



Role Based Access Control

by Steve Slater

One of the most common buzzwords in IT security and compliance is RBAC, or Role-Based Access Control. The concept of RBAC is very simple, and has even been codified into an ANSI standard. But what do roles and role-based access controls really mean to the end user? This article presents some of the business benefits of managing access rights via roles, discusses whether or not roles are right for you, and provides guidance for deploying role-based access controls.

What are roles?

First, we need to define a role. As mentioned above, the concept of RBAC is simple; a role is nothing more than a collection of user access rights. This collection of privileges could be limited to a single business application or include privileges from several applications or systems. In addition to access rights, a role could include other roles, leading to role hierarchies.

How do we define a role and which privileges belong in which role? Who decides which users are in which roles? What happens when users need privileges that are not part of their

current role? These and similar concerns are what make RBAC deployments complicated and challenging for any organization.

Why Role-Based Access Controls?

The business benefits of RBAC are potentially very great. The most obvious is the tremendous time savings when combined with an automated provisioning tool. System and application accounts for new hires are created immediately with the correct set of access privileges needed for their job, based on predefined roles for the user's title or responsibilities.

Furthermore, compliance and security controls are significantly enhanced by using role-based access control. The roles themselves are already pre-defined. When managers and business owners perform periodic certifications of access rights, they simply need to review a handful of roles to ensure the user is in the correct role, rather than reviewing tens or hundreds of individual access rights. Moreover, automated tools can very easily detect user privileges that are outside of the approved role and provide a mechanism for handling exceptions.

Both of these core drivers equate to improved security governance and a reduced risk of audit findings and compliance deficiencies. In addition, the extra automation leads to significant savings of both time and cost.

Our ideal end state for our organization is therefore a well-oiled machine wherein every user access right or privilege is included in one or more roles. Role hierarchies are utilized to create “roles of roles” that enable cross-application enterprise roles. A limited number of top-level roles will exist and each user in the company is assigned to a handful of these top-level roles. The role assignments are linked to job descriptions and functional requirements, and no exceptions exist wherein users are granted individual access privileges.

Unfortunately, this RBAC nirvana is extremely unlikely to happen. The reality of nearly any workplace is that the work still needs to get done and various skill sets across the company are leveraged to perform tasks that may not fit cleanly into any particular job description. This inevitably leads to either many exceptions to existing roles or the creation of a unique role for each user.

We therefore want to be sure to set realistic expectations and a plan for efficient handling of RBAC exceptions. As we’ll see later in this paper, we also know that including every business application in our RBAC deployment is extremely unlikely. Given this proper perspective on RBAC, we can follow the steps described in this paper to successfully evaluate and implement a role-based access control program and achieve the security and cost savings we expect.

The RBAC deployment process

The process of an RBAC implementation can be simplified into four basic steps, each with its own unique set of challenges and desired outcomes. Failure to complete any of these phases can lead to serious complications in the deployment process. Our recommended steps include needs analysis, scope, planning, and implementation. We will walk through each of these processes to identify common pitfalls and provide recommendations to ensure success.

Needs analysis

RBAC is not for everyone. Even though this paper is focused on enabling a successful RBAC deployment, and my company provides products and services to enable RBAC, some of us simply should not attempt such a potentially massive undertaking. Or perhaps the scope needs to be more limited. Whatever the case, conducting an honest needs assessment will guide us down the best path.

We need to ask ourselves a few key questions. First, what are the business drivers? What pain points are we trying to address with the project? We presented two common drivers above, and you likely have your own unique environment that has other imperatives. These pain points become the core of our project’s success criteria and should always remain forefront during subsequent project phases. One of the most critical tasks of the RBAC project manager will be to continuously refer back to the primary objectives and keep the team focused so that months don’t get wasted learning a “cool feature” that doesn’t really address a core problem.

It is also crucial to quantify the inefficiencies or compliance failures. An RBAC deployment can be time-consuming and hard-dollar cost savings will help keep executive support alive throughout the project and provide a true return on investment calculation.

The next question we should ask is, will RBAC really help me? We may have several struggles related to access rights, but are roles the answer? Let’s look at a few considerations.

- **Common functionality versus custom responsibilities:** What percentage of users can be grouped into common responsibilities and therefore common roles? Typical cases where this happens are in helpdesk or customer service positions. Often several people perform identical functions. In the medical community, most doctors and nurses need the same access privileges as other doctors and nurses.

Conversely, information security staff and IT administrators are some of the hardest positions to fit cleanly into roles. In smaller companies, often nearly every employee wears many hats and has custom access needs. A quantitative measure of the percentage of users that will be covered by roles provides a key needs analysis metric.

