Attribute and Role-Based Access Control Models

Michael Haythorn

East Carolina University

4/13/2014

Abstract

Access Control is a fundamental responsibility of information security professionals. The basic need to provide users with the access required to complete their job creates the equal need to restrict this access in some way. Since it is not always appropriate for every member of an organization to have access to every object, a division of this access based upon the duties and responsibilities of individuals is created. This is where Role-Based Access Control can be implemented in an organization to create this division between jobs in order to prevent users from gaining inappropriate access, prevent malicious actions from internal users as well as prevent fraudulent occurrences.

Additionally, the relatively new concept of Attribute-Based Access Control can also be implemented in combination with Role-Based Access Control to further increase the amount of security applied to access and role permissions. It is the responsibility of information security professionals, including risk management teams, access administration and senior management to implement the solution that is the most effective for their particular organization. There are a variety of options when addressing with access control, attribute and role access control is a common choice for organizations looking to increase security around access.

The following article presents a history and details on Role-Based and Attribute-Based Access Control methods including the various models that are optional for increasing the amount of security. Implementation and management of this type of security solution requires cooperation of all members of an organizations structure and can be applied to any existing information security practice including organizations that are small or large.

Table of Contents

1 History of Role-Based Access Control

Until the 1990's, the best known U.S computer security standard was the Trusted Computer System Evaluation Criteria or TCSEC introduced by the Department of Defense. TCSEC specified two types of access control, Mandatory Access Control (MAC) and Discretionary Access Control (DAC). MAC was a well suited access control model for military and government organizations where there existed an obvious division between levels of confidentiality. Each member was classified at a certain level and with their level came a rigid set of permissions. The system was developed to be so rigid so no users could access levels they were not authorized to. On the other hand, DAC catered toward small organizations where individual owners of resources were able to assign access based on knowledge and other criteria.

When it came to addressing access control inside of larger organizations, MAC was missing the granular access restriction abilities as most organizations were not set up with, nor had the need for firm divisions between confidentiality levels. In an organizational setting, MAC both over restricts access to lower level users and over grants access to higher level users. DAC also did not provide an ideal model for organizations due to the difficulty of scalability as well as the lack of ability to monitor access privileges. The DAC method works on small scales typically in smaller organizations where individual data owners know who needs access to their data and can manage these requests. No central management can create a confusing web of access that can be next to impossible to audit.

Out of this, the concept of Role-Based Access Control or RBAC began. RBAC provided large organizations with the ability to create roles based on an individual's duties, responsibilities or qualifications that enabled them to be grouped together in some way. In 1992, the National Institute of Standards and Technology published an article on Role-Based Access Control that began to define this concept and its usefulness in organizations. One of the major drivers for RBAC is to allow organizations the ability to easily administer, review and audit access. Organizations that utilize this model can have a means to manage authorization privileges.

2 Role-Based Access Control Models

The basic concept of Role-Based Access Control is to establish permissions based roles dependent on functional responsibilities of users within an organization. Users can be assigned to

different roles based on their duties, responsibilities or projects. Organizationally, roles are structured based subsets of responsibilities that can be grouped into common access requirements. The intent of role creation and provisioning reduces the amount of exceptional access by grouping users through commonalities of access. This process can greatly reduce the amount of overhead required to manage access, as well as reduce the amount of requests for access that would normally be required for each user.

There are three main requirements in order to implement Role-Based Access Control in an organization. First there must be users who can be a part of a role; a user can be an employee of the organization, a contractor who works outside but with access requirements to company assets, a generic account responsible for running jobs, a printer necessary for proper documentation, etc. Once the list of users has been defined the next requirement is what the users can be allowed to access. These permissions need to be categorized based on criticality and amount of access to be granted. Permissions can be access to applications, functions within applications, access to systems or resources, access to audit logs, etc. Once the permissions have been identified the roles can then start to be created. The users that have been identified initially can then be grouped together by similarities of permissions. Out of this combination the organizations, risk management teams can create one or many roles to meet the needs that have been identified. A role is essentially the combination of users and their permissions grouped into one. Figure 2 below provides this basic concept showing that a user is assigned to a role and then inherits the permissions that have been granted through the role itself.

