Role Explosion: Acknowledging the Problem

A. A. Elliott and G. S. Knight
Math and Computer Science, Royal Military College, Kingston, Ontario, Canada

Abstract - *In large enterprises subject to constant employee turnover and challenging security policies, the administration of Role-based Access Control (RBAC) is a daunting task that is often highly centralized in a small team of security administrators. The aim of this work is to determine why existing models for Administrative Role-based Access Control (ARBAC) have failed to achieve success and thus motivate the requirement for a new model named One+ RBAC Administration (ARBAC1+). In order to meet this objective, the term role explosion is symptomized and supported with case studies that identify misconceptions found in previous ARBAC models. Then ARBAC1+ is proposed within the context of the Government of Canada, however, its use is not limited to this organization.*

Keywords: Information Management, Identity Management, Authentication, Role-based Access Control, Authorization.

1 Introduction

In recent years, the use of Role-based Access Control (RBAC) has evolved within large organizations such as the Government of Canada (GoC). Although, RBAC provides a solid foundation for managing information security, administrators may be burdened with the maintenance of hundreds or thousands of roles across several applications. Managing these roles, users and their interrelationships is a formidable task that is often highly centralized in small teams of security administrators [8]. This is a daunting task in large organizations where security practitioners perform access control as a secondary duty and are provided various levels of training and formalized knowledge [1].

Administrative Role-based Access Control (ARBAC) models have been proposed as one means of formalizing the management of these roles, users and their interrelationships [8][9][5][4]. The idea of using RBAC to manage RBAC is promising but no studies have been found indicating the adoption of ARBAC in any capacity. Previous models for ARBAC are challenged due to intuitions inherent in their design and misconceptions found in the literature. As a result, no standardized model for RBAC administration exists when there are "… more and more new types of applications that require controlled sharing of resources or discrimination of information …" [1]. The aim of this work is to determine why existing models for Administrative Role-based Access Control (ARBAC) have failed to achieve success and thus motivate the requirement for a new model named One+ RBAC Administration (ARBAC1+). In order to meet this objective, the term role explosion is symptomized and supported with case studies that identify misconceptions found in previous ARBAC models before ARBAC1+ is introduced.

The rest of this paper is organized as follows; Section 1 provides background information, motivates this work and specifies the research aim. Section 2 introduces the notion of role explosion and provides supporting case studies. Section 3 synthesizes a new approach for RBAC administration and section 4 concludes this work.

1.1 Information Management System

Information Management (IM) systems control access to corporate information in dynamic, heterogenous infrastructures with challenging combinations of employee turnover and security policy. Protecting information and client data is an on-going concern for large public and private organizations in Canada, ranking highest in each of the last two editions of the annual Symantec Pulse of IT Security in Canada surveys [11][12]. In the 2007 survey, Canadian organizations, both public and private, acknowledged that defending against unauthorized access by an employee represents a whole different set of challenges [12]. In the Government of Canada (GoC), one of these challenges is the least-privilege principle. Under section 16.4.3 of the Operational Security Standard: Management of Information Technology Security, security administrators must keep user access to the minimum required for an individual to perform their duties. Furthermore, security administrators must ensure that access control implementations are regularly updated to accurately reflect the current responsibilities of any individual in the organization [14].

1.2 Role-based Access Control

The RBAC model was formally introduced by David F. Ferraiolo and Richard Kuhn at the 15th National Security Conference in October 1992 [3]. It has produced a standard, ANSI INCITS 359-2004, intended for software engineers designing products with role-based access control features [10].

A role is a semantic construct associated with permissions. Roles are created for the various job functions in an organization and users are assigned roles based on their

responsibilities. Users can be easily reassigned from one role to another. Roles can be granted new permissions as new applications and systems are incorporated and permissions can be revoked from roles as needed. Role-role relationships can be established to implement broad policy objectives. This simplifies the effort required to manage security by reducing the number of administrative actions as illustrated in Figure 1. In case 1 there are five users and five tables, the total number of administrative actions is twenty-five when granting object permissions directly to users and ten when using the RBAC model – a savings of fifteen administrative actions. In case 2, the savings is eight thousand eight hundred administrative actions.

