Role-Based Access Control

GIAC Security Essentials Certification (GSEC)

Practical Version 1.4b

Option 1

November 8, 2003

Steve Frame

Table of Contents

Abstract

This paper will describe the first RBAC model presented in December 1992, as well as detail the later aspects of the model as it matured and took on new components. Then, using the components of the RBAC model will set up users and roles based on a file system and applications of a fictitious company.

Role based access can help solve the administrative problems of current access systems. By implementing RBAC policies a company can administer their access requests faster, with fewer errors and fewer administrators.

Introduction

Corporations rely heavily on information processing systems to meet their operational and financial goals. The integrity, availability and confidentially of systems and data are critical to the operation of the organization.

The protection of Corporate computing assets is a main concern of the security administrator. The number of administrators, the number of help desk personnel needed and the complexity of the administration increase with the size of the organizations. Historically, ACL's or access control lists, group based access and explicitly granted accesses have been the mainstays of access control.

Security Administration with RBAC consists of determining the operations that must be executed by persons in a particular job and assigning employees to the proper roles.

RBAC Beginnings

Role Based Access Control (RBAC) was formally introduced by David F. Ferraiolo and Richard Kuhn at the 15th National Security Conference in October 1992. The presentation introduced the idea that a users access to the resources of a company should be determined by the role the individual plays in the organization. The main motivation behind RBAC is to specify and enforce enterprise specific security policy so that it follows the organizations structure.
The first model combined several existing and emerging concepts i.e. OS user groups, BMS privilege groups.

The basis of RBAC is the concept of the role. A role is type grouping that categorizes subjects based on various properties. These properties pertain to the functional responsibilities of the user in the organization.

There were problems identified with early RBAC models. There were no provisions for separation of duties. Separation of duty is when a user has the authority to perform two or more functions that taken together would create a conflict of interest. Provisions for separation of duty were added as components in later models. A situation called role precedence was also identified. Much of RBAC modeling is set based. Using set terminology when two roles intersected or are applied at the same time, which role's permissions would take precedence?

The problems of separation of duty and role precedence are related to authorized role set. The idea is to limit a users role usage to a subset of the roles from his authorized role set. This would become his active role set.

Role activation also provides natural mechanism resolving role precedence, in case of conflict between roles; active roles take precedence over inactive roles.

.

Role Based Access is an Alternative to the more traditional forms of access. There are 3 basic access control models.

1. Discretionary Access Control (DAC) is usually identity based. This is defined by the system administrator and based on the users needs.

2. Mandatory access control (MAC) is found primarily in the military or other highly sensitive systems. This is based on access to objects based on the sensitivity of the data and the matching level of authorization. A user is granted access only when the user and object have corresponding clearance levels.

3. Non-Discretionary Access Control is usually role based, centrally administered with authorization decisions based on the roles individuals have within the organization. The system Administrator grants or revokes system privileges based on the users role.[1]

Core RBAC

Users being added to roles and permissions being assigned to those roles can characterize the RBAC model. A role is a method of naming the many to many relationships among individual users and permissions. The core model also specifies a set of sessions, which are defined as the relationship between the user and the activated subset of roles.

- Many to many relationships among individual users and privileges
- Session is a mapping between a user and an activated subset of assigned roles.
- User/role relations can be defined independent of role/privilege relations.
- Privileges are system/application dependent
- Accommodates traditional group-based access control
- Each role may be authorized to perform one or more transactions.

[1] http://www.peaccfulpacker.com/it_solutions/xias0107.htm

- Users may execute transactions.

3 basic rules are required:

1. Role assignment - a user can execute a transaction only if the user has selected or been assign a role.
2. Role Authorization - a subjects active role must be authorized for the subject.
3. Transaction authorization - a subject can execute a transaction only if the transaction is authorized for the subject's active role.[2]

Hierarchal RBAC

This RBAC component is a natural means of structuring roles to reflect an organization's line of authority and responsibility.

