reassurance that each agency ensures that the information within that agency is authorized to be received by those who have been granted need-to-share based on established credentials. If the DHS and DNI are able to determine that a receiver of information is not providing the appropriate protection and/or changes in credentials to effect need-to-share privileges, then that federal, state, local or tribal entity's access will be reduced to the minimum allowed. Once the entity has reestablished the appropriate protections, then their trust can be *incrementally increased* until they return to full open access (rollback). The monitoring of this trust must be dynamic and able to respond to the changing needs of large and small organizations.

2.2. Scenario B

A coalition of 12 national militaries (e.g., U.S., Germany, Belgium, France, U.K. etc), governmental agencies (e.g., DOS, DOE, CIA) and non-governmental organizations (NGOs) (e.g., Médecins Sans Frontières, American Red Cross, UNICEF, Red Crescent) are involved in a stabilization and humanitarian relief effort in Orange Land, a sub-Sahara nation, in the mist of inter-tribal conflict and a three year drought. The Orange Land government is generally pro-west, but there are at least two factions within the government that have ties to terrorist organizations through their rhetoric and tribal affiliations.

Operational environment. A tactical wide area network was established to support the coordination and cooperation in all facets of the operations. As such, each national military and governmental agency as well as the NGOs is on the network with common access based on attributes associated with the group. The military consistently presents information on insurgent locations and dangerous areas (e.g., improvised explosive device (IED) locations) to allow non-military group use of the information for safety and planning. Additionally, the military provides time-lines for general operations that will go force-on-force with insurgents to ensure the non-military efforts are not caught up in these operations, which could result in civilian casualties.

Key to the level of information sharing provided is the trust established between the organizations that information would be available to each group but groups would not share between themselves the information. Thus each organization has its own 'information compartment' which prevents cross-talk, but allows for coordinated approaches to resolving issues. Over the last months this trust relationship has allowed the military to successfully eliminate a number of insurgent strongholds and clearly map the IEDs planted. Most of the IEDs were destroyed, but some are still in areas too 'hot' to get into, but the military is planning operations to solve that.

A trust problem. It has become apparent to the military organization that the information being posted about upcoming efforts is being leaked to insurgents. The last three

operations have the military arriving within minutes of the insurgents departing and often to find a number of IEDs and other traps established. The military is considering removing all non-military organizations from the network, but sees that as too extreme an action which could lead to unnecessary endangerment of humanitarian efforts.

Rollback-access capability. The decision is to reduce the trust management level for the Orange Land government as well as for those NGOs that are overtly sympathetic to the insurgents. The expectations are that each organization is remaining in its 'information compartment' and thus will not notice that the granularity, frequency or fidelity of the information provided has changed. As the trust of organizations is rebuilt, the military will allow greater information flow into the appropriate organizational compartments while further reducing that of less trustworthy organizations.

Information compartments. A major challenge is how to deal with information in organizational compartments. Ideally, information in a compartment should be redacted in a controlled way, to allow for granularity. However it is important that there is separation between the different instantiations of information compartments: e.g., it should not be possible to write to earlier instantiations unless/until the trust level justifies this.

3. OUR APPROACH

3.1. Vanilla-rollback-access: a state of suspension

Our approach is based on, and extends [9], which describes a human-centric TM model with vanilla-rollback-access (VRA). In [9], trust is supported by peer communities and exploits the effectiveness that humans have in understanding their roles in their communities. This model recognizes that authentication is scalar rather than Boolean.

If a user, say Alice, has forgotten her password or pin, the network system will still allow her access to some basic services—*vanilla access*, for short periods. The system contacts her peer-community and if sufficient trust is mustered then Alice will get full access. An important enabling feature of VRA is that if vanilla-Alice logs out before successful authentication is accomplished, the session manifestation can be maintained in a suspended state, neither committed nor discarded. If the questionable session is later authenticated, all manifestations can be triggered and the system state updated as though the actions were taken at the time they were initiated by vanilla-Alice. Conversely, if an impersonation attempt is recognized, vanilla-rollback (restore mode) can revert the system to its original state, essentially rolling back all changes that vanilla-Alice performed. VRA is effectively a human-centric escrow recovery mechanism.