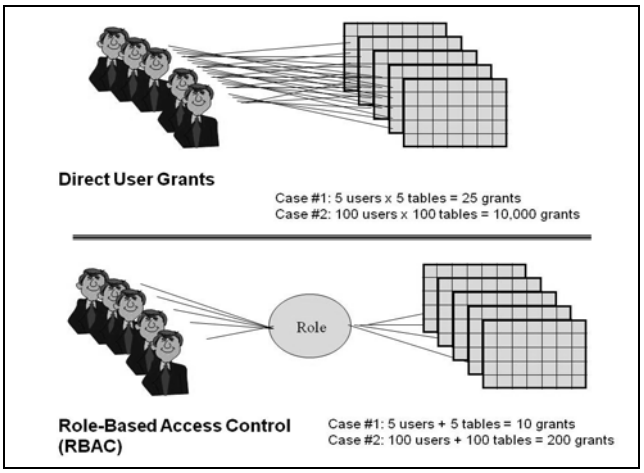

Figure 1. Classic Example of RBAC Administrative Savings.

1.3 Employee Turnover and Security Policy

Although RBAC provides a solid foundation for managing security in enterprise environments and Figure 1 is an excellent "textbook" example, no standard exists for the administration of RBAC itself. As stated earlier, the idea of using RBAC to manage RBAC with Administrative Role-based Access Control (ARBAC) is promising but previous models have failed to achieve success. This is due to usability issues in enterprise environments where employee turnover and challenging security policies are dynamic forces that introduce flux and perturb the administration of RBAC beyond the expectations of existing models.

In an enterprise environment, a new employee may be granted access to several IM systems using an administratively heavy person-based process [2]. This compounds the problem of RBAC administration because employees are constantly entering, exiting and moving within enterprise organizations. An RBAC implementation for one small representative IM system might look like the one pictured in Figure 2 where the employee turnover process is occurring. This IM system requires twelve roles for five employees and it restricts access to information on a "need to know" basis. Several administrative actions are required to revoke access from the departing employee (1) and grant access to the new employee (2). Furthermore, this may be one system amongst dozens found in an enterprise organization and employees often require access to several systems [2].

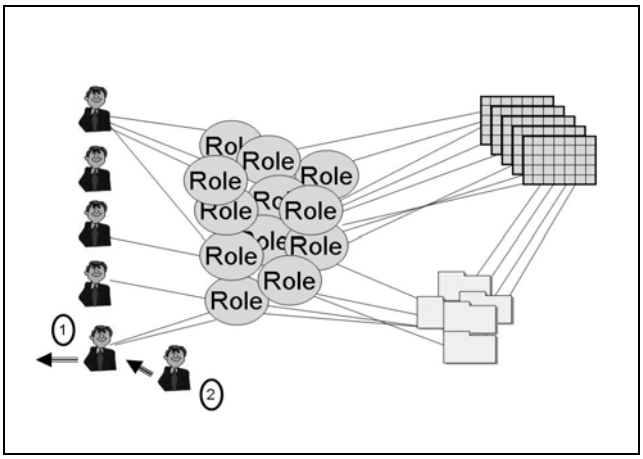

Figure 2. Old-employee-out (1) and New-employee-in (2).

As a result, it is very challenging to maintain a practical RBAC implementation that is tightly coupled with an organization's security policy. In the Government of Canada (GoC), access control policy is specified in the Government of Canada Security Policy (GoC-SP) requirements section entitled Access Limitations. Under section 10.8, departments must limit access to classified and protected information and other assets to those individuals who have a need to know the information and who have the appropriate security screening level [13].

In large enterprise organizations, security policies are regularly amended in response to cultural, technological and social change. Previous models for ARBAC have failed to support this flux in enterprise organizations and they have failed to quantify the savings or return on investment that organizations achieve by implementing these models. Without an accepted standard, it is difficult for organizations to determine their adherence to this standard and in turn the relative quality of their RBAC implementation.

