An Attribute Based Framework for Risk-Adaptive Access Control Models

Savith Kandala
Principal
SRA-Touchstone Consulting Group
www.touchstone.com
Email: savith.kandala@touchstone.com

Ravi Sandhu
Executive Director and Endowed Professor
Institute for Cyber Security
University of Texas at San Antonio
Email: ravi.sandhu@utsa.edu

Venkata Bhamidipati
Vice President
MiCore Solutions
www.micoresolutions.com
Email: vbhamidi@gmualumni.org

***Abstract*—The concept of risk-based adaptive access control (RAdAC, pronounced Raid-ack) has been recently introduced in the literature. It seeks to automatically (or semi-automatically) adjust security risk for providing access to resources accounting for operational needs, risk factors and situational factors. In order to make progress in this arena we need abstract models analogous to those that underlie the sustained and successful practice of discretionary, mandatory and role-based access control. Such models define a formal structure and components for policy specifications, while allowing for a variety of enforcement architectures and detailed implementation. In this paper we develop a novel approach to capture these characteristics of RAdAC using attribute-based access control. We further show that this RAdAC model can be expressed in the UCON usage control model with suitable extensions, and discuss how other UCON elements not used in this construction could beneficially improve the RAdAC vision.**

***Index Terms*—Access control, Risk-Adaptive Access Control, RAdAC, Risk-Based access control, Usage Control, UCON**

I. Introduction and Motivation

Risk Adaptable Access Control (RAdAC) is an emerging concept in access control, conceived in context of modern large-scale computing environments such as the US Department of Defense Global Information Grid (GIG). The vision for such systems is a globally interconnected, end-to-end set of information capabilities for collecting, processing, storing, disseminating, and managing information on demand [15], [16]. This requires a dynamic balance between the need to access information in view of mission priorities, risk and cost of information compromise, and overall operational and threat status of the system. Similarly, in the commercial sector businesses are increasingly providing access to information and services over multiple platforms while adjusting the level of access and consequent risk. Today this is often done on a relatively static basis, such as whether or not a particular personal computer (PC) has previously interacted with the system for a given user, or whether user access is from a smartphone or PC. In applications such as healthcare, dynamic adjustments can improve service delivery especially in emergency situations. In the bigger consumer space businesses can provide additional functionality and conveniences to their customers through RAdAC.

In general, the risk of granting a specific access to a user varies depending on the type of access or transaction being requested by the user and the environment from which the user is requesting it. For example, the one-time display of a classified document on a secure computer terminal to a cleared individual in a highly physically secure facility is inherently less risky than providing that same individual the ability to copy that document on to removable media or providing the same individual the ability to print that document [13]. Similarly, allowing a user to make a payment from a PC that the user has previously used during hours that are normal for that user is inherently less risky than allowing the same transaction from an unknown PC. RAdAC brings this concept of risk, from all the components used for access, into the access control decision process. In addition, it also brings the concepts of *operational need* and *situational factors* into the access control decision process. The concept of operational need is usually addressed as need-to-know in the literature and can be represented at a coarse grain as a person's membership in some community of interest or an organization. Situational factors are defined as the environmental or external conditions under which the access decision is being made.

The concept of RAdAC was introduced in [12] wherein the impediments of traditional access control approaches to sharing of information and the main conceptual characteristics of RAdAC were discussed, but without articulation of a precise formal model. The MITRE Report [13] recommends focus on risk and proposes three guiding principles: measuring risk, establishing acceptable levels of risk and ensuring that information is accessible at acceptable risk levels. Cheng et al [5] introduce methods that can be used to quantify risk associated with information access, and give a case study for implementing a multilevel security access control model (Fuzzy MLS). Ni et al [14] further builds on the Fuzzy MLS example by introducing risk estimations and fuzzy inferences for risk-based access control models.