Role/role relation defining membership and privilege inheritance
Reflects organizational structure and functional delineations
Two types of hierchies;
Limited hierarchies – these are imposed restrictions which would result in a flat structure
General hierarchies – Role can contain a role

Static Separation of Duties (SSD)

SSD policies deter fraud. Collaboration has to exist between job related capabilities for SSD to fail. Separation of duties requires that no single individual be able to execute all the transactions within a set. The most obvious example is payment authorization, and actual disbursement of the payment.[2] SSD can be handled with policy. If the user is in one role then the user cannot be in another role that will create a conflict.

Dynamic Separation of Duties (DSD)

DSD also deters fraud. This component would restrict the combination of roles that, when activated during a session, would create a conflict of interest for the user. These situations would be difficult to identify without a thorough understanding of the current set of access controls and permissions.

[2] http://Hissa.ncsl.nist.gov/rbac/paper/rbac1.html

Planning for RBAC

Administrators will have to continue to perform their daily work and also be available to work in the construction of new roles. There will be possible changes to the file system and renaming of groups to new naming standards, all of which can have a profound affect on the daily operation of the company if incorporated into the production environment incorrectly.

The time and disruption it would take for a company to recreate file systems to make the most efficient use of a pure role based system is more than most companies could or want to commit to. All of the RBAC features do not have to be instituted at one time. Many of the RBAC components are not necessarily suited for every company.

Role based access can be factored into a current setting. Old groups can be used if they provide the proper access to the necessary resources by making it a member of the role. If explicit rights are granted to a user for access to a folder or file, this access should be moved to the role.

Planning for the project should include a thorough understanding of the current access control setting. It goes without saying that the target department contacts and corporate support can provide invaluable information on the current situation. Existing groups should have correct mappings documented. Obsolete systems that have been removed often have empty groups that were once associated with the obsolete application. This amounts to debris in the system and should be removed. Groups that "no one knows anything about" can have a few members removed with their management's permission and see if the user's performance is impacted in any way. If they can be deleted, delete them. You should eventually end up with a clean directory structure that has only valid elements. Management of resources would be more efficient.

Users at XYZ Corporation

The latest model will be applied to a company with an existing file structure. The fictitious files will reside on servers, which are currently in operation. The operating environment is Microsoft Windows.

The following will be a representation of the creation of Roles for four individuals in the Accounting department of XYZ Corp.

- Jim Smith - Mr. Smith is a member of senior management with responsibilities that exceed the boundaries of the accounting department. He is a member of top-level committees with officers from all functional areas of the company.

- Will Jones - Mr. Jones is the department manager. He has the responsibility of daily management of the department. He has the authority to make changes to accounting databases.

- Betty White - Ms. White is the supervisor of the payables section. She has the responsibility of the daily operation of the payables section.

- James Brown - Mr. Brown is a clerk in the payables section.

Basic Policies

The following policies are necessary for the Administration of access control to work more efficiently.

1. Access control managed centrally in the organization - All requests for access changes routed from user to one team of administrators and provision for those administrators to notify the requestor when complete.

2. All changes made for access control are made by the same administrative group - make sure there is only one group making changes to access control of company resources.

3. Policy for access control is published and known by administrative security staff. Policy to maintain Static and Dynamic separation of duty must be known by the security staff.

4. Enlist help from the departments - Users from each functional area of the organization to assist with department user needs and communication with the security administration group.

ROLES for Accounting Staff

RBAC-AC-Vice-President

The Vice President role would receive the permissions to all the resources necessary to perform this job. Within the Accounting department, the Vice President (VP) role could need access to the applications used to perform the daily accounting functions. The VP Role would also need access to applications that run on machines outside the department and reports generated by applications that might be written to disk outside the accounting department. During the creation of the role, policy should be developed to prohibit any conflict of interest that might occur because of permissions granted the VP role.

RBAC-Senior-Management

This type of role would contain all the individuals in the firm designated as senior management. Roles of this nature would provide access to resources common to senior management.

RBAC-AC-Manager

The Accounting Manager (AM) role could be included in every application in the department. However, the permissions granted this role could vary considerably from one application to another. The AM role would make direct entries into the general ledger database but only need read and execute access to other applications. The AM role could also have sole access to sensitive data such as department salaries or determine who get bonuses. These policies and permissions would require careful planning. Once the role is complete, it would need little maintenance.