2 Role Explosion

In their introduction to administrative role-based access control (ARBAC) [8], the authors of ARBAC97 offer the following commentary with respect to large enterprise systems, "The number of roles can be in the hundreds or thousands and users in the tens or hundreds of thousands." This belief, intuition or sense is repeated in ARBAC99 and ARBAC02 without reference to the origin of these numbers [9][5].

If the architects of these models were under the assumption that the number of users is disproportionately superior to the number of roles in an IM system then this influenced the development of their models. This work provides contradictory evidence, and suggests that the number of roles approaches or surpasses the number of users in large IM systems with specialized employees.

Although the identity management problem is well understood [6], research performed over the last several years suggests that the separate problem concerning the proliferation of roles is not generally appreciated within the academic and practitioner communities. Consider the following motivating example from a large enterprise organization:

- 1 Employee
- 10 Applications or Services / Employee
- 2 Roles / Application

If one employee requires ten applications or services with two roles per application then the number of roles being managed for one employee is twenty. A deeper understanding of the practical issues facing today's enterprise organizations is required in support of the next generation of models for RBAC administration. To build a clear understanding of role proliferation within the academic and practitioner communities, the term role explosion is introduced and the following symptoms are indicative.

Symptom 1 An enterprise organization requires employees to access several IM systems and most (or all) of the systems autonomously manage their own set of role (or group) information.

This symptom contributes to the role explosion problem because it is administratively costly to introduce and maintain redundant role information across several IM systems. The problem is similar to the identity management issue because each system does not need to hold its own role information. A centrally managed role repository should be used instead. This first symptom of role explosion is not uncommon in enterprise organizations and should not surprise academics or practitioners of RBAC but perhaps the second symptom will.

Symptom 2 An enterprise organization has one or more IM systems where the total number of users approaches or surpasses the total number of roles.

This symptom contributes to the role explosion problem because it is administratively costly to maintain an IM system where the number of roles is directly proportional to the number of users. In fact, no previous model for ARBAC has anticipated this scenario [8][9][5][4]. The following case study for symptom 2 is an investigation and analysis of the role information found in one representative IM system where hundreds of roles have been defined with respect to the Operational Security Standard: Management of Information Technology Security (MITS), a derivative of the Government of Canada Security Policy (GoC-SP) [13][14].

2.1 A Representative IM System Case Study

Role-based Access Control is the foundation for "need to know" implementations in IM systems. As a result, organizations with extensive work breakdown structures are naturally afflicted with role explosion over time as more and more applications are integrated into an organizational infrastructure and more and more specialization (or customization) is supported in an IM system. Table 1 lists role and user information for the IM system. The role granularity metric is a simple ratio, comparing the total number of roles, ∑R, to the total number of users, ∑U, for the representative IM system by calendar year.

Table 1. Role and User Information for a Representative IM System.

Calendar Year	Object Type	Net Objects Added During Year	Cumulative Object Total at Year End	Role Granularity Metric ∑R / ∑U
2005	Role	348	348	
2005	User	271	271	
2005				1.3
2006	Role	+93	441	
2006	User	+24	295	
2006				1.5
2007	Role	+65	506	
2007	User	+40	335	
2007				1.5
2008	Role	+57	563	
2008	User	+44	379	
2008				1.5

In 2005, a database upgrade occurred and yearly information has not been obtained before this time period. The data shows role objects outnumbering user objects from the onset. In 2006, role information was added at an approximate 4:1 ratio with respect to user information and the role granularity metric increases from 1.3 to 1.5. In 2007 and 2008, the role granularity metric remains static with a net of one and one half roles being added for every user added to the IM system. Based on this case study, one can see that role explosion exists in production IM systems. Therefore, it is a fallacy to assume that the number of users is orders of magnitude greater then the number of roles in an IM system.

