

# 5 key steps to defining your application-access control

*An Evidian white paper*

**EVIDIAN**  
A Groupe Bull Company

***Evidian Professional Services***

**Version 1.0**

## **Summary**

- How to control access to your applications?
- From RBAC to Extended RBAC
- Top-Down or Bottom-Up approach
- The 5 key steps to defining an extended RBAC policy using a Bottom-Up approach
- Extended RBAC / Bottom-Up approach at the service of your organization

© 2007 Evidian

*The information contained in this document represents the view of Evidian on the issues discussed at the date of publication. Because Evidian must respond to changing market conditions, it should not be interpreted as a commitment on the part of Evidian, and Evidian cannot guarantee the accuracy of any information presented after the date of publication.*

*This is for informational purposes only. EVIDIAN MAKES NO WARRANTIES, EXPRESS OR IMPLIED, IN THIS DOCUMENT.*

*We acknowledge the rights of the proprietors of trademarks mentioned in this book.*

# Table of contents

---

How to control access to your applications?.....	4
The 3 branches of identity and access management .....	4
Defining a policy .....	4
From RBAC to Extended RBAC.....	5
Limits of the RBAC approach .....	5
The extended RBAC model .....	6
Extended RBAC rules .....	6
Top-Down or Bottom-Up approach.....	7
Creating an access-security policy .....	7
Comparison of Top-Down and Bottom-Up policies .....	7
The 5 key steps to defining an extended RBAC policy using a Bottom-Up approach.....	8
Step 1 - Collecting information about users' real accesses to systems and applications .....	8
Obtaining the list of real accesses .....	8
Step 2 - Associating a role and an organization with each user .....	9
Defining roles and their dependencies .....	9
Defining organizations .....	9
Associating a user with an organization or a role .....	10
Step 3 - Simplifying the model .....	10
From a multi-profile user to a single-profile user ....	10
Associating the right applications with the right profiles.....	11
Selecting only significant organizations .....	11
Grouping applications by family .....	11
Role mining for complex organizations .....	11
Step 4 - Determining the security policy .....	11
Creating a security policy .....	12
Reconciling the data with an existing policy .....	12
Step 5 - Reconciling the policy with the policies defined in applications and systems .....	13
Creating a link between a right and a user .....	13
Reconciling the policy with the target applications and systems .....	13
Extended RBAC / Bottom-Up approach at the service of your organization.....	14

## How to control access to your applications?

---

### The 3 branches of identity and access management

Identity and access management can be divided into three sub-branches:

**Access management:** controlling how a user logs on to his or her applications. This branch handles issues like authentication, filtering accesses according to users, the authentication method used, workstation type and location, or even access delegation or centralized-access audit.

**Identity management:** emphasis here is mainly on the creation of user-identity attributes. It is all about a user's job attributes, which define his or her position in the company and appear in the company's LDAP directory, as well as the user's technical attributes declared in the information systems.

**Role management:** defining an access policy and applying it to access and identity management.

### Defining a policy

Over the past few years, policy definition has been considered as increasingly critical. Due to new regulations and past identity and access-management projects, IT companies are focusing more and more on creating and implementing security policies.

Role-assignment-approval **workflow**<sup>1</sup> may be part of the policy process. For instance, a workflow enables you to validate a user's job description - after which a policy will be applied in order to define the associated access rights.

With time, IT departments very quickly focused their attention on user, application and permission-management models.

Most policy models are based on the notion of **RBAC**<sup>2</sup>. This consists in assigning each user a role which corresponds to his or her tasks in the organization. It should be possible to make the policy comprehensible, applicable and maintainable by standardizing these roles and assigning them to a set of users.

---

<sup>1</sup> Workflow: step-by-step validation processes. Refers sometimes to the computer tool used to automate these processes or all the processes defined inside an organization.

<sup>2</sup> Role-Based Access Control.

## From RBAC to Extended RBAC

---

### Limits of the RBAC approach

The RBAC model has quickly shown its limits in the face of organizations' realities.

The main problems result from standardizing the roles. In fact, to be effective, the RBAC approach requires the entire organization to share the same working processes, the same set of tasks and the same role definitions.

Now, what makes a good organization is the fact that tasks are optimized at the operational level. It is at the level closest to field workers that tasks are assigned according to talents, capacities and even each person's desire, to enable the team to attain its objectives in the most effective manner possible.