The main contribution of this paper is to specify a formal framework in terms of the components and their interactions to develop abstract models for RAdAC. The models proposed in this paper are at the policy layer, and do not lay out enforcement architectures and implementation details [10], [23]. We believe that such abstract models are needed to make progress in this area similar to those that underlie the successful practice of discretionary, mandatory and role-based

access control (RBAC) models. For example consider the abstract models for RBAC presented in [22] which inspired a sizable literature based on it. NIST proposed a RBAC standard in [7], administrative models for RBAC were proposed in [6], [17], [18], [21], separation of duty constraints in RBAC were explored in [2], [8], [11], [24], workflow models based on RBAC were discussed in [1], [4], [9], and delegation models for RBAC were proposed in [3], [26]. All of these extensions and enhancements have been accomplished based on a formal RBAC model that was first proposed in [22]. Similarly, the groundwork developed in this paper aims to provide the formal and structural foundations to further develop RAdAC.

The paper is organized as follows. In section II, we review the core characteristics of RAdAC as specified in [12]. In section III we list the components that are needed to satisfy these characteristics and illustrate their interactions. Section IV gives a formal definition of our attribute-based RAdAC model. In section V, we describe the usage control (UCON) model and show that by adding suitable features and components it can capture the characteristics of RAdAC. We also discuss how aspects of UCON that are not directly used in this construction can beneficially improve the RAdAC vision. Section VI gives our conclusions and directions for future work.

II. Core Characteristics of RAdAC

In this section, we review the core characteristics of RAdAC which are stated as follows in [12].

1) *Operational Need:* Operational need is the reason for the user access. It can manifest itself in many ways in an access decision, e.g. at a coarse grain it could be represented by a user's membership in some community of interest or group. It can also be viewed as a supervisor or other approving authority attesting to a user's need to have access to specific information. In RAdAC, it is proposed that this characteristic convey some quantifiable measure in determining the access decision.
2) *Security Risk:* Security risk is described as a real time, probabilistic determination of risk. It is calculated from various factors such as trustworthiness of users, the protection capabilities and robustness of IT components, the operating threat level of the environment, and the access history. In RAdAC, it is proposed that the security risk evaluation be based on risk associated with each of these components, as well as a composite risk.
3) *Situational factors:* Situational factors are conditions under which the access decision is being made. National, enterprise or local situations may determine these conditions. For example, the national terrorist threat level may be considered a situational factor for access that could restrict or loosen access rules. In RAdAC, it is proposed that situational factors be considered along with operational need and security risk in making access control decisions.
4) *Adaptable Access Control Policy:* Access Control Policies specify the rules for access control for various classes of information objects under different conditions. In RAdAC, it is proposed that the access control policies be adaptable as access decision are made on acceptable levels of risk based on operational needs and situational factors.
5) *Heuristics:* Heuristics can be used to help fine-tune access control policies and improve future access decisions. For example knowledge of compromises that have resulted under various access decisions in the past, may help to more accurately determine risk and make better decisions. In RAdAC, it is proposed that heuristics be considered in the access decision process.

III. Components of RAdAC

In this section, we describe the components needed to model RAdAC and illustrate them in Figure 1. We also discuss how these components can capture the RAdAC characteristics. (In Figure 1, a solid arrow with a single arrowhead indicates a relation which has one element at that end whereas a double arrowhead indicates a relation which has many elements at that end, allowing us to visually distinguish one-to-one, one-to-many and many-to-many relations.)

The core components required to model RAdAC are identified as users, devices, purposes, objects, operations, connections, sessions and local and global situational factors. We require that the components users, devices, objects, purposes, operations, and connections have attributes which are properties (or distinguishable characteristics or capabilities) that are used for making the access decisions. We define situational factors as functional predicates that evaluate to true or false. The components relate to the above core characteristics of RAdAC as follows.