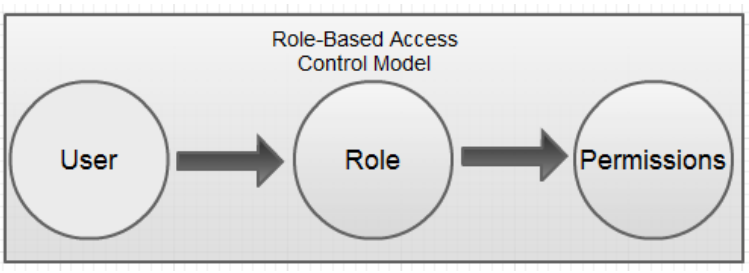

Figure 2 – Role-Based Access Control

The creation of roles simplifies the process to request or obtain access as well as the process to audit existing access. For example, if ten users are a part of the same role, it is much easier to review the access in the role rather than reviewing individual access of each of these users. At

the same time if this role grants access to ten applications and functions, it is much easier for each user to request one role, versus requesting access individually to each application and function.

There are multiple ways to implement Role-Based Access Control into an organizational structure. The correct option is based on a combination of organizational size, possible permissions and number of different roles and responsibilities. Due to this several models inside of RBAC have been established to provide variations of implementation depending on multiple factors, including increasing security. As mentioned, it is not always appropriate to assign all users access to all permissions; therefore roles must separate users based on their responsibilities. Maintaining the least privilege possible while providing users with the access they need is an essential balance for organizations who implement role based access control.

2.1 - Flat RBAC

The Flat Role-Based Access control is the most basic principle associated with the RBAC. In this model there are permissions assigned to a role and then the roles are assigned to members. Users can be assigned to multiple roles and each role can be assigned to multiple users. There is no restriction of access between role permissions in flat RBAC. Due to this, each user is authorized any permission by each role they are assigned. Figure 2.1 below provides a sample of this flat level of access, a user is assigned roles A, B and C and from this membership the user is granted all access associated with the permissions assigned to each role.

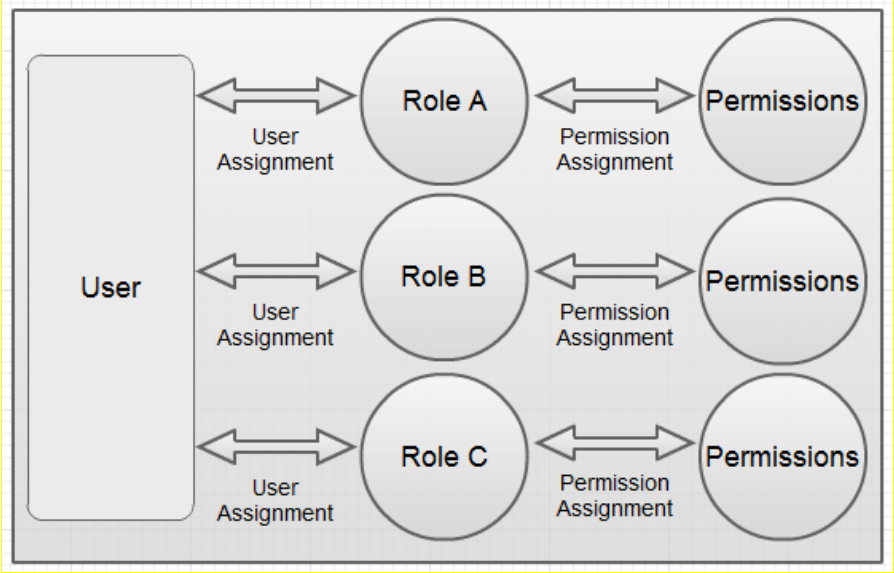

Figure 2.1 – Flat Role-Based Access Control Model

There are many drawbacks to flat RBAC especially around the need to restrict access in certain cases. Granting access is a simple process, but the real difficulty comes when access needs to be restricted. Additionally the flat model is difficult to scale because it requires a single role be created for each user as there is no inheritance of access beyond the first role. This creates the requirement to have multiple top level roles in order to satisfy the needs of the organization for various jobs.