2.2 A Representative Employee Case Study

This case study is an investigation and analysis of the on-boarding administrative actions for one GoC employee. The participating employee is Bob, the Administration Officer for the Information Services department. The roles and responsibilities of Bob's staffing position dictate that he is granted access to the systems listed in Table 2. Table 3 captures the role information associated with each of Bob's accounts as listed in Table 2.

Table 2. IM Systems Required by a Representative Employee.

#	Acronym	IM System
1	NET	Network Access
2	MAIL	Mail Account
3	AD	Directory Account
4	DWAN	Defense Wide Area Network Account
5	FMAS	Financial and Managerial Accounting System
6	CLX	Claims-X Web
7	SHP	Sharepoint
8	CISA	Enterprise Application
9	PORTAL	Enterprise Portal

Table 3. User-role Assignments for a Representative Employee

#	System	Role(s)
1	NET	STAFF, CIS
2	MAIL	STAFF, CIS
3	AD	STAFF, CIS
4	DWAN	KG-CIS, U-DomainUsers
5	FMAS	UU19, UU38
6	CLX	KG-CLAIMS-X
7	SHP	CIO, CIS
8	CISA	P123456, SWEMAN, SPM
9	PORTAL	GRP_STAFF

In measuring current procedures and practices, the foundation or baseline is formed from which one can lay claim to an improved administration model for identity and access management (IAM). In Table 4, the total number of administrative actions associated with each on-boarding workflow is tabulated for various stages.

Table 4. Administrative Actions for the On-boarding of a Representative Employee.

#	System	RQ	AP	User	Role	NT	Admin. Actions
1	NET	1	1	1	0	1	4
2	MAIL	0	0	1	0	0	1
3	AD	0	0	1	0	0	1
4	DWAN	1	2	1	1	1	6
5	FMAS	1	3	1	2	1	8
6	CLX	1	1	1	1	1	5
7	SHP	1	1	1	1	1	5
8	CISA	0	1	1	2	0	4
9	PORTAL	0	1	0	0	0	1
		5	10	8	7	5	35

At the user account request phase (RQ) an administrative action such as providing the applicable form is counted. During the authorization phase (AU) the number of signatures and/or electronic approvals are summed. At the user account creation phase (User) each IM system or dependent service includes a manual account creation process with the exception of the PORTAL where accounts are automatically created based on events in the IM System. Next, all user-role grants (Role) are summed if and only if the administrator must manually assign a role to the user account. Finally, the notification column (NT) identifies systems where the user is manually notified when their account is available and provisioned with their username and password in this process.

In summary, this case study shows that for one employee of the GoC, a total of nine applications (or dependent services) are required to fulfill the responsibilities of the staffing position. A total of seventeen roles are associated with the employee and thirty-five administrative actions are required during the on-boarding process.

3 One+ RBAC Administration

In the previous section, the concept of role explosion is used to motivate the requirement for an improved ARBAC model. One+ RBAC Administration (ARBAC1+) acknowledges role explosion, adding a layer of "role abstraction" that defines a 1-1 relationship between a centralized role repository and the staffing positions maintained in an HR system. In heterogenous infrastructures with challenging combinations of employee turnover and security policy, this simplifies the automation of user-role grants as illustrated in Figure 3.

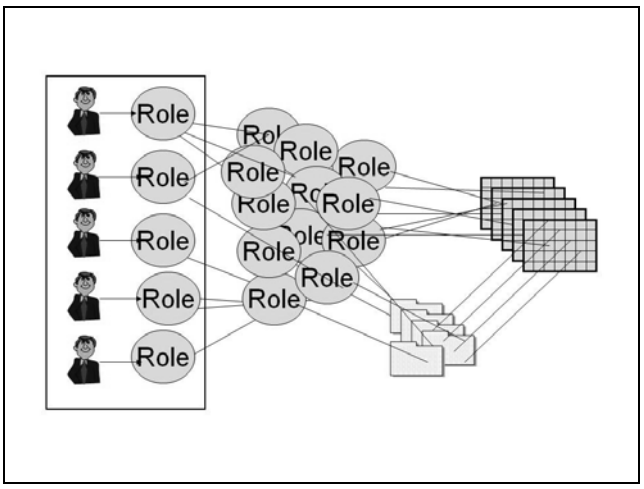

Figure 3. One+ RBAC Administration.