A. Component for Operational Need

We use the term Purpose (or Mission) to capture the characteristic of determining *"Operational Need"*. We think that this is more appropriate terminology to capture this concept. We define Purpose as the reason for the user's access request. An example where purpose attributes are used for making access control decisions is when a bank employee's request for access to a customer's account is granted only if there is a certified record of the customer's consent to access the record [12]. In the previous section, it was stated that there are many ways in which Purpose can manifest in an access decision, e.g., it could be a user's membership in a role or it could be that an authority is attesting to a user's need to access the object. In our abstract model, we consider Purpose as an attribute as it could cover both these and other use cases. The first one can be viewed as a user's explicit membership in a role and in the second case an authority's attestation for the user's need for access could grant the user a membership in the desired role. More generally fine-grained Purpose could be asserted by the user. For example, emergency treatment could be asserted as a Purpose in healthcare scenarios. Similarly, an impending power emergency could be asserted as a Purpose in industrial control systems.

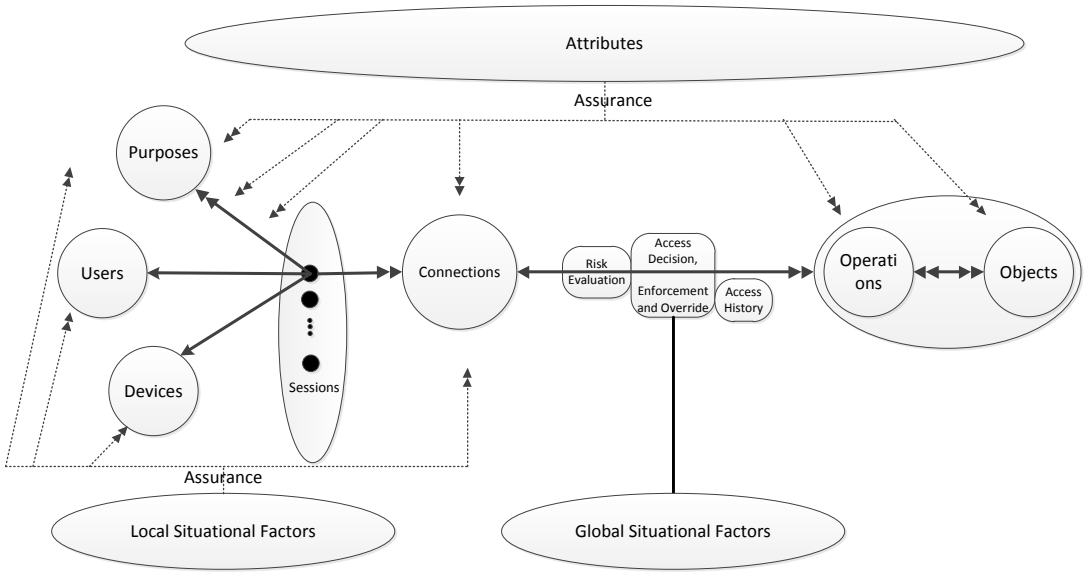

Fig. 1. RAdAC components

B. Components for Security Risk

We define the following components to capture *"Security Risk"*.

User: A User is defined as an entity that is requesting access. Although the concept of a user can be extended to include machines, networks, or intelligent autonomous agents, for simplicity we limit a user to a human being. We assume that the users and other components listed below are defined and represented by their attributes. Examples of user attributes include user's identity, user's security clearance, and so on.

Device: A device is defined as a computing device from which a user is requesting access. Examples where attributes of a device are used in making access control decisions are a user's access to on-line banking account is restricted to a pre-registered mobile device or an organization's policy allow VPN access to employees from a device registered in its domain.

Object: We define an Object as an entity that contains or receives information. Example of object attributes include the object security label (or classification).

Operation: We define Operation as an executable action, which upon invocation executes some function for the user on the object. Examples of operation attributes include the type of operation such as privileged, sensitive or normal and so on. The example from the Introduction section where a one-time display of a classified document on a (secure) computer terminal to a (cleared) individual is inherently less risky than providing that same individual with a paper copy (print operation) of the same document illustrates how the access control decision can be based on operation attributes.

Connection: We define Connection as any transport or communication channel on the user's device over which an operation is performed on an object. Examples of connection attributes are authentication strength, encryption key size and encryption algorithm.