Should we trust Alice? Alice is highly reliable: her contribution is pivotal to the operation of our system. But she is known to be on occasion careless. Should we trust her?

In many security-critical applications we may have to work in such environments. We therefore seek a flexible trust infrastructure with a rollback-access functionality that is not necessarily triggered by a forgetful Alice,[1] but by security alerts, or more generally, intelligence—usually of a temporal and/or locational nature.

3.2. Our model

We build our model on a trust management system that provides adequate flexibility: e.g., a TM based on credentials such as KeyNote [5] and SPKI/SDSI [10], or roles such as RT [18]. For our purpose it is sufficient that the trust will support our additional functionality.

TM systems provide a unified approach in specifying and interpreting security policies, credentials and relationships. Their functionality is to *authorize* actions of entities (individuals or processes). We denote by TM^{auth} the authorization functionality specified by the TM system and say that TM^{auth} *realizes* TM.

In our model, the functionality TM^{auth} is restricted by the threat level (or more generally, intelligence) that applies when it is invoked. If θ is the threat level, the restricted functionality is denoted by TM^{auth}_{θ}. Threat levels can be local or global, and may be linear or non-linear. The DHS uses a threat model with four threat levels which is a global, linearly ordered set. In general, threat level systems are modeled by partially ordered sets $(\Theta, \succeq)$.

We denote the set of trust management systems that Θ induces on TM by,

$$\mathrm{TM}_{\Theta} = \{\mathrm{TM}_{\theta}\}_{\theta\in\Theta},$$

and call it, a *multi-domain* trust management system with *rollback* access (R-TM). TM_{Θ} is realized by the functionalities: $\mathcal{TM}^{auth}_{\Theta} = \{TM^{auth}_{\theta}\}_{\theta\in\Theta}$.

There is a natural dominance relation "$\succeq_{auth}$" in $\mathcal{TM}^{auth}_{\Theta}$, for which: $TM^{auth}_{\theta_1} \succeq_{auth} TM^{auth}_{\theta_2}$, if every action that is authorized by $TM^{auth}_{\theta_2}$ is also authorized by $TM^{auth}_{\theta_1}$. In our model we link the threat level order to the TM dominance by requiring that these be inversely related:

$$\theta_1 \succeq_{th} \theta_2 \quad \Rightarrow \quad TM^{auth}_{\theta_2} \succeq_{auth} TM^{auth}_{\theta_1}. \qquad (1)$$

Consequently by lowering the threat level, authorization is extended until eventually it is fully restored. Conversely by

[1] Although such a functionality can be useful in our threat model.

raising the threat level, authorization is restricted until eventually it is reduced to vanilla.

3.3. Rollback-access

Assign to each access action α an *access threshold threat* value $\theta(\alpha)$: $\theta(\alpha)$ is the highest threat level at which the action α is authorized, independently of the authorization of the underlying TM system.

When the threat level θ is raised to θ^+, rollback-access (RA) is triggered and the functionality $TM^{auth}_{\theta^+}$ is invoked: actions that are executed while the threat level is raised, and which are not authorized by the new functionality, get suspended (rollback: withdrawal mode) and a record of their partially executed state is temporarily stored (for later retrieval). What characterizes RA is that: (i) it is suspended, (ii) transitory, (iii) segregated, and (iv) can rollback (restore mode) an authorization to a lower threat level.

To capture this we introduce the concept of an *information compartment* (IC). ICs are defined for threat levels θ, and are denoted by IC_θ. IC_θ is a (logical) memory block in which are stored records of partially executed actions that get unauthorized when the threat level is raised to the next level above θ. In particular, when the execution of an action α is suspended because the threat level is raised from θ to θ^+, then a record of its suspended state is stored in $IC_{\theta(\alpha)}$. Note that in general several actions may be suspended when the threat level is raised, so several 'intermediary' $IC_{\theta(\alpha)}$ may be involved, with $\theta^+ \succ_{th} \theta(\alpha) \succeq_{th} \theta$.