RBAC-AC-Payables-Supervisor

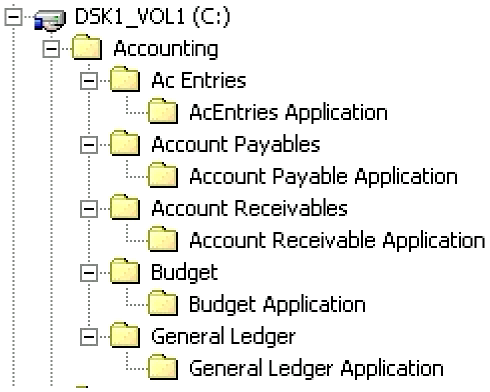
This Payables role provides the access necessary to operate the payables area. The supervisor would have access to the Payables application. The Payables role would also have access to applications like attendance that would be located on a Human Resources server.

RBAC-AC-Payables-Clerk

The payables clerk role would have access to the payables application. This application would have databases located on an application server like Microsoft's SQL Server. The role name with the domain prefix (<Domain Name>\RBAC-AC-Payables-Clerk) could be used as a login name, added under NT authentication and function just as a single user.

Each of these roles would be created using user manager in the company Domain as Global groups.

The following picture is an example of how the accounting department would be constructed by RBAC standards. The structure is somewhat vertical, but follows the organizational structure of the department, each functional area answering to the department head.



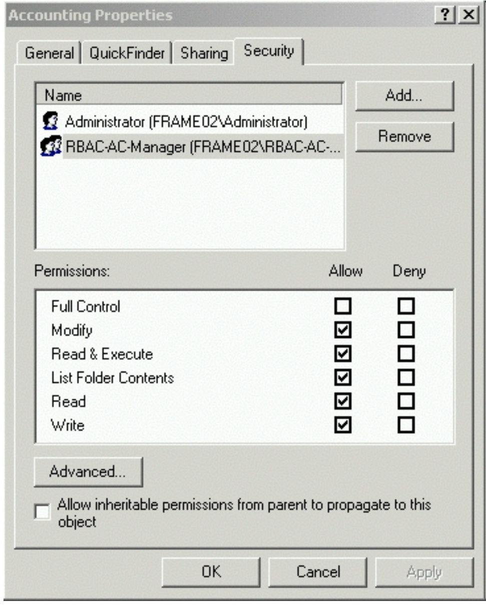
```
DSK1_VOL1 (C:)
    Accounting
        Ac Entries
            AcEntries Application
        Account Payables
            Account Payable Application
        Account Receivables
            Account Receivable Application
        Budget
            Budget Application
        General Ledger
            General Ledger Application
```

In reality these accounting functions could be scattered over several servers throughout the company. All departments make budget entries as well as general ledger entries. This situation is not a problem with RBAC. Simply map the role to the resource. The role will not have the use of the Role Hierarchy component of RBAC.

Now that we have created the roles and defined the file structure to be secured, we can apply permissions to the roles.

The Accounting Manger role has been given full access (Figure 1) at the Accounting folder. The role has the necessary permissions to read, write, execute and delete. The only function he cannot perform is to give these rights to someone else. The full control box is not checked.

Figure1



Accounting Properties

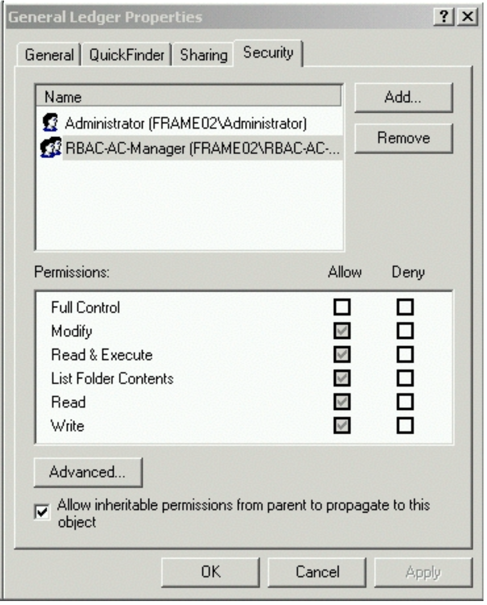
General | QuickFinder | Sharing | Security

Name
- Administrator (FRAME02\Administrator)
- RBAC-AC-Manager (FRAME02\RBAC-AC-...