Attribute Providers and Level of Assurance: One important aspect in determining security risk is the trust aspect of the attributes values. This trust can also be viewed as the assurance a relying party has in the attribute values asserted by the attribute provider. The assurance depends on two factors: the assurance in the provider of the attribute and the strength of the binding between the attributes and their values. An Attribute Provider may also assert different levels of assurance for a given attribute so the level of assurance asserted by an attribute provider also becomes an important factor in determining risk.

C. Component for Situational Factors

We define Situational Factors as environmental or system-oriented decision factors. We distinguish between two types of situational factors: Local Situational Factors and Global Situational Factors. A situational factor such as the national terrorist threat level is a Global Situational Factor. Situations related to a particular user or a group of users such as location, current local time for accessible time period (e.g., business hours), current location for accessible location checking (e.g., area code, connection origination point) are considered to be Local Situational Factors. In our model, we define Situational Factors as functional predicates that can be evaluated to be true or false.

D. Component for Heuristics

We define Access History as a function to capture the characteristic of *"Heuristics"*. The Access History Function

performs the following two tasks. First it updates the object access history repository with the attributes in the access request and the access control decision, and second it provides access to this data for making future access decisions.

E. Components for Adaptable Access Control Policies

Adaptable access control policies can be defined based on all the components defined in this section. Purpose is defined to capture user's need to access an object. Various other components such as device and connection, are defined to capture the security risk. Situational factors are defined to capture various conditions under which an access decision can be made and access history is defined to capture the access decisions that are made and also to provide input to the access decision process. Our model also allows for an override process where by an approved authority can override an access decision made by the system under specific conditions.

IV. RAdAC Abstract Model

We define the components described above formally and show how some sample adaptable access control policies can be defined in this model.

Definition 1: U, D, OBS, OPS, C, P and S are fixed sets of users, devices, objects, operations, connections, purposes and sessions respectively. Situational factors are functional predicates that evaluate to true or false.

session is defined as a four tuple $< u_i, d_j, p_k, c_l >$ where $u_i \in U$, $d_j \in D$, $p_k \in 2^P$ and $c_l \in 2^C$. That is, a session is associated with a single user and single device but can be associated with multiple purposes and multiple connections.

To formalize the attribute definitions we use the algebra introduced in [20]. We assume a vocabulary Σ of attribute names and domains. Each attribute is assumed to have a name *a*, a value in a domain denoted dom(*a*), the identity of the attribute provider *ap* and a level of assurance *loa* (if asserted by the attribute provider). UA, DA, CA, PA, OPA and OBA are user attributes, device attributes, connection attributes, purpose attributes, operation attributes and object attributes respectively.

Definition 2: Access Request (R) : An access request is of the form $R \equiv$
$\{(ap_1, a_1, v_1, loa_1), (ap_2, a_2, v_2, loa_2), ... (ap_k, a_k, v_k, loa_k) | (ap_i, a_i, v_i, loa_i) \in UA \bigcup DA \bigcup CA \bigcup PA \bigcup OPA \bigcup OBA$ for $1 \leq i \leq k\}$

Definition 3: Access Control Policy: A policy P is a function $P : R \rightarrow E$ from the domain of requests R onto the domain of decisions E, where $E = \{permit, deny\}$.

Definition 4: Access Decision Function: We also define an access decision function ADF() which applies all applicable access control policies to an access request and returns an access decision. The $ADF()$ function in turn implements a combining algorithm $combine\_f()$, which combines the decision results returned by the access control decision function of each policy and makes a final access decision. An access request is permitted if the final access decision is permit and denied otherwise.