If the threat level is later lowered to θ, then the TM system will rollback (restore mode) all those records of suspended executions of access actions α in $IC_{\theta(\alpha)}$ that get authorized by the new functionality. We describe these two actions in more detail below.

Initially $IC_\theta \leftarrow \emptyset$ for all $\theta \in \Theta$.

Rollback of IC_θ: θ is raised to θ^+

1. Put in $IC_{\theta(\alpha)}$ a record of every suspended access action α. Note that there may be several actions that get suspended, so this may involve several information compartments $IC_{\theta(\alpha)}$: $\theta^+ \succ_{th} \theta(\alpha) \succeq_{th} \theta$.
2. Invoke the functionality $TM^{auth}_{\theta^+}$.
3. Every object β produced while the threat level is θ^+ is assigned the threat value θ^+ (in addition to the classification of the underlying TM system).

Rollback of IC_θ: θ is lowered to θ^-

1. All the records in the information compartment IC_{θ^*}: $\theta \succeq_{th} \theta^* \succeq_{th} \theta^-$, that are authorized by the new functionality $TM^{auth}_{\theta^-}$ get restored: they get labeled as objects with threat value θ^*, and removed from IC_{θ^*}.
2. Invoke the functionality $TM^{auth}_{\theta^-}$.
3. Every object β produced while the threat level is θ^- is assigned the threat value θ^- (in addition to the classification of the underlying TM system).

3.4. Architecture

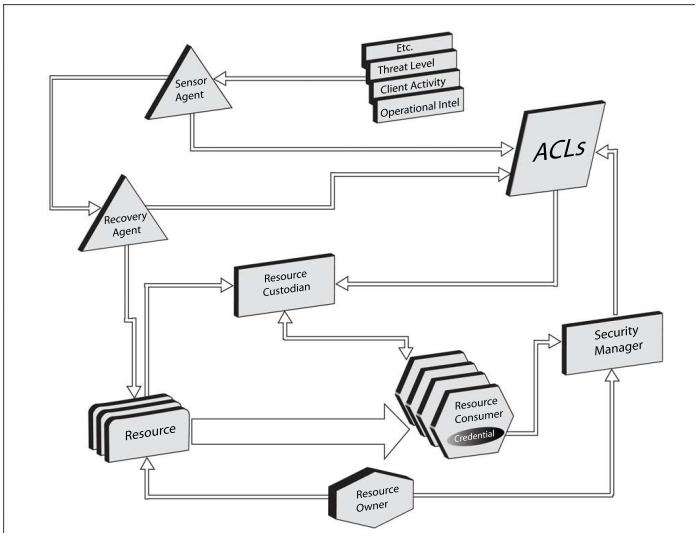

Figure 1: An architecture for an agent based TM with rollback-access functionality.

The architecture of an agent based TM with rollback-access functionality consists of:

- A Security Manager
- A Sensor Agent and a Recovery Agent,
- Environmental inputs (e.g., threat level, client activities, operational intelligence, etc.)
- A Resource Custodian
- Access Control Lists (ACLs)
- Resource Owners, Resources, and Resource Consumers.

Existing trust systems use the two dimensions of credentials and access requirements whereas TM with rollback access brings in the third dimension of threat as a parameter.

While our model can reflect many different sensor notions, in this paper, the Sensor Agent is represented by the four DHS threat levels. Based on the environmental conditions, the Sensor provides input to the ACL system in terms of the DHS threat level. The Security Manager coordinates with the resource owners to properly represent desired resource

security properties in terms of entity credentials and in the ACL system. The Resource Custodian makes access decisions based on the credentials presented, the ACL system, and the threat level. The Resource Custodian provides the resource or access to the resource to the client.