ARBAC1+ is based on real-world practice and the work of authors who have contributed sound theory to the concept of role administration [7][8][9][5][4]. Unlike the concept of groups, which specify a collection of users, roles identify a collection of users and a related collection of permissions. For this reason, RBAC administration is multi-faceted. Assigning users to roles, assigning roles to roles and assigning permissions to roles are distinct sets of actions required to bring users and permissions together [8]. In the following subsections, ARBAC1+ definitions and details are provided for user-role assignment (URA1+), role-role assignment (RRA1+) and permission-role assignment (PRA1+).

3.1 User-Role Assignment (URA1+)

While acknowledging the elegance of ARBAC97, the authors of ARBAC02 illustrate redundancies and unnecessary couplings for both user-role assignment (URA97) and permission-role assignment (PRA97) [5]. To address these practicality issues, ARBAC02 introduces the notion of organizational units as the logical containers or "pools" for new users and permissions. This eliminates redundancy in the organization, making ARBAC02 more resilient to dynamic forces of change like employee turnover and security policy.

In ARBAC02 notation, the '@' symbol is a "pointer" to an external HR system. This modification addresses the issue of multi-step user-role assignment in ARBAC97 where, for example, new employee John must be sequentially assigned to the role Employee followed by the roles Engineering Department, Engineer #1 and finally Quality Engineer #1. Instead, ARBAC02 uses the assignment of John to the organizational unit Engineering Department in the Human Resources system to place John @ the user pool where he may be assigned to Engineer #1 and then Quality Engineer #1. This eliminates the first two user-role assignments of ARBAC97, thus providing a fifty percent administrative savings.

One weakness of ARBAC02 is its failure to explicitly incorporate the activities of the Human Resources (HR) group. In ARBAC1+, the activities of the HR group are used to maintain the "role abstraction" layer. This layer abstracts notional groups and abilities away from the user with persistent, secondary role-role relationships. This facilitates employee turnover because enterprise organizations typically choose to remove all user accounts when an individual exits the organization, thus implicitly removing all user-role assignments [2]. If these user-role relationships are not formally documented they may be irrevocably lost.

For a set of Human Resource roles, HR, and a given set of position roles, P, let PR denote the set of all possible nodes, trees and exclusions that can be formed using the roles in P. Position hierarchies are maintained in the HR system.

Definition 1 The URA1+ model controls user-role assignment with the relation $can\_assignu \subseteq HR \times PR$

Definition 2 The URA1+ model controls user-role revocation with the relation $can\_revokeu \subseteq HR \times PR$

The meaning of *can_assignu(x, y)* is that a member of the Human Resources role, *x*, can assign a user to be a member of the position nodes permitted by *y*. The meaning of *can_revokeu(x, y)* is that a member of the Human Resources role, *x*, can revoke a user from the position nodes permitted by *y*.

3.2 Role-Role Assignment (RRA1+)

RRA1+ modifies the mutually disjoint roles introduced in RRA97 by redefining Abilities and Groups and replacing UP-roles with Positions [8]. Abilities (A) are roles that can have permissions, other abilities, groups and positions as members. Abilities aggregate the permissions required to perform some task into a role. Abilities may be organized into hierarchies and assigning abilities to roles is the same as assigning permissions to roles. Groups (G) are roles that can have abilities, other groups and positions as members. Positions (P) can have users, groups and abilities as members.

For a set of position roles, PR, and a given set of abilities, A, let AR denote the set of all possible nodes, trees and exclusions that can be formed using the roles in A. Position roles may only assign roles to or revoke roles from subordinate position roles as maintained in the HR system.

Definition 3 The RRA1+ model controls ability-role assignment with the relation $can\_assigna \subseteq PR \times AR$

Definition 4 The RRA1+ model controls ability-role revocation with the relation $can\_revokea \subseteq PR \times AR$

The meaning of *can_assigna(x, y)* is that a member of the position role, *x*, can assign a subordinate position role abilities permitted by *y*. The meaning of *can_revokea(x, y)* is that a member of the position role, *x*, can revoke from a subordinate position role abilities permitted by *y*.