$ADF(r) = combine\_f\{ADF(P1(r)), ADF(P2(r)), ... ADF(Pm(r))\} = \{permit, deny\}$

Definition 5: Access History is a function performing the following two tasks: first it updates the object access history repository with the attributes in the access request and the access control decision and second it provides access to this data for making future access decisions. The Risk Evaluation Function takes as input the *request* and Access History for the *request* and returns a risk value. The quantified risk value (rv) of a request is defined as:

$rv_1(r_1) =$
$RiskEvaluationFunction(r_1, ObjectAccessHistory(r_1))$

Example: Consider a modified MAC policy stated in [12]. A user's request to read a classified object is permitted only if the user's clearance equals or exceeds the classification of the object; the user has a documented need-to-know for accessing the object; the INFOCON level is at 3, 4, or 5; and the DEFCON level is at 3, 2, or 1. To this example, we add the condition that the risk level is acceptable (less than a specified value say x_1). The determination of the risk level considers factors such as the device of the user (authorized secure device or unsecured device), the facility from where the user is accessing (secure facility, or unknown facility), the location of the user and also the access history for such requests. Policy P1 can be expressed as:

$$P1(r) = \begin{cases} permit: \\ if \\ RiskEvaluationFunction(r) \leq x_1 \wedge \\ operation = read \wedge \\ INFOCONLevel \geq 3 \wedge \\ DEFCONLevel \leq 3 \wedge \\ (SubClearance(r) \geq \\ ObjClassification(r)) \\ \\ deny: \\ Otherwise \end{cases}$$

where r is a request of the form$\{(ap1, a1, v1, loa1), (ap2, a2, v2, loa2), (ap3, a3, v3, loa3)\}$ and x_1 is the maximum risk value acceptable for policy P1.

There can be numerous ways of expressing the RiskEvaluationFunction depending on the system risk tolerance requirements. For example, Fuzzy MLS [5] quantifies the risk of an access request based on a sigmoid function on the difference between a subject's clearance and an object's security label. Ni et al [14] provide a general methodology to implement customized risk-based access control by specifying fuzzy rules and show that access risks can be estimated from fuzzy inferences. Wang and Jin [25] propose a novel risk-quantification method for patient privacy protection in health information systems. In this paper, we are not trying to develop or propose ways for calculating risk of each access transaction. Our focus is to develop abstract models for RAdAC and as such keep these concrete details outside the direct scope of this paper.

V. Interpreting the RAdAC Model in UCON

So far, we have described the concepts of RAdAC and the components needed to express the RAdAC model. This raises an important question: can these concepts be supported by other attribute based access control models. We show that by appropriately defining the components needed we can interpret RAdAC characteristics with attribute based access control models such as UCON [19] . We selected UCON to simulate RAdAC due to its strong expressive power and policy specification flexibility. We start with a brief introduction of the concept of UCON and present a modified UCON model for RAdAC.

A. UCON Overview

The UCON model has six components: subjects and their attributes, objects and their attributes, rights, authorizations, obligations, and conditions. Authorizations, obligations, and conditions are components of usage control decisions in UCON. Authorizations are predicates based on the attributes; obligations are activities that have to be performed by subjects before or during an access; and conditions are system or environment restrictions that are imposed either before or during an access. The most important properties that distinguish UCON from traditional access control models are continuity of usage decisions and mutability of attributes. Continuity means that control decisions can be determined and enforced not only before an access but also during the period of the access and mutability of attributes means that subject and/or object attributes can be updated as the result of an access [19], [27].

B. Mapping RAdAC components in UCON

Before we can specify UCON policies for RAdAC, there are some key components and concepts that are missing in UCON. In this section we identify them and propose adding them to UCON to accommodate RAdAC. The key missing items are: (1) Subject definition, (2) Access History and (3) Risk Evaluation.

Subject definition: In UCON, a subject is defined as representing an individual human being with attributes [19]. We propose decomposing this generic definition into components needed for RAdAC which are users, device, purpose and connection. The session concept described in sections III and IV will combine these components for the purpose of capturing some of the characteristics of RAdAC.

Access History: One important characteristic of RAdAC is to consider the knowledge of past access control decisions in making each subsequent access decision. In UCON, the concept of such a feature is not defined although it can be partially captured via mutable attributes which change as the history unfolds. We propose adding this explicitly to UCON.