Operationally, the sensor-based TM executes similarly to the traditional Bell-LaPadula model. When DHS elevates the threat level due to unauthorized activities or other environmental conditions, the Sensor Agent modifies the corresponding ACLs according to the new restrictions that come into play for each of the resources and the resource consumers. The Resource Custodian then starts to rollback (withdraw mode) access from those resource consumers that do not have the authority to work with resources previously open to them. This rollback of access may include documents and data objects that the resource consumer was an author of and as such they are not able to modify and possibly even read the object at this heightened threat level. In those instances where operations must continue a new time stamped version of the data object may be established which allows modification within the new threat environment.

During this time of heightened threat level, the Recovery Agent is assessing the status established by the Sensor Agent to determine whether conditions have changed such that resource consumers can have their access rollback to its original openness. As the environment returns to "normal operations" the threat level decreases and the Sensor Agent returns to the resource consumer (rollback, restore mode) the visibility and modification rights previously enjoyed. Additionally, the Recovery Agent will assess whether the information introduced on the new time stamped version of the data object is valid and acceptable to be consumed by the earlier data object. Thus work done during the heightened threat environment is preserved but also adjudicated prior to wholesale acceptance as valid.

4. A WORKING PROTOTYPE

The prototype uses a TM system with the DHS threat level set $\Theta = \{\mathrm{Green}(G), \mathrm{Blue}(B), \mathrm{Orange}(O), \mathrm{Red}(R)\}$, with $R \succ O \succ B \succ G$ (where $\succ$ indicates "greater than"). We only discuss the additional RA functionality.

4.1. Rollback-access

Assign to each object α an access threat value $\theta(\alpha)$, that reflects its vulnerability to external/internal threats.

1. If an object β is produced as a result of action α (e.g., the access mode is $\underline{\mathrm{w}}$ (write), $\underline{\mathrm{a}}$ (append), or $\underline{\mathrm{e}}$ (execute)), then β is assigned the threat value of α, and the classification $\theta(\alpha)$.
2. If the threat level θ rises above $\theta(\alpha)$ while α is executed this action is suspended: an object $suspend(\alpha)$ is generated and assigned the threat value $\theta(\alpha)$, and the classification of α.
3. If (later) the threat level θ drops below $\theta(\alpha)$, then access to $suspend(\alpha)$ is restored, and its execution can be completed, provided this is authorized by TM. (e.g., by its owner, or anybody assigned access by the owner).

4.2. Compatibility

We require that the threat values and the classification levels of objects be inversely related. That is, the higher (lower) the threat value of an object β, the lower (higher) the classification level of β. The justification for this is that the RA capability is intended to support the security of the underlying trusted information system (which is based on controlling information flows—the simple security property [3]).

4.3. Example

Suppose that the threat level is $\theta = B$, that the threat value for action α is $\theta(\alpha) = B$ and that Alice has authorized TM access to α. Then Alice has R-TM access to α.

If the threat level is (later) raised to $\theta = O$ while α is executed, then this action is suspended: an object $suspend(\alpha)$ is generated with threat value O and classification level that of α, which defines the state of the partially executed action. Now Alice cannot access α, nor its partly executed state (for example, if she was writing a report regarding insurgent activities in Orange Land this report is suspended), even if the TM functionality allows α: the threat level O overrules this.

For Bob, α is not TM-authorized (he doesn't have discretionary access). He cannot access α even if the threat level θ is G.

4.4. Threat access control

There are three levels at which an access action α has to be authorized:

- the *discretionary* level,
- the *mandatory* level, and
- the *threat* level.