2.2 - Hierarchical RBAC

The natural next level above flat RBAC is to add inheritance of permissions based on role. This model is known as Hierarchical Role-Based Access Control where the establishment of a hierarchical system is made. In hierarchical RBAC there are senior roles and junior roles where a senior role inherits the access associated with a junior role. A senior role is restricted to privileged members who require additional or administrative access to the lower roles. The intention is that members can be assigned to any one role, and with that role be granted sub roles that contain additional permissions based on their job requirement.

There are two sublevels to this model including general and limited hierarchical RBAC. General provides support for an arbitrary partial order as the role hierarchy. This type of hierarchical RBAC provides access to all sub roles and sub permissions based on the primary role assigned to

a user. This can grant an extensive amount of access depending on the level of the role and the number of junior roles below it. Figure 2.21 shows and example of this general hierarchical model where User 1 is assigned to Primary Role A and thus is granted permissions to 3 Sub Roles. One of these Sub Roles can be assigned to another employee, where they would not inherit permissions of any equivalent roles, only those that are below it in the hierarchy.

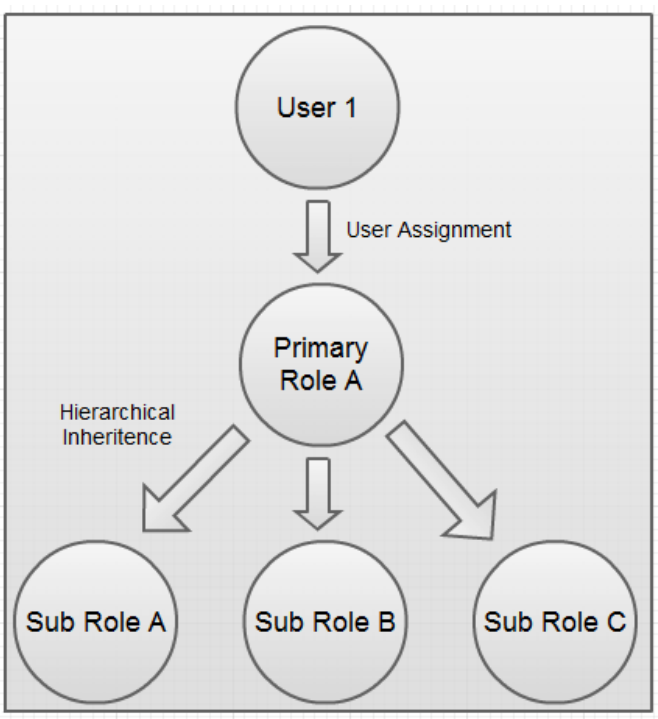

Figure 2.21 – General Hierarchical Role-Based Access Control Model

Limited Hierarchical RBAC provides natural blocks between roles so that one senior role will necessarily be granted access to all sub roles. This style of hierarchical RBAC was established to prevent the excess of role provisions based on large numbers of senior roles. This restriction would be similar to the structure of a tree where higher level senior roles don't necessarily inherit all access below it; instead they branch off based on job category or required duties. Figure 2.22 provides an example of limited hierarchical RBAC by showing that between two users, User 1 and User 2 they only inherit one Sub Role in common based on their primary role assignment. The other three Sub Roles are independent the inheritance as they may contain permissions that should not be granted through other roles.