Permissions:	Allow	Deny
Full Control	□	□
Modify	☑	□
Read & Execute	☑	□
List Folder Contents	☑	□
Read	☑	□
Write	☑	□

□ Allow inheritable permissions from parent to propagate to this object

Figure 2



General Ledger Properties

General | QuickFinder | Sharing | Security

Name
- Administrator (FRAME02\Administrator)
- RBAC-AC-Manager (FRAME02\RBAC-AC-...

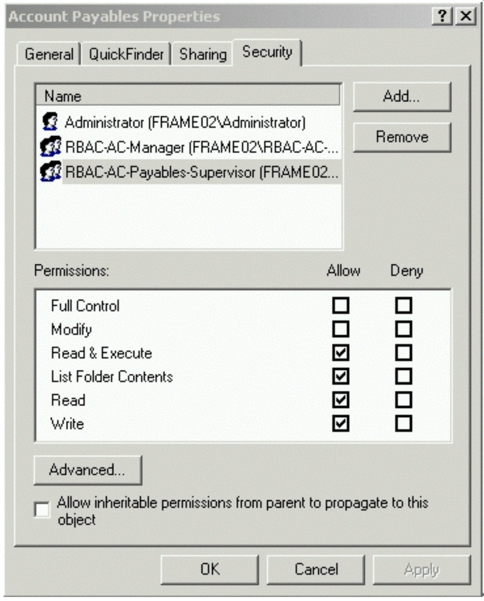
Permissions:	Allow	Deny
Full Control	□	□
Modify	☑	□
Read & Execute	☑	□
List Folder Contents	☑	□
Read	☑	□
Write	☑	□

☑ Allow inheritable permissions from parent to propagate to this object

Notice in Figure 2 how the permissions have been inherited to the last area, general ledger. The Accounting Manager role has the same level of access in all areas of the accounting department.

Below (Figure 3) is the mapping of the RBAC-AC-Payables-Supervisor role to the Payable folder. It has been determined that the role should not be able to delete files from the payables area so the modify box is left unchecked.

Figure 3

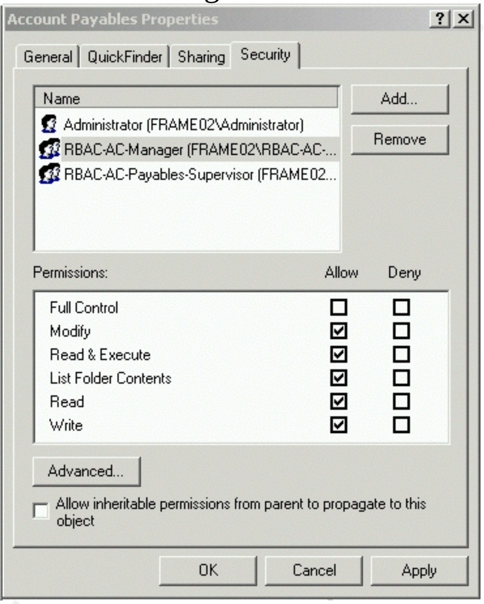


Name
Administrator (FRAME02\Administrator)
RBAC-AC-Manager (FRAME02\RBAC-AC-...
RBAC-AC-Payables-Supervisor (FRAME02...

Permissions:	Allow	Deny
Full Control	□	□
Modify	□	□
Read & Execute	☑	□
List Folder Contents	☑	□
Read	☑	□
Write	☑	□

□ Allow inheritable permissions from parent to propagate to this object

Figure 3a



Name
Administrator (FRAME02\Administrator)
RBAC-AC-Manager (FRAME02\RBAC-AC-...
RBAC-AC-Payables-Supervisor (FRAME02...