The first two define the functionality of the TM system. The last defines the extended functionality proposed in this paper. Threat access control is temporal and locational, and determined by the relation between the threat level θ

at the time and place[2] of the action α and the threshold value $\theta(\alpha)$. We refer to this authorization as *threat-level* (tl)-authorization. We have:

Simple threat-level (stl) property:

- *If $\theta(\alpha) \succeq \theta$, then the action α is* tl-authorized *and the suspended state of any incomplete α-instantiation is* tl-restored.[3]
- *If $\theta \succ \theta(\alpha)$, then the action α is not tl-authorized, and any incomplete α-instantiations that are not already tl-suspended, will get suspended and assigned the threat threshold value $\theta(\alpha)$ (and possibly a suspend bit [4]).*

The stl-property is a counterpart of the ss-security (simple security) property of the Bell-LaPadula model [3]. In our case it is used to protect objects in variable-threat environments. As in [3] it will protect objects (information containers) rather than contents (the information itself). In Bell-LaPadula a *-property is used to protect information flows. Our model assumes a secure TM infrastructure, and in particular the Bell-LaPadula security requirements: consequently it inherits this level of security.

The easiest way to show this is through an illustration. We use the threat level model in Fig. 2. Suppose that an entity in U.S. has write R-TM access to an object α that was generated by an entity in Orange Land in which a sudden change of the threat level there prevents its completion. Suppose that the task is completed in U.S. and let β be the resulting object. Then by the TM-functionality requirements, α, β have the same classification, and by our requirements in Section 4.1, they have the same threat value $\theta(\alpha) = \theta(\beta)$. This prevents "illegal" information flows.

5. The way ahead

There are several areas in which research on TM systems with rollback access shows promise. Below we highlight three such areas:

1. *The structure of the threat level system.* In this paper we have focused on a global, linear structure. Local structures that address issues such as, the threat level in Orange Land is different from that in U.S., capture more fully the scenarios described in the Introduction—see Fig. 2. Observe that if an action is suspended in Orange Land because of a sudden increase in the threat level there, it may be possible to complete it in U.S. where the threat level may be lower.

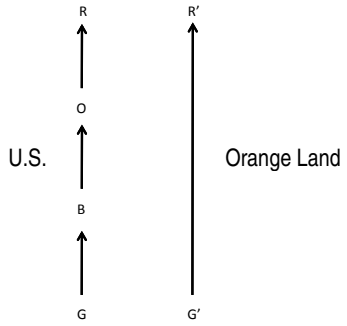

Figure 2: A basic threat level structure for Scenario B: when the threat level in Orange Land is raised, access is reduced to minimal, while in the U.S. raising it from G to B is less restrictive.

2. *The impact of threat dominance on the functionality of TM systems.* We have not discussed how this works, other than require that it is inversely proportional to the classification levels: in particular that an increased threat level will not support additional functionality (Section 3.2, end). In general an elevated threat level should affect differently the commander in chief from a field worker. So the relation between threat levels and classification levels need not be smooth. For example, we may use a threat model for which the threat level for the commander in chief is always low (G). Alternatively we may link threat values to clearance levels.
3. *Extending the trust model to allow for a human-centric functionality.* By the nature of the effort being exploratory, we anticipate demonstrating the feasibility of the approach through developing a prototype and initiating a set of indicators of dynamic threat levels. Through the process of development the threat indicators will be formalized and attributed with greater granularity.

6. Conclusion

Access control and trust management are the basic components of a trusted information system. In this paper we propose a new access control mechanism that supports a more flexible approach to trust management. This mechanism is triggered by threat levels: when the threat level is raised beyond a certain threshold, processes, or partially completed processes, may be suspended. Later, when the

[2]This refers to the topology of the network, and is not necessarily geographical.

[3]Full authorization/restoration requires that the TM access requirements are also satisfied.

[4]This is for an additional functionality: when a suspended object is later restored, all entities that get access to it will be notified.

threat subsides, these are restored thus providing a rollback access functionality.

References

[1] Martín Abadi, Michael Burrows, Butler Lampson, and Gordon Plotkin. A calculus for access control in distributed systems. *ACM Transactions on Programming Languages and Systems*, 15(4):706–734, September 1993.

[2] Lujo Bauer, Michael A. Schneider, and Edward W. Felten. A General and Flexible Access-Control System for the Web. In *Proceedings of the 11th USENIX Security Symposium*, San Francisco, CA, August 2002.