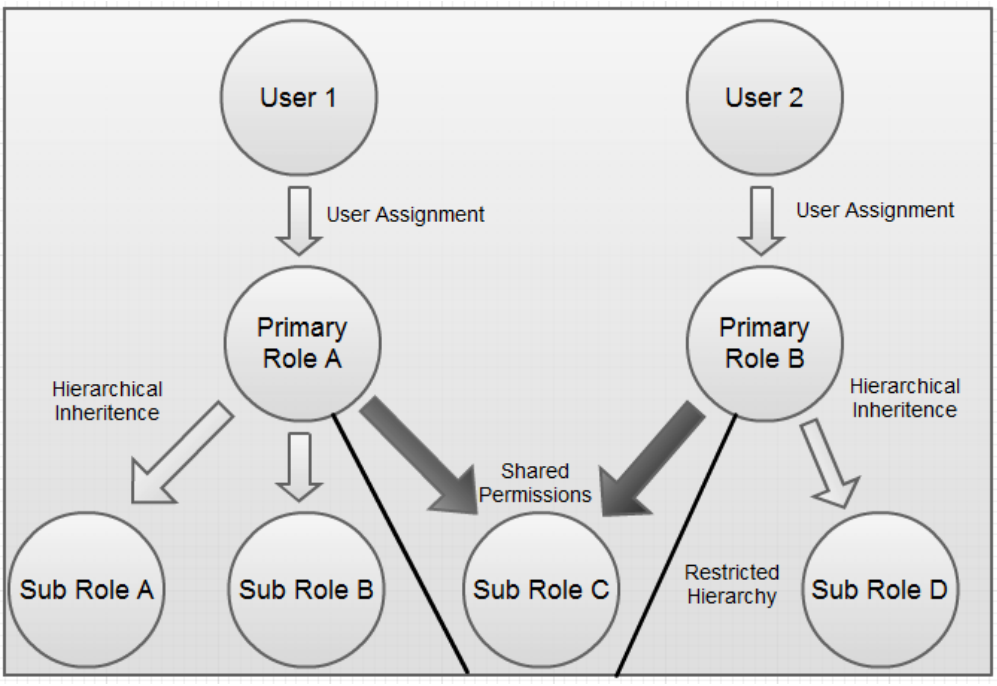

Figure 2.22 – Limited Hierarchical Role-Based Access Control Model

The idea behind this concept is simple, the more responsibilities that one individual gains typically means the more access they will need to complete their job. As users require this additional access they can be granted membership into higher level hierarchical roles to inherit additional permissions. It is not always appropriate to assume a user gaining privileges and access will need to maintain the access they currently have, and allowing them to inherit additional permissions above what they already have may be an improper assessment. The managers and administrators of the projects or assignments should not necessarily have all the access that their direct reports have as they are not involved with the project or task in the same way. This type of model can require the extrapolation of multiple roles and sub roles to create the proper division between users.

2.3 - Constrained RBAC

Constrained RBAC adds the ability to account for Separation of Duties or SOD, something that was not available in flat or hierarchical RBAC. The need to create a provision for organizations to utilize SOD is essential in order to create a division between job roles. This action can help to prevent one single individual from committing fraud or accidental damage because they have all the access that is necessary to do this. By creating roles that have this separation in mind the

responsibility to see actions to their end point is spread out among multiple users; thus increasing the amount eyes and hands on a particular task.

There are two types of constrained RBAC, the first is Static where the constraints to apply SOD rules are applied to the roles themselves. This means that enforcement of separation of duties requires that this be accomplished in the creation of the roles as well as the addition of members to these roles. For example, if a user is authorized to be a member of one role, they cannot be a member of another role due to separation requirements. Constraints of static constrained RBAC are inherited from the role so any sub roles or sub permissions are not hierarchically obtained. Figure 2.31 below shows that User 1 is able to be a part of Primary Role A, but because of separation requirements they cannot be part of Primary Role B. Hierarchical permissions from roles cannot be inherited since the assignment is restricted at the previsioning level.