- **Employee change and turnover rate:** How often do employees and contractors come and go, and how often do roles for existing staff change? If changes are infrequent, it is much harder to justify the expense of a full RBAC deployment. Perhaps a simpler auditing or compliance tool is more appropriate.

- **Compliance and security benefits:** How much time will be saved or audit risks be

minimized through role-based access controls? For example, nearly every company must perform some type of periodic review and certification of user access rights. In many situations, each employee being reviewed could have tens or hundreds of individual access rights. Reviewing each privilege for every user in a company will either take an extraordinary amount of time for the reviewer, or else (more likely) the reviewer will simply “rubber stamp” the certification; in which case no security value is gained. Conversely, if each user has only several roles and a handful of exceptions, the approver is much more likely to stop and consider the request before blindly approving it.

By reviewing all of these factors, an educated recommendation can be made whether or not to proceed with the project. It is possible that the needs might not be great enough to warrant continuation of the project and potentially up to millions of dollars can be saved by avoiding a project with limited chance of success. On the other hand, the analysis may show an overwhelming potential for cost reduction and compliance improvement. Either way, our needs analysis has led us to our success criteria as well as provided metrics for an ROI calculation.

Information security staff and IT administrators are some of the hardest positions to fit cleanly into roles.

Scope

Armed with accurate knowledge of the true business needs, we now need to determine the scope of our project. Resist the temptation to take on every application in the company. As with nearly every identity-related project, an RBAC deployment conforms to the law of diminishing returns.

Some applications will simply take too much effort and offer too little benefit to be justified. Though the exact number will vary by company, a typical ideal scope will be between 60-90% of the key business applications.

Compliance regulations are often key aspects to consider and will narrow the scope to primarily compliance-critical applications.

Other factors to consider are the internal relationships between various business units. A key part of the RBAC project will be engaging business owners to help define which privileges should be part of which roles. Some applications will be either out of scope or part of a follow-on deployment phase simply because of expected challenges.

If you will be deploying commercial tools or relying on consultants to assist your deployment, this is an ideal time to begin engaging third parties. Now that we have identified our core business drivers and set our scope appropriately, the internal and external costs estimates can be weighed against the expected benefits.

Planning

The old adage that says “if you fail to plan, you plan to fail” was written with an RBAC deployment in mind. Be sure to allow for time in your estimates. For a typical RBAC project, well over 50% of the time will be spent in the planning phase. The goal of the planning phase is two-fold. The first is to plan the technical side of the project. Which access rights belong in which roles and which tools best fit our needs? The second goal is to determine if we can actually succeed in an RBAC deployment. Do we have support from application owners and administrators across the enterprise? Do the benefits exceed the expected costs?

An RBAC project is typically driven from the IT and/or information security team. Always remember that we are not an island. Specialized knowledge of a diverse set of applications will be necessary to move the project forward. Application administrators are essential to understanding often cryptic access rights profiles. Line of business managers need to approve the proposed role definitions. The successful RBAC project will have a large

virtual team that includes representatives from each of the in-scope applications.

Now that we have everyone on-board, it is time to achieve the primary goal of the planning phase and determine which privileges belong to which role. This is a very daunting task, especially at the beginning. Various automation tools have been developed to assist in the process and we will discuss several types later in this paper.

For the RBAC core team, this is the most critical phase to keep moving and the easiest place in the process for the project to spiral into analysis paralysis. Never underestimate the value of strong executive support. When other priorities begin to compete with the RBAC project, we must prevent it from being delayed. Stress the ROI numbers we calculated above to gain the support and ensure a high priority. But perhaps the most important tip in the planning phase is to be extremely clear and direct with our extended virtual team. We are often “borrowing” people’s time to complete this project and we need to give clear guidance to avoid wasting their time.

Application administrators are essential to understanding often cryptic access rights profiles.

For business owners, planning involves working with the RBAC team to define the roles. Some of the details we need to collect include:

- Total number of privileges in the application.
- Pre-existing roles within the native access rights definition.
- Types or categories of people that use the application.
- Access rights that are relevant to all users.
- Access rights that are assigned to only a subset of users.
- Existing users that we can use as a prototype, a starting point for a role. This is often a nurse, doctor, helpdesk user, customer service representative, or similar position.

Using this data, we perform access rights data mining and combine that with the business and administrator knowledge to accurately define the necessary roles for each ap-

plication. Remember to allow for exceptions when necessary. Our initial needs analysis most likely did not include placing 100% of user access rights into roles. More often we are striving for cost reduction and compliance improvement. These can be achieved even with exceptions as long as our exception handling process is simple and efficient.