[3] D. Elliott Bell and Leonard J. LaPadula. Secure Computer Systems: Mathematical Foundations. Technical Report TR #2547, MITRE Corporation, 1973.

[4] K.J. Biba. Integrity Considerations for Secure Computer Systems. Technical Report TR #3153, MITRE Corporation, 1977.

[5] Matt Blaze, Joan Feigenbaum, John Ioannidis, and Angelos Keromytis. KeyNote Trust Management system, Version 2. IETF RFC 2704, September 1999, ftp://ftp.isi.edu/in-notes/rfc2704.txt.

[6] Matt Blaze, Joan Feigenbaum, and Angelos D. Keromytis. The role of trust management in distributed systems security. In *Secure Internet Programming*, pages 185–210, 1999.

[7] Matt Blaze, Joan Feigenbaum, and Jack Lacy. Decentralized trust management. In *IEEE Symposium on Security and Privacy*, pages 164–173. IEEE Computer Society, 1996.

[8] Matt Blaze, John Ioannidis, and Angelos D. Keromytis. Experience with the KeyNote Trust Management System: Applications and Future Directions. In Paddy Nixon and Sotirios Terzis, editors, *iTrust*, volume 2692 of *Lecture Notes in Computer Science*, pages 284–300. Springer, 2003.

[9] Mike Burmester, Breno de Medeiros, and Alec Yasinsac. Community-Centric Vanilla-Rollback Access, or: How I Stopped Worrying and Learned to Love My Computer. In Bruce Christianson, Bruno Crispo, James A. Malcolm, and Michael Roe, editors, *Security Protocols Workshop*, volume 4631 of *Lecture Notes in Computer Science*, pages 228–237. Springer, 2005.

[10] Dwaine E. Clarke, Jean-Emile Elien, Carl M. Ellison, Matt Fredette, Alexander Morcos, and Ronald L. Rivest. Certificate Chain Discovery in SPKI/SDSI. *Journal of Computer Security*, 9(4):285–322, 2001.

[11] C. Ellison, B. Frantz, B. Lampson, R. Rivest, B. Thomas, and T. Ylonen. SPKI Certificate Theory. IETF RFC 2693, September 1999, ftp://ftp.isi.edu/in-notes/rfc2693.txt.

[12] Carl A. Gunter and Trevor Jim. Policy-directed certificate retrieval. *Softw., Pract. Exper.*, 30(15):1609–1640, 2000.

[13] M. A. Harrison, W. L. Ruzzo, and J. D. Ullman. Protection in Operating Systems. *Communications of ACM*, 19(8):461–471, 1992.

[14] Trevor Jim. SD3: A Trust Management System with Certified Evaluation. In *IEEE Symposium on Security and Privacy*, pages 106–115, 2001.

[15] Butler Lampson, Martín Abadi, Michael Burrows, and Edward Wobber. Authentication in distributed systems: Theory and practice. *ACM Transactions on Computer Systems*, 10(4):265–310, 1992.

[16] Ninghui Li, Joan Feigenbaum, and Benjamin N. Grosof. A logic-based knowledge representation for authorization with delegation. In *CSFW*, pages 162–174, 1999.

[17] Ninghui Li and John C. Mitchell. RT: A Role-based Trust-management Framework. In *DISCEX (1)*, pages 201–212, 2003.

[18] Ninghui Li, John C. Mitchell, and William H. Winsborough. Design of a Role-Based Trust-Management Framework. In *IEEE Symposium on Security and Privacy*, pages 114–130, 2002.

[19] Ninghui Li, William H. Winsborough, and John C. Mitchell. Distributed credential chain discovery in trust management. *Journal of Computer Security*, 11(1):35–86, 2003.

[20] Ravi S. Sandhu, Edward J. Coyne, Hal L. Feinstein, and Charles E. Youman. Role-Based Access Control Models. *IEEE Computer*, 29(2):38–47, 1996.

[21] Stephen Weeks. Understanding trust management systems. In *IEEE Symposium on Security and Privacy*, pages 94–105, 2001.