The general management may offer a task and role-definition framework, but it is inside the field workers' team that the real focus of each definition is determined.

To meet this operational need, the RBAC models must gradually include more and more specific roles, to finally become too complex.

To create a logic that cuts across the entire organization, some approaches propose heuristic methods which analyze existing rights in target systems and propose some roles. This also comes up against some implementation problems:

- *How to ensure proper association between a user and his or her login in a system?*  
Login-creation rules may vary from one system to the other. In some cases, a login may not have any logical relation with its owner's name.
- *How to manage shared accounts?* These accounts, generally used by members of the same team, are not associated with a specific user. Doing away with such accounts is basically one of the main targets of an access-security policy. Therefore, they cannot be excluded from the project.
- *How to delete assignment errors made in target systems?*  
Unfortunately, target systems and applications have a role and access-rights logic that is marred by errors. The existing policy may be loose-knit because no security study had been conducted. There may be some 'dormant' accounts for users who have changed jobs and should no longer access their initial applications. Finally, exceptional rights, created as a result of an urgent request, may remain in the system because their deletion has never been asked for...

The result of a global, target-system analysis may lead to a transfer of the flaws and errors declared in the systems and applications to the general policy.

## The extended RBAC model

There is a model which takes account of the specific characteristics of each profile inside each organization: the **extended RBAC** model. This model simply takes into account the user's organization in the same capacity as his or her role (or task).

It takes into consideration each organization's definitions of user roles (tasks).

For instance, the role "*Salesperson*" covers sales functions only in Germany where the enterprise is well established. But the same "*Salesperson*" role may include sales AND operational marketing functions in Italy where canvassing takes up most activities. In Belgium, "*Operational Marketing*" may also comprise a "*partner relations development*" function.

Organization-based user role definition enables you to adapt the model to your organization's operational realities (roles) and organizational realities (structures and, thus, working methods).

## Extended RBAC rules

An extended RBAC policy uses two basic rules:

- **General rules**, which associate the organization/role pair with an application, a system, a resource or even a specific authorization<sup>3</sup>.
- **Exceptions**, which associate a given user with an application, a system, a resource or even a specific authorization.

A properly applied policy must reduce the exceptions to a bare minimum and limit the number of general rules while perfectly covering the company's access policy.

---

<sup>3</sup> An authorization is used to define the parameters which will enable a user to perform a specific transaction such as 'processing orders covering over 1 million euros', 'reading VIP-customer data', or even 'modifying customer details'.

## Top-Down or Bottom-Up approach

---

### Creating an access-security policy

Up till now, the most common access-security policy approach has been the Top-Down approach.

This approach starts either with the definition of an access-security model that is consistent with an organization's activities and information system, or with an analysis of the rights defined in systems and applications.

Another approach does exist, in which field activities and access realities are used as basis. This approach, known as the Bottom-Up approach, consists in simply collecting information about users' access to their applications over a given period and automatically associating each user with the login he or she uses to access his or her application.

After the information about users' real accesses is collected, users are associated with their role and organization. An optimized policy is deduced from this. This policy can then, through the users' login, be compared with the rights declared in the target systems so as to analyze the differences and determine the necessary adjustments.

### Comparison of Top-Down and Bottom-Up policies

Let us take as reference the ideal company policy which allows a right (or permission) to be associated with a user.

- The Top-Down approach, which is based on the rights declared in systems and applications, generates a policy that is *more permissive* than the target policy, because this approach incorporates the flaws existing in the systems and applications.
- The Bottom-Up approach, based on users' real accesses over a given period, generates a policy that is *more restrictive* than the ideal policy because this policy does not take into account users' accesses outside the said period.

By comparing the differences between the Bottom-Up policy and the Top-Down policy, we can focus the analysis on points that would enable us to get close to the ideal policy.

## The 5 key steps to defining an extended RBAC policy using a Bottom-Up approach

The five key steps to defining an extended RBAC policy using a Bottom-Up approach are:

1. Collecting information about users' real accesses to systems and applications
2. Defining user attributes
3. Simplifying the model
4. Creating a policy from users' real accesses
5. Reconciling the policy with the existing model or the declarations made in the target systems and applications.

### Step 1 - Collecting information about users' real accesses to systems and applications

#### Obtaining the list of real accesses

Collecting information about **users' real accesses** to systems and applications is the basis of the Bottom-Up approach. At the end of this step, you will have a list of the application accounts that are actually used, and the names of those who use them.