Risk Evaluation: Another important characteristic of RAdAC is the consideration of security risk for each access transaction. The UCON model compares attributes needed and their values and does not have this concept of quantifying risk for an access decision. We propose adding this as well feature to UCON.

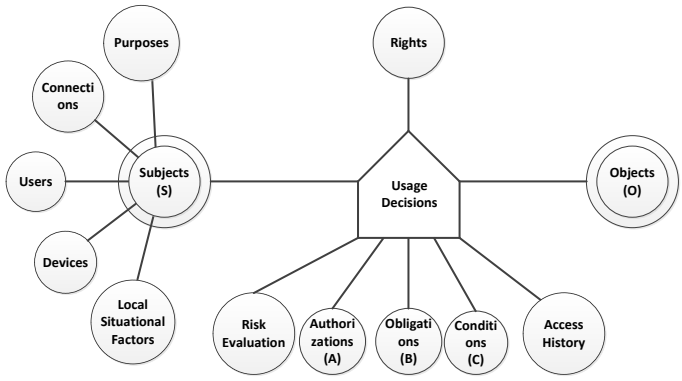

Fig. 2. UCON model with RAdAC Components

The components of RAdAC can be mapped to the extended UCON model with the decomposed subject definition as follows. The components of users, devices, purposes, objects and connections are same in both the models. The RAdAC component global situational factors can be captured in terms of UCON conditions. Additionally, RAdAC local situational factors can be expressed by subject attributes. The operations component in RAdAC is similar to the rights component in UCON. Authorizations and Usage Decision components in UCON is similar to the component of Access Enforcement and Decision depicted in Figure 1. The functions of Risk Evaluation and Access History in RAdAC become part of the Authorizations component in UCON so that the Authorizations component can evaluate risk for usage decisions. Figure 2 illustrates the extended UCON model.

C. Extending the UCON Principles to RAdAC

From the perspective of UCON, the concepts of attribute mutability and decision continuity are missing in RAdAC. We believe that these are important concepts that need to be considered for effective RAdAC. The concept of attribute mutability is important as attributes can change as a side effect of subjects access to objects. For example mutable attributes are credits/capabilities (e.g., $10 worth usage, five times per day, print twice), security clearance with relaxed (weak) or no tranquility, usage log (e.g., already read portion cannot be read again), and so on [19]. Decision Continuity is important as there is a need to continuously check (after a system grants access) in case certain attributes or situational factors changed and requirements for access are no longer satisfied. These concepts do not change the RAdAC components depicted in Figure 1, since we assume that the attribute providers assign the values. For mutable attributes that change as a side effect of access the access control system itself could be considered as an attribute provider. To maintain decision continuity the Risk Evaluation and Situation Factors components have to be

monitored repeatedly during the period of the access to ensure the access requirements are satisfied. In our opinion the UCON model with the decomposed subject definition and the added functions of access history and risk evaluation is most suitable for modeling and implementing the RAdAC concept.

VI. Conclusion and Future Work

In this paper we presented some preliminary concepts of RAdAC and showed that it can be expressed with a suitably extended form of UCON. It is our belief that RAdAC principles of real-time, adaptable, risk-based access control addresses real world scenarios where risk is an important factor in making access decisions. We did not include architecture and implementation issues in this paper and purely focused on the abstract models following the practice of keeping the model, architecture and implementation layers distinct.

The work presented in this paper can be extended along several directions. One such possible direction could be moving towards enforcement and implementation by defining the architecture, protocols and mechanisms for the proposed models. Another possible direction could be to look into the area of evaluating risk values for access transactions. The important question in this area would be to ask how can one be sure that a given RiskEvaluationFunction will perform as expected.

VII. Acknowledgments

The work of Ravi Sandhu is partially supported by grants from AFOSR-MURI, NSF and State of Texas.

References

[1] G.J. Ahn, Kang M., Park J., and R. Sandhu. Injecting RBAC to secure a Web-based workflow system. *ACM RBAC Workshop*, 2000.