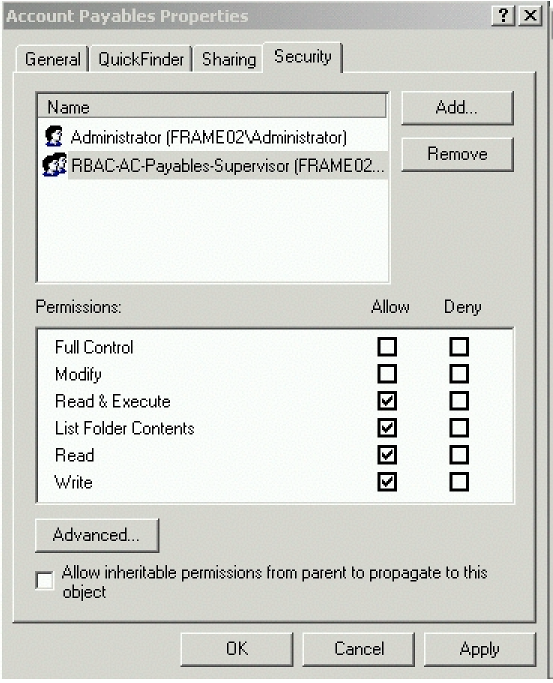
Permissions:	Allow	Deny
Full Control	□	□
Modify	☑	□
Read & Execute	☑	□
List Folder Contents	☑	□
Read	☑	□
Write	☑	□

□ Allow inheritable permissions from parent to propagate to this object

Notice that the RBAC-AC-Manager role is present (Figure 3a). It has been determined during research for the construction of the role that no one should have read, write, execute and delete to both the Account Payables and Account Receivables functions. The Manager role currently has full access to both areas.

This is how we would set the policy: First, click on and remove the check from the box "Allow inherited permissions from parent to propagate to this object" and remove the role (Figure 4). Then re-add it (Figure 5) giving the RBAC-AC-Manager role only read and execute to the Account Payables folder (Figure 6).
This will remove any conflict of interest created by having full rights to both objects.

Figure 4



Account Payables Properties

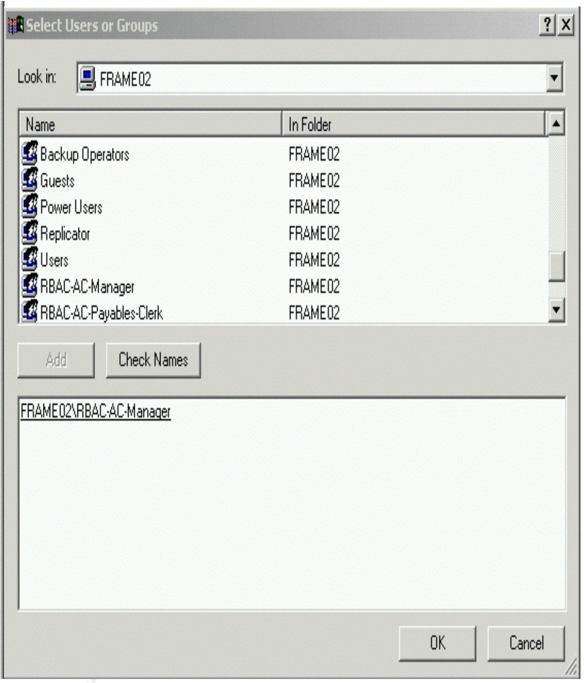
General | QuickFinder | Sharing | Security

Name
Administrator (FRAME02\Administrator)
RBAC-AC-Payables-Supervisor (FRAME02...

Permissions:	Allow	Deny
Full Control	□	□
Modify	□	□
Read & Execute	☑	□
List Folder Contents	☑	□
Read	☑	□
Write	☑	□