For a set of position roles, PR, and a given set of groups, G, let GR denote the set of all possible nodes, trees and exclusions that can be formed using the roles in G. Position roles may only assign roles to or revoke roles from subordinate position roles as maintained in the HR system.

Definition 5 The RRA1+ model controls group-role assignment with the relation $can\_assigng \subseteq PR \times GR$

Definition 6 The RRA1+ model controls group-role revocation with the relation $can\_revokeg \subseteq PR \times GR$

The meaning of *can_assigng(x, y)* is that a member of the position role, *x*, can assign a subordinate position role groups permitted by *y*. The meaning of *can_revokeg(x, y)* is that a member of the position role, *x*, can revoke from a subordinate position role groups permitted by *y*.

3.3 Permission-Role Assignment (PRA1+)

In ARBAC1+ the activities of the Information Management (IM) group are explicitly incorporated into permission-role assignment (PRA1+) in support of role-role assignment (RRA1+). ARBAC1+ considers pemission-role assignment the domain of the IM group, whose technical knowledge enables them to abstract and group low-level assignments such as "grant select on table [x] to role [y]" to a high-level ability such as "grant [course registration] to [notional group/positional role]". This eliminates the delegation of permission-role assignment (outside the IM group) and facilitates the delegation of role-role assignment in a scalable architecture where the IM group assigns abilities and groups to staffing position roles as authorized. Abilities may then be subdelegated as required (and permitted) by the holder(s) of the applicable positional role(s) and Groups may be assigned (as permitted) by the holder(s) to the applicable positional role(s).

3.4 Towards a Unifying Standard

There remains no unifying standard for the administration of RBAC despite the introduction of ARBAC more then ten years ago [8]. Current ARBAC models have failed to maintain a balance between access control and usability because of intuitions and assumptions inherent in their architecture. Exacerbating these design flaws is the concept of role explosion which results from constant employee turnover and security policy revision.

ARBAC97 offered a promising solution whereby administrators use RBAC to manage RBAC [8]. However, the duplication of the organizational hierarchy in an equivalent role hierarchy and the prerequisite and redundant role grants inherent in this model are too administratively heavy. Its immediate predecessor ARBAC99 adds additional complexity with very little gain [9]. With ARBAC02, a significant step forward is achieved, in that, the authors recognize the mirroring of the organizational hierarchy and introduce the concept of user and permission pools [5]. This is an interesting point of integration for enterprise organizations with Human Resource (HR) systems. Unfortunately, ARBAC02 fails to explain how this integration is to be accomplished, considering this integration challenge out of scope. A-ERBAC extends ARBAC02 with the introduction of scopes, providing additional generalizations for RBAC administration [4].

This section introduced a novel approach for RBAC administration. ARBAC1+ extends ARBAC02 with the concept of position roles and uses ideas presented in A-ERBAC as a practical means to constrain the assignment and delegation of roles. ARBAC1+ does not inherit the intuitions and assumptions of previous models. It completely "walks away" from the concept of a disjoint administrative role hierarchy as presented in ARBAC97 [8]. In ARBAC02, the notion of a user pool replaces the prerequisite role(s) of the can_assign relation in ARBAC97. Although the authors of ARBAC02 conclude that this is a practical step forward, they fail to highlight this shift in control away from the administrative hierarchy. In fact, the ARBAC02 model effectively bypasses one or more levels of the administrative role hierarchy by design. ARBAC1+ bypasses all levels and completely eliminates the requirement for an administrative hierarchy.

Furthermore, ARBAC1+ does not share the belief or intuition that less roles implies a better RBAC implementation. To the contrary, ARBAC1+ contends that more roles increase the opportunities for automation and delegation which implies less manual administration. The foundation of ARBAC1+ is a 1+ "role abstraction" layer that facilitates user-role automation and role-role delegation for enterprise environments with specialized employees.