[2] G.J. Ahn and R. Sandhu. Role-based authorization constraints specification. *ACM TISSEC*, 3(4), 2000.

[3] E. Barka and R. Sandhu. Framework for role-based delegation models. *Computer Security Applications Conference*, 2000.

[4] E. Bertino, E. Ferrari, and V. Atluri. The specification and enforcement of authorization constraints in workflow management systems. *ACM TISSEC*, 2(1), 1999.

[5] P.C. Cheng, P. Rohatgi, and et al. Fuzzy multi-level security: An experiment on quantified risk-adaptive access control. *IEEE Symposium on Security and Privacy*, 2007.

[6] J. Crampton and G. Loizou. Administrative scope: A foundation for role-based administrative models. *ACM TISSEC*, 6(2), 2003.

[7] 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), 2001.

[8] V. Gligor, S. Gavrila, and D. Ferraiolo. On the formal definition of separation-of-duty policies and their composition. *Proc. IEEE Symposium on Security and Privacy*, 1998.

[9] S. Kandala and R. Sandhu. Secure role-based workflow models. *IFIP TC11/WG11.3 Conf. on Database and App. Security*, 2001.

[10] R. Krishnan, R. Sandhu, and K. Ranganathan. PEI models towards scalable, usable and high-assurance information sharing. *ACM SACMAT*, 2007.

[11] D.R. Kuhn. Mutual exclusion of roles as a means of implementing separation of duty in role-based access control systems. *ACM RBAC Workshop*, 1997.

[12] R. McGraw. Risk-Adaptable Access Control (RAdAC). *NIST Privilege (Access) Management Workshop*, 2009.

[13] MITRE. Horizontal integration: Broader access models for realizing information dominance. *JSR-04-32, December 2004, http://www.fas.org/irp/agency/dod/jason/classpol.pdf*.

[14] Q. Ni, E. Bertino, and J. Lobo. Risk-based access control systems built on fuzzy inferences. *ASIACCS 2010*.

[15] Department of Defense. Directive 8000.01. *February 10, 2009, http://www.dtic.mil/whs/directives/corres/pdf/800001p.pdf*.

[16] Department of Defense. Global information grid architectural vision. *Vision for a Net-Centric, Service-Oriented DoD Enterprise, June 2007, http://cio-nii.defense.gov/docs/GIGArchVision.pdf*.

[17] S. Oh and R. Sandhu. A model for role administration using organization structure. *ACM SACMAT*, 2002.

[18] S. Oh, R. Sandhu, and X. Zhang. An effective role administration model using organization structure. *ACM TISSEC*, 9(2), 2006.

[19] J. Park and R. Sandhu. The UCON$_{ABC}$ usage control model. *ACM TISSEC*, 5(6), 2007.

[20] P. Rao, D. Lin, E. Bertino, N. Li, and J. Lobo. An algebra for fine-grained integration of xacml policies. *ACM SACMAT*, 2009.

[21] R. Sandhu, V. Bhamidipati, and Q. Munawer. The ARBAC97 model for role-based administration of roles. *ACM TISSEC*, 2(1), 1999.

[22] R. Sandhu, E J. Coyne, H L. Feinstein, and C E. Youman. Role-based access control models. *IEEE Computer*, 29(2), Feb 1996.

[23] R. Sandhu, Ranganathan K., and Zhang X. Secure information sharing enabled by trusted computing and pei models. 2006.

[24] R. Simon and M.E. Zurko. Separation of duty in role-based environments. *Comp. Sec. Foundations Workshop (CSFW)*, 1997.

[25] Q. Wang and H. Jin. Quantified risk-adaptive access control for patient privacy protection in health information systems. *ASIACCS*, 2011.

[26] L. Zhang, G.J. Ahn, and B.T. Chu. A rule-based framework for role-based delegation and revocation. *ACM TISSEC*, 6(3), 2003.

[27] X. Zhang, M. Nakae, M. Covington, and R. Sandhu. Toward a usage-based security framework for collaborative computing systems. *ACM TISSEC*, 11(1), 2008.