□ Allow inheritable permissions from parent to propagate to this object

Figure 5



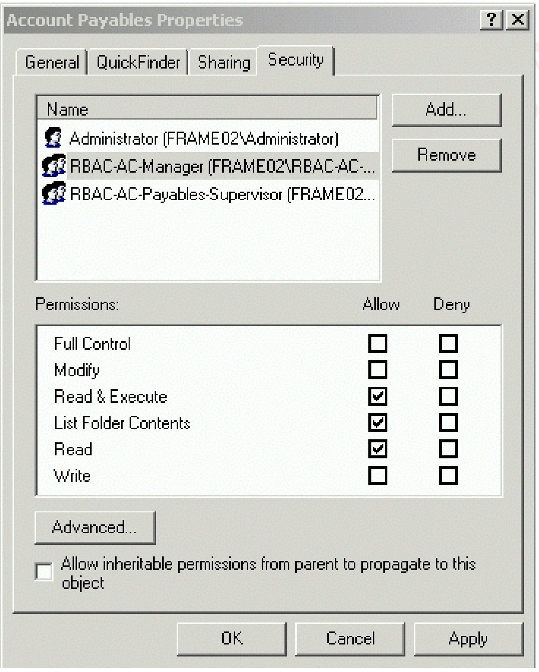
Select Users or Groups

Look in: FRAME02

Name	In Folder
Backup Operators	FRAME02
Guests	FRAME02
Power Users	FRAME02
Replicator	FRAME02
Users	FRAME02
RBAC-AC-Manager	FRAME02
RBAC-AC-Payables-Clerk	FRAME02

FRAME02\RBAC-AC-Manager

Figure 6



Account Payables Properties

General | QuickFinder | Sharing | Security

Name
Administrator (FRAME02\Administrator)
RBAC-AC-Manager (FRAME02\RBAC-AC-...
RBAC-AC-Payables-Supervisor (FRAME02...

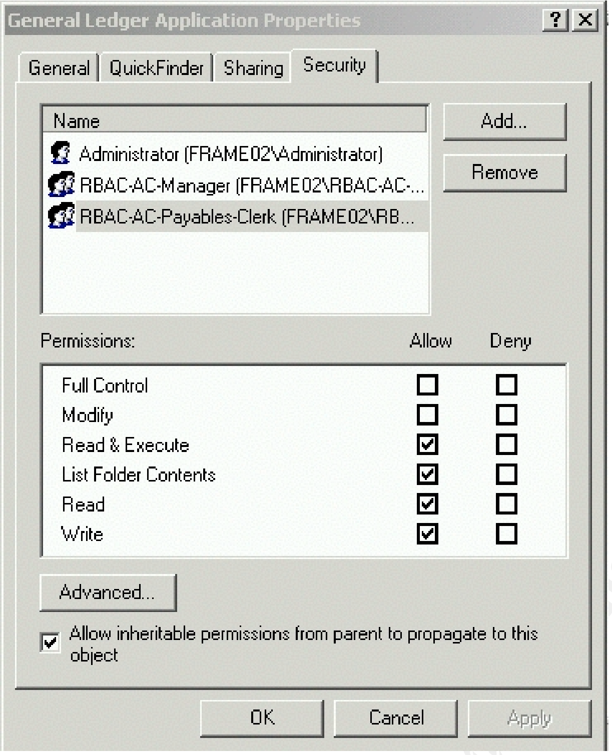
Permissions:	Allow	Deny
Full Control	□	□
Modify	□	□
Read & Execute	☑	□
List Folder Contents	☑	□
Read	☑	□
Write	□	□

□ Allow inheritable permissions from parent to propagate to this object

This treatment of the Accounting manager role is an example of static separation of duty and one way to solve the problem.

The RBAC-AC-Payables-Clerk role is responsible for running the payables application. The role will require read and execute to the Payables application folder. This application also writes a file that is input to the General ledger application folder. The RBAC-AC-Payables-Clerk role will need write access to the General Ledger Application Folder. Add the role to the General Ledger Folder and click on write then apply. Figure 7 shows the RBAC-AC-Payables-Clerk with read, write and execute to the General Ledger Application folder.

Figure 7



Name
- Administrator (FRAME02\Administrator)
- RBAC-AC-Manager (FRAME02\RBAC-AC-...
- RBAC-AC-Payables-Clerk (FRAME02\RB...