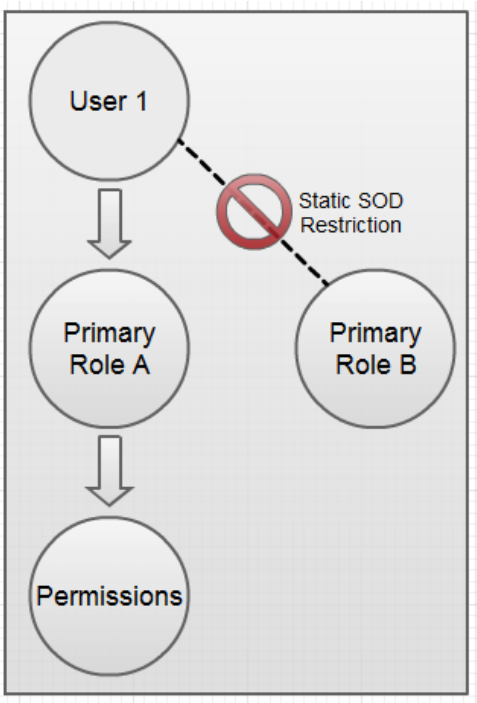

Figure 2.31 – Static Constrained Role-Based Access Control Model

The other type of constrained RBAC is dynamic, which must be established as part of the policy for the user when they are added to multiple roles. Dynamic Constrained RBAC can allow the granting membership to multiple roles that could violate the separation of duties principle, but the member of the multiple roles is not able to be part of all roles at the same time. Figure 2.32 below shows an example of this where User 1 is currently a member of three roles dynamically.

There only inherit the permissions from two of these roles as the SOD principle restricts all three roles from being invoked at the same time.

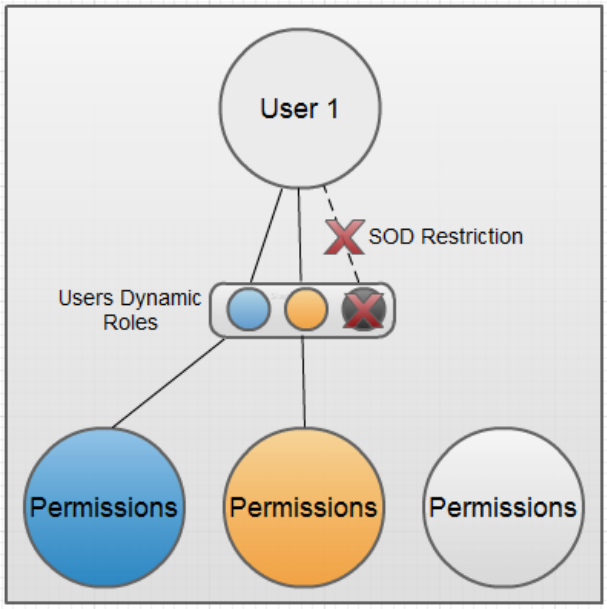

Figure 2.32 – Dynamic Constrained Role-Based Access Control Model

Implementing a constrained RBAC model requires that organizations have an in depth understanding of job specific functions. In order to statically or dynamically separate access between roles all permissions must be taken into account, especially those that may overlap to break the separation requirement. This in-depth knowledge requires additional administration and dedication to role based access control, but when it is established properly, it can greatly reduce the risk associated with role permissions.

2.4 - Symmetric RBAC

Access requirements change constantly within organizations, and with this change comes the importance to regularly review access that is contained in roles and thus by role members. The next stage above constrained RBAC to increase the security of the model is to add Symmetric Role-Based Access Control where the addition of regular role reviews is introduced. The purpose of this step is to enhance the security and access held in each role by regularly revisiting it to verify that it remains appropriate.

If reviews were not conducted on a regular basis, there would be stale access and members inside of each role potentially granting access to applications or functions that were now restricted. Symmetric RBAC begins to introduce the importance of a custodial figure that is ultimately responsible for the role and thus the members and permissions it grants access to. A custodian can be an individual close to the responsibilities of the role members, who has ultimate say in what access is granted to the role as well as approving or denying requests to be added to the role. Ultimately holding a custodian responsible can greatly reduce the size of each role as well as the risks associated with them.

3 Attribute-Based Access Control

A fairly new concept in the area of Role-Based Access Control is adding a layer of access restriction based on the enabled attributes of a user. This approach enhances the security provided by traditional RBAC; this method is called Attribute-Based Access Control or ABAC. Attributes can vary depending on criticality of access or attribute enforcement by organization. Attributes can be location based, time based, or require an additional verification before authentication is granted. This attribute provides organizations an extra level of security as well as additional audit capabilities. Figure 3.1 below provides a visual example of the basic concept of ABAC. A user is part of a role that is granting them permissions; however the attribute of the user will decide if the permissions are actually granted. The user can be either approved or denied this access based on their attributes.