This list will enable you to define a policy that reflects the reality of accesses, and to do away with errors hidden inside the applications and systems.

This information should be collected over a representative period of time. The period should be long enough (at least four months), to cover the company's "respiration" cycle, its operational-production and analysis and reporting periods.

Where to find the information

You can collect this information with the help of a self-registration tool installed on the users' PC. This tool detects the display of an application's login window, prompts the user for his or her login and password, enters this information in the login window and checks that the connection is accepted.

You can also use the information provided by an enterprise SSO, which keeps trace of all accesses made.

This collection will enable you to associate each user's access with the login he or she uses.

Thanks to this collection, you can obtain the following relations:

user ↔ application access ↔ user login (an account)

## Step 2 – Associating a role and an organization with each user

At the end of this step, each of your users will have one or more roles and belong to specific places in your organization. A user may belong to more than one place in your organization, and have different roles in each place.

### Defining roles and their dependencies

Roles are simply defined by their name. They may have one or more parent roles from which they inherit access rights within the same organization.

Thus the role "*Sales Manager*" may have two parent roles which are "*Sales*" and "*Manager*". It will thus give access to leave-validation applications ("*Manager*" role) and sales-management applications ("*Sales*" role).

All the roles are attached to a generic role "ALL\_ROLE", which will cover accesses that do not depend on roles but on organizations.

The only constraint inherent in the description of roles is that the parent/child relationship should not lead to a circular path.

#### How to define roles

The definition of these roles is far less critical than in the RBAC model. In fact, the target policy will define the roles according to the user's organization. Therefore, it is not mandatory to define a different role for each function inside the organization.

Where to find the information

Just take a list of roles that is extensive enough to cover the needs of the target organizations, and confirm it with your Human Resources Department or Information System Department. This type of codification exists already in administrative organizations which publish statistics or process business-related data.

### Defining organizations

Organizations follow a hierarchy pattern the top of which is the entity we wish to analyze. This entity may be a company, a department, etc.

For instance, we may have:

- Company/R&D/Development/
- Company/R&D/Validation/
- Company/Sales/Asia/
- Company/Sales/America/North America/
- Company/Sales/America/South America/

Where to find the information

This description may be obtained from the Human Resources Department or by analyzing the company's directory.

## Associating a user with an organization or a role

A user generally belongs to an organization and has a role (a function). This defines what we will call a profile. For instance, we may have:

- (Company/Sales/EU, Salesperson)
- (Company/Sales/Sales Manager)
- ...

Where to find  
the information

In general, you can obtain this information from your company directory or from the database of your Human Resources Department. You may also have to run a questionnaire through your company hierarchy in order to obtain this information.

## Step 3 - Simplifying the model

You now have for each user:

- The main login of that user
- His or her profile (organization, role)
- The actual list of his or her accesses over a given period
- The logins he or she used to connect to his or her target applications.

Step three will enable you to modify the model so that the security policy will reflect your organization's reality as much as possible. Certain relations are simplified and specified in this step.

Where to find  
the information

This step is basically an analysis step and has to be performed in collaboration with Information System managers and operations departments. You can also use statistical analysis tools to group data together.

## From a multi-profile user to a single-profile user

Sometimes, a user may work for different organizations, or have different functions (roles). If the differences between these profiles have an impact on application access, you have to differentiate this user with as much significant profiles. Thus, we could have two different profiles for the same user:

- Company/Sales/America/North America/Salesperson

and

- (Company/Sales/America, Marketing)

### Associating the right applications with the right profiles

If a user has several profiles and each profile gives him or her access to a set of different applications, it is important to differentiate these profiles. In this case, you have to assign to each different profile the list of associated applications, and create a family of virtual users corresponding to a user's different profiles.

### Selecting only significant organizations

An organization is generally a decisive factor between geographic, functional and personal constraints. Although information system organization is subject to the same constraints, it does not necessarily have the same boundaries.

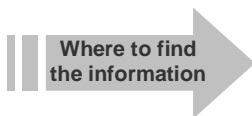
Thus, the geographic sub-divisions of a sales department are not important for the application-access policy. In this model, deleting these insignificant organizational levels will enable you to analyze the model more rapidly and more pertinently.