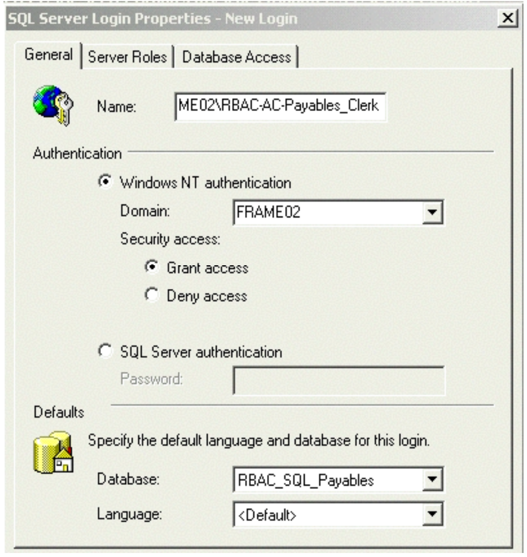
Permissions:	Allow	Deny
Full Control	□	□
Modify	□	□
Read & Execute	☑	□
List Folder Contents	☑	□
Read	☑	□
Write	☑	□

☑ Allow inheritable permissions from parent to propagate to this object

The primary source of data for the Account Payables application is located on a SQL server. The role needs read and write permission to the RBAC_SQL_Payables database. The RBAC-AC-Payables-Clerk role can be added to a SQL environment.

Figure 8 is the SQL Server Login screen for new logins. In this case, the Domain name is FRAME02. The role name is added in the Name field after the Domain name (Frame02\RBAC-AC-Payables-Clerk). The role will gain access to the SQL environment based on its NT credentials provided the role is a valid member of the domain. Notice Figure 9 is the same SQL Server Login Properties screen only here the Database Access tab is selected. Access is needed to the RBAC_SQL_Payables database. The database name is checked in the upper area and db_datawriter and db_datareader are checked in the lower area.

Figure 8



SQL Server Login Properties - New Login

General | Server Roles | Database Access

Name: ME02\RBAC-AC-Payables_Clerk

Authentication

● Windows NT authentication

Domain: FRAME02

Security access:

● Grant access

○ Deny access

○ SQL Server authentication

Password:

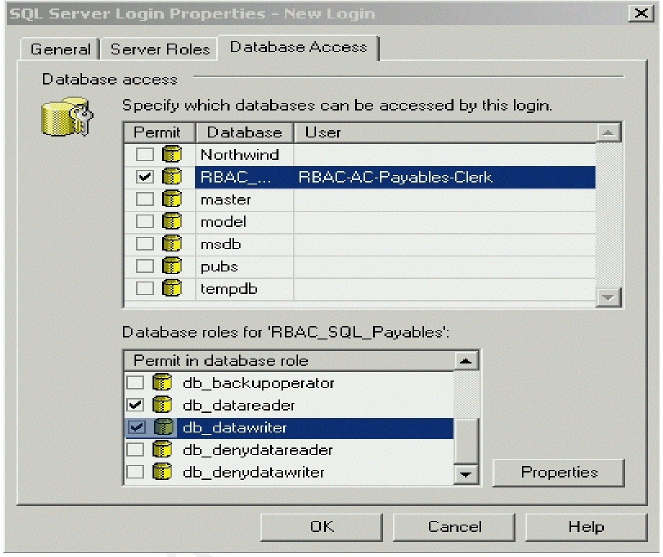
Defaults

Specify the default language and database for this login.

Database: RBAC_SQL_Payables

Language: <Default>

Figure 9



SQL Server Login Properties - New Login

General | Server Roles | Database Access

Database access

Specify which databases can be accessed by this login.