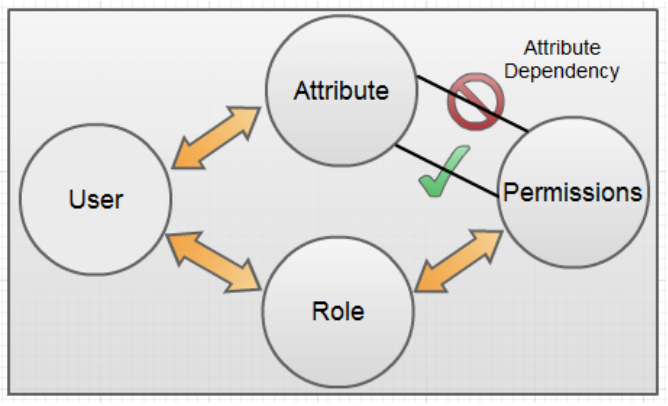

Figure 3.1 – Attribute-Based Access Control Dependency

Utilizing the addition of an attribute in conjunction with RBAC could be based on a large array of attributes. They could be location or timing restrictions, based on an additional password

authentication, a requirement to request access to an available role, answering a phone call when prompted, etc. The amount of attribute required will depend on the criticality of the asset.

For example, a technical support engineer needs to assist with a high priority issue with a server that has failed. This engineer does not have full-time access to this server due to its criticality. The engineer must first visit an internal website and invoke the role required gain these permissions. Additional comments are required in order to enable this access including a description of the problem and an incident ticket number of the ongoing issue. These additional pieces allow the attribute to grant the user the requested permissions. Another example involves a user who is attempting to access confidential documentation after hours at their organization. The established required attribute is that this data only be accessed during business hours. Despite having permissions to access this documentation through their role, this user will be denied this access based on the attribute requirement.

ABAC is a new concept that is gaining in strength when used in conjunction with RBAC. Requiring members to take additional steps for the most critical of access reduces the amount or mistakes or malicious activity in organizations. The introduction of ABAC into an existing RBAC model can enable organizations to provide fine-grain authentication and access control decisions based on detailed information about a user. The divisions can be created to restrict user's access based on a varying number and criticality of attributes. This enhancement to the role based access control model can greatly increase the security applied to existing RBAC structures.

4 Implementation

Organizations must consider multiple requirements in order to properly implement an Attribute-Based or Role-Based Access Control solution. This process can be initiated with the goal of providing least privilege access to users, maintaining separation of duty requirements and allowing ad hoc and periodic reviews of access and members. Since this process will be unique for every organization there is not one proper method to follow. The implementation process will depend on the number of responsibilities, duties and jobs within an organization along with the number of members. These factors combined will begin to determine the necessity of the number of roles that need to be created along with how many members should be assigned to each role.

It is also important to have an individual or group who can be responsible for the provisioning of access as well as the verification of the periodic review process. It can be helpful if this person is the same in both occasions to speed up the time needed to complete this process. Role necessity is a dynamic process and all members of an organization are not appropriate to have access to all roles. This is where a data custodian comes in to play who knows the data and the access granted by the role, they have the ultimate say in granting membership.

The implementation process can be complex depending on the size of the organization. Starting with blueprint of the end goal can help organizations continue to work in the right direction. The level of security required, number of roles needed and number of individuals will have a direct impact on the implementation planning. Once a solution has been implemented, it can then be passed onto a team dedicated to the audit and review of roles on a regular basis. All of these items working together can aid organizations in providing necessary access to critical personnel. This will reduce the impact of malicious actions and improve the experience of end users.

5 Management

In order to manage an Attribute-Based or Role-Based Access Control solution, there must be an established process put in place that is repeatable, sustainable and scalable. There are varying sizes of organizations, including the need to have a small or large amount of roles with a small or large number of members in each role. The management of the access authorization contained in the role as well as the abilities of the members to gain this access falls to a structured management plan. Successful management includes the ability to request access and then approve it from a custodial standpoint. This data custodian is the party who is capable of determining if the requestor should actually have the access they are requesting. The next part of this process is to have an automated system or team that has the ability to provision the access for the user.