### Grouping applications by family

If you have applications used by exactly the same persons, you can create a group of applications. This group of applications will make it possible to process once all accesses to these applications within the same model, and to create a single set of rules applicable to all the applications.

### Role mining for complex organizations

The afore-mentioned procedures may be used during a case-by-case analysis for the simplest organizations.



On the other hand, if the organization becomes too complex, with, for instance, tens of thousands of users, it is possible at this stage to use role mining tools which will analyze actual accesses as well as the associated roles and organizations. These tools will group certain roles together in order to simplify the model.

## Step 4 - Determining the security policy

You now have for each user:

- His or her main login
- His or her profile (optimized organization, role) simplified in Step 3
- The actual list of his or her accesses over a given period
- The logins he or she used to connect to his or her target applications.

You can then either create an optimized security policy or perform a reconciliation operation with your existing policy.

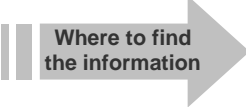
## Creating a security policy

If you do not have a very reliable centralized security policy, you can use Evidian's permission mining tool which automatically generates an optimized extended RBAC policy from the data collected so far.

The elements provided by this tool are:

- The list of rules used to associate a profile (organization, role) with an application or system
- The list of exceptions also used to associate a user with an application or system
- The exhaustive list of access rights, using the defined policy: for each rule, the selected users as well as the associated application logins.

Evidian provides this tool within the scope of its access-security-policy-definition support service.



Where to find the information

The tool is also accessible online for test, at [www.evidian.com/PolicyCreatorOne](http://www.evidian.com/PolicyCreatorOne). The proposed tests may be based on existing examples or a subset of your data.

The result may be integrated into a centralized role management tool.

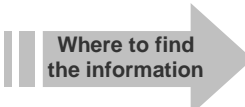
## Reconciling the data with an existing policy

It is possible to compare the collected data with an already existing policy. This so-called reconciliation operation consists in comparing line by line the rights associated with a user by your policy with the accesses actually made.

A first adaptation of your policy can be made by analyzing the differences.

These reconciliation operations may be performed in two different ways:

- **Manually**, by comparing the rights defined in the applications and systems with those stemming from the policy deduced from accesses
- **Automatically**, for complex organizations, using a role management tool which will unfold your policy for comparison with the collected access data.



Where to find the information

This reconciliation operation will be used to check the pertinence of the rules defined, and adjust them if necessary.

## Step 5 - Reconciling the policy with the policies defined in applications and systems

This last step consists in comparing the access-based policy with the policy defined already in the target applications and systems, in order to finalize the access policy.

### Creating a link between a right and a user

Analyzing the accounts declared in the target systems and applications helps determine the following relationship:

an account ↔ a declared access right

This relationship is completed by the relationship:

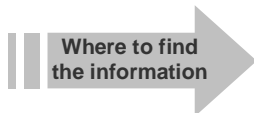
a user ↔ an application ↔ an account (user login)

obtained while collecting rights-related data. They will be used together to compare actual accesses with the rights declared in the systems, and to define an optimum access policy.

### Reconciling the policy with the target applications and systems

By comparing the differences, this second reconciliation step will make it possible to analyze the differences between the declared user rights and your policy.

For each difference, you will thus understand whether the right granted to a user is an error due to an exception that has not been deleted, a too loose-knit policy or a real right to be preserved.



Just like for the first reconciliation step, this operation can be performed manually or automatically using a role management tool. This tool will go and query the target applications and systems or the already existing temporary systems<sup>4</sup>.

---

<sup>4</sup> Provisioning tool: a tool which manages user accounts in target applications from a central console

## Extended RBAC / Bottom-Up approach at the service of your organization

---

Using the extended RBAC / Bottom-Up approach will enable you to define an access-security policy that is consistent with your organization's activity.

Your access-security policy will be comprehensible, maintainable and applicable.

The main advantages of this approach are:

- The user's organization is taken into account, in addition to his or her role, while granting his or her rights.
- The rights declared for each user are determined through the following relations:

user ↔ application ↔ account (login) ↔ declared right

- The policy model, accesses actually made and the rights declared in the target applications and systems are compared (reconciled in two phases).

These three elements will bring you close to your ideal access policy while optimizing the list of rules used.

For more detailed information, please go to [www.evidian.com/](http://www.evidian.com/)

Email: [info@evidian.com](mailto:info@evidian.com)