4 Conclusions

The aim of this work is to determine why existing models for Administrative Role-based Access Control (ARBAC) have failed to achieve success and thus motivate the requirement for a new model named One+ RBAC Administration (ARBAC1+). In order to meet this objective, the term role explosion is symptomized and supported with case studies that identify misconceptions found in previous ARBAC models. Then ARBAC1+ is proposed within the context of the Government of Canada, however, its use is not limited to this organization.

A unifying standard is required for the administration of RBAC as previous models have failed to achieve general acceptance during the last decade. Unlike its predecessors, ARBAC1+ anticipates *role explosion* in enterprise organizations where employee turnover and security policy perturb the administration of RBAC beyond the expectations of previous models. ARBAC1+ excludes the administrative role hierarchy of current ARBAC models and contends that more roles increase the opportunities for automation and delegation which implies less manual administration.

5 References

[1] Beznosov, K., Inglesant, P., Lobo, J., Reeder, R., and Zurko, M. 2009. Usability meets access control: challenges and research opportunities. In Proceedings of the 14th ACM Symposium on Access Control Models and Technologies (Stresa, Italy, June 03 - 05, 2009). SACMAT '09. ACM, New York, NY, 73-74.

[2] Elliott A. and Knight S. 2009. One Employee and Several Applications: An Information Management Case Study. In Proceeding of the 2009 International Conference on Software Engineering Research & Practice (Las Vegas, Nevada, USA, July 13 - 16, 2009). SERP 2009. CSREA Press, pp. 179-185.

[3] Ferraiolo, D. and Kuhn, R. 1992. "Role-based Access Control", Proceedings of 15th NIST-NCSC National Computer Security Conference, Baltimore, MD, 13-16 October 1992, pp. 554-563.

[4] Kern, A. 2002. Advanced Features for Enterprise-Wide Role-Based Access Control. In Proceedings of the 18th Annual Computer Security Applications Conference (December 09 - 13, 2002). ACSAC. IEEE Computer Society, Washington, DC, 333.

[5] Oh, S. and Sandhu, R. 2002. A model for role administration using organization structure. In Proceedings of the Seventh ACM Symposium on Access Control Models and Technologies (Monterey, California, USA, June 03 - 04, 2002). SACMAT '02. ACM, New York, NY, 155-162.

[6] Recordon, D. and Reed, D. 2006. OpenID 2.0: a platform for user-centric identity management. In Proceedings of the Second ACM Workshop on Digital Identity Management (Alexandria, Virginia, USA, November 03 - 03, 2006). DIM '06. ACM, New York, NY, 11-16.

[7] Sandhu, R., Coyne, E.J., Feinstein, H.L., and Youman, C.E., "Role-based access control models", IEEE Computer, February 1996, Volumne 29, Number 2, pp 38-47.

[8] Sandhu, R., Bhamidipati, V., and Munawer, Q. 1999. The ARBAC97 model for role-based administration of roles. ACM Trans. Inf. Syst. Secur. 2, 1 (Feb. 1999), 105-135.

[9] Sandhu, R. and Munawer, Q. 1999. The ARBAC99 Model for Administration of Roles. In Proceedings of the 15th Annual Computer Security Applications Conference (December 06 - 10, 1999). ACSAC. IEEE Computer Society, Washington, DC, 229.

[10] Sandhu, R., Ferraiolo, D. and Kuhn R. 2004. "American National Standard for Information Technology – Role Based Access Control", ANSI INCITS 359-2004, February 3, 2004.

[11] SYMANTEC 2008. Pulse of IT Security in Canada – Volume V. Branham Group Inc. February 20, 2009.

[12] SYMANTEC 2007. Pulse of IT Security in Canada – Volume VI. Branham Group Inc. May 24, 2007.

[13] Treasury Board of Canada Secretariat. (2002, February 1). Government Security Policy.

[14] Treasury Board of Canada Secretariat. (2004, May 31). Operational Security Standard: Management of Information Technology Security (MITS).