Once access and members have been added to roles the memberships and authorizations must be held accountable to periodic reviews or audits to verify that the access existing remains relevant. This process to review access typically will fall on the custodian to certify the access they have granted throughout a certain time period is still applicable. This process can be done yearly where all access changes can be shown to the custodian and they can decide if it remains

appropriate. The process of granting access and reviewing access that has been granted requires a large commitment from the stewards of these roles, as well as the access inside of the roles. Utilizing a team of security professionals to determine what needs to be reviewed can be an essential part of the process to make the review easier for the data custodian.

The responsibility of access falls on the data custodians, who ultimately determine if access inside of a role is appropriate and if members requesting access to a role is appropriate. Without a custodian making this determination, there would be no ability to have ABAC or RBAC in an organization.

6 Closing

Proper implementation and management of an Attribute-Based or Role-Based Access Control solution can be a complimentary addition to any organizations information security structure. There are many different types of access control, but RBAC used in conjunction with ABAC can provide granular, manageable and auditable access to all individuals within an organization. This structured method, when implemented properly can greatly increase the security by providing least privilege access as well as a method to separate critical duties between users. With the support of data custodians and risk management teams the solution can be easy to use and understand as well as simple to audit.

The chosen implementation can be less or more secure depending on what the organization is interested in achieving based on their needs. Organizations can choose from flat, hierarchical, constrained or symmetric RBAC and then choose whether or not to add attributes to the requirements of an individual member. The possibility of role, permission and attribute combination selection are endless and can be tailored to fit each individual organization. Ultimately, utilizing ABAC and RBAC can be a welcome addition to any organizations risk management and access control strategies.

7 References

(1) D.F. Ferraiolo and D.R. Kuhn (1992) "Role Based Access Control" 15th National Computer Security Conference, Oct 13-16, 1992, pp. 554-563. http://csrc.nist.gov/rbac/ferraiolo-kuhn-92.pdf

(2) R. S. Sandhu, E.J. Coyne, H.L. Feinstein, C.E. Youman (1996), "Role-Based Access Control Models", IEEE Computer 29(2): 38-47 . http://csrc.nist.gov/rbac/sandhu96.pdf

(3) R. Sandhu, D.F. Ferraiolo, D, R. Kuhn (2000), "The NIST Model for Role Based Access Control: Toward a Unified Standard," 5th ACM Workshop on Role Based Access Control, July 26-27, 2000, Berlin, pp.47-63 . http://csrc.nist.gov/rbac/sandhu-ferraiolo-kuhn-00.pdf

Haythorn, Michael. "Best Practices, Procedures and Methods for Access Control Management." East Carolina University 1 (2013): 1-16. Print.

Kuhn, Richard, Edward Coyne, and Timothy Well. "Adding Attributes to Role-Based Access Control." IEEE Computer 43 (2010): 79-81. National Institute of Standards and Technology. Web. 12 Apr. 2014. http://csrc.nist.gov/groups/SNS/rbac/documents/kuhn-coyne-weil-10.pdf

Merkow, Mark. Information Security: Principles and Practices. 2nd. Indiana: Pearson Certification, 2014. Print.

S. Gavrila, J. Barkley, *"Formal Specification for Role Based Access Control User/Role and Role/Role Relationship Management"* (1998), Third ACM Workshop on Role-Based Access Control. http://csrc.nist.gov/groups/SNS/rbac/documents/design_implementation/gavrila-barkley-98.pdf

W.A. Jansen, *"Inheritance Properties of Role Hierarchies,"* 21st National Information Systems Security Conference, October 6-9, 1998, Crystal City, Virginia. http://csrc.nist.gov/groups/SNS/rbac/documents/design_implementation/pp-rbac-fin.pdf

"Attribute Based Access Control (Draft)." Building Block 1 (2014): 1-8. National Institute of Standards and Technology. Web. 12 Apr. 2014. http://csrc.nist.gov/nccoe/Building-Blocks/NCCoE_ABAC_Building_Block_Draft_20140221.pdf

"How to Succeed With Role-Based Access Control (RBAC)." Security Compliance Corp. Security Compliance Corp. Web. 13 Apr 2014. http://www.securitycompliancecorp.com/RBAC-SCC-InsecureMag.pdf.