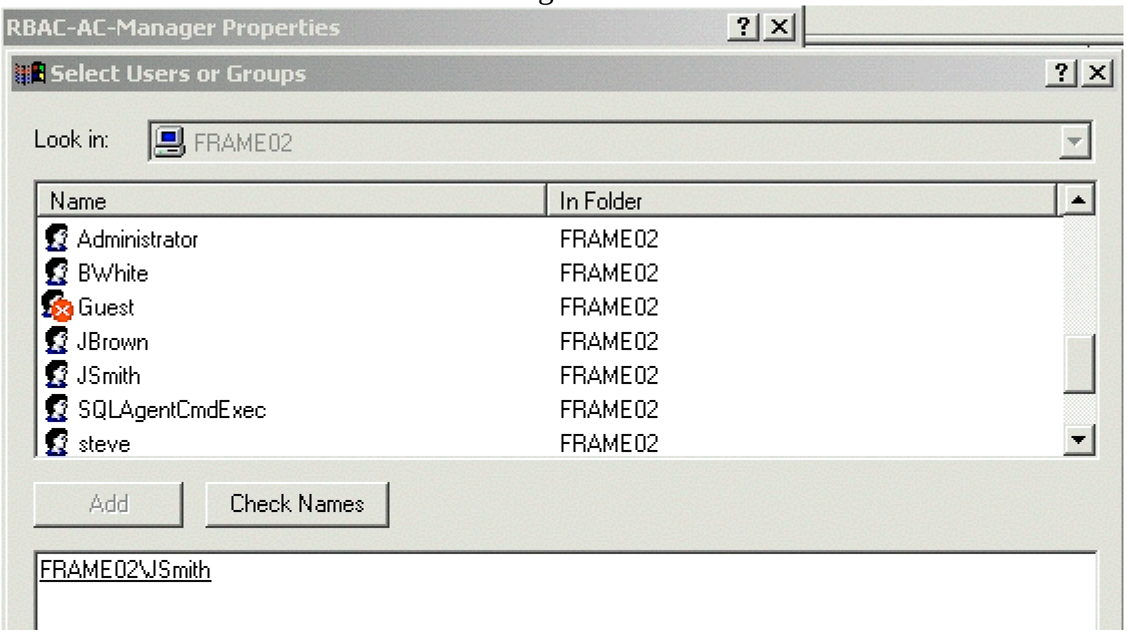
Permit	Database	User
□	Northwind	
☑	RBAC_...	RBAC-AC-Payables-Clerk
□	master	
□	model	
□	msdb	
□	pubs	
□	tempdb	

Database roles for 'RBAC_SQL_Payables':

Permit in database role	
□	db_backupoperator
☑	db_datareader
☑	db_datawriter
□	db_denydatareader
□	db_denydatawriter

After creating the roles and assigning permissions, the only thing left to do is add the users. Notice that the role and permission assignments were created independently of assigning users to the role. This function could be performed by another group of administrators.

Figure 10



RBAC-AC-Manager Properties

Select Users or Groups

Look in: FRAME02

Name	In Folder
Administrator	FRAME02
BWhite	FRAME02
Guest	FRAME02
JBrown	FRAME02
JSmith	FRAME02
SQLAgentCmdExec	FRAME02
steve	FRAME02

FRAME02\JSmith

The last step is to add the users to the role (Figure 10). Mr. Smith now has all the permissions that were given to the role.

Conclusion

RBAC can help control the burden of access control administration. Staff can be reduced and still perform at the same level. The level of complexity of security administration can be reduced as each organizational unit is moved to role-based environment. RBAC components can be implemented without a major overhaul of the file systems that they impact. Many major vendors have included RBAC functions into their products.

The current trend in corporate America is to do more with less. Role-Based Access can play a large part in attaining that goal by simplifying the Security Administration process

Appendix

"Access Control Models". http://www.peacfulpacker.com/it_solutions/xisa0107.html.

Covington, Michael J., Matthew J. Moyer and Mustaque Ahamad. "Generalized Role-Based Access Control for Securing Future Applications". http://www.cc.gatech.edu/~covingto/portal/ pubs/covington-2000-nissc.pdf.

Christodorescu, Mahai. "Mahai's Summary for Role-Based Access Control". http://www.cs.wisc.edu/~mihai/my_misc/Summaries/2003-09-10_html .

Ferraiolo, David, Rick Kuhn, and Ravi Sandhu. "Proposal for Fast-Tracking NIST Role-Based Access Control Standard". http://csrc.nist.gov/rbac/rbac-std-proposal.ppt .

Ferraiolo, David and Rick Kuhn. "Role-Based Access Controls". http://hissa.nist.gov/rbac/paper/rbac1.html.

Ferraiolo, David, Ravi Sandhu, and Serban Gavrila. "A Proposed Standard for Role-Based Access Control". http://csrc.nist.gov/rbac/RBAC-std-draft.doc.

Kuhn, Richard. "Role Based Access Control". http://csrc.nist.gov/rbac/rbac-std-ncits.pdf.

Sandhu, Ravi. "Future Directions in Role-Based Access Control Models". http://list.gmu.edu/confrnc/misconf/mms0l-rbac-future.pdf

"Vendors and System Integrators on Role-Based Management". http://www.eurekify.com/rbac_vendors.htm.