I often hear claims that some tool or widget can automate this entire planning and role-mining process. Don’t believe the myth. Some tools are very useful for sorting and collecting data, and can save a tremendous amount of time. But no tool can explain the business-related uses of the application itself that are so essential to the process of defining roles. You should always plan on significant involvement from key application owners.

After completing role analyses for each application, we then combine these results across

the entire company and again look for commonalities. This is an iterative process where we seek to identify a percentage of common attributes based on key values such as job title, cost center, or organizational charts.

We use intelligent trial and error to attempt to fit a high percentage of common privileges into distinct roles. When we think we are close, we then verify with the extended project team to confirm and modify based on the irreplaceable human knowledge. Our end result will be a set of enterprise roles that group together various application-specific roles.

Remember that we are not seeking to have zero exceptions. But if we begin to see nearly as many roles as we have users in the company, we should not ignore that warning. Maybe we need to re-evaluate our role definitions. Perhaps our organization is not currently managed in a way where RBAC helps. It is far better to recognize that now before devoting resources to implementing a solution that can't achieve the project goals.

Implementation

If the first three phases were completed successfully, the implementation phase will be the easiest part of the entire RBAC project. All of the hard work is already complete. Our success criteria are defined, our scope is set, and

the majority of our enterprise roles have already been created. For those deploying a third-party RBAC tool, simply complete the installation and configuration. For others developing in-house systems, the required use cases are now defined.

Next, our provisioning and de-provisioning functions need to be altered to include the role definitions. This is where we realize a majority of true cost savings. The provisioning tool or process needs to integrate with the RBAC definitions.

For most commercial tools, this is easily achievable. For internally-developed tools or processes, be sure to consider the modification costs as part of the project planning. If the provisioning system is not RBAC-aware, the automation benefits will be difficult to realize.

Finally, the compliance and security benefits are significant. The periodic certification of access rights that previously involved certifying hundreds of privileges for each user now consists of certifying a handful of roles and exceptions. In addition, separation of duties conflicts can be defined at a role level and reports can instantly detect users with exceptions to the roles. Reports help identify users with excessive access rights before a catastrophic event occurs like the recent trading incident at Societe Generale.

Reports help identify users with excessive access rights before a catastrophic event occurs like the recent trading incident at Societe Generale.

Hazards and tips

We have just completed a four-step process to deploy role based access control. In some cases, it might go very smoothly. More than likely, however, one or more of the following pitfalls may threaten the project.

Bogged down in the details

At many places in the process, some applications will be very difficult to understand. The scope of the initial RBAC deployment therefore needs to be carefully and realistically set. If certain applications present unusual chal-

lenges, consider delaying them for later. If all applications seem to be overly difficult, consider if RBAC is appropriate and/or engage outside consulting help to view the project from a fresh perspective.

Too many exceptions

During the planning phase, it is unavoidable that certain users will not fit neatly into roles and will require exceptions. At SCC we coined the phrase "The Law of Exceptions." The number of exceptions is inversely proportionate to the number of roles.

Or more clearly, the more roles you have, the fewer exceptions, and vice versa. If you have too many roles in the organization, the RBAC efforts become pointless and you approach one role per user. If you have too many exceptions, RBAC again does not meaningfully improve upon the current situation. We strive for a compromise between number of roles and number of exceptions.

If a high number of users are requiring exceptions, consider this a warning sign. Especially for small and mid-sized organizations, RBAC is often not appropriate due to the many hats individual people must wear.

This highlights the importance of detailed planning to identify potential issues before committing significant resources.

Change and apathy

Change in any part of life should never be underestimated. While change is often good, it is also often resisted. As we attempt to modify internal processes and gain more efficiencies, we will inevitably be disrupting the status quo. We may be embraced by some and ignored by others.

The scoping exercise should anticipate resistance by thinking outside the box. Are people concerned about their job being eliminated? Do they have too many other priorities to fully engage in our RBAC project? Honestly consider these factors and ensure a strong executive support to help overcome possible apathy.

While role-mining tools look great in a demo, the value is extremely difficult to predict.

Tools

Technologists love tools. They make our lives more efficient and automate very labor-intensive tasks. But how do we know which tools we really need? In this section, I will give a brief description of the general classes of tools related to role-based access control, and provide some considerations for your own evaluation.

Role-mining tools

The mission of a role-mining tool is to analyze existing identity and access data within the organization and suggest role definitions. Using a variety of formulas, the tools will consider existing privileges, organizational charts, job title, cost center code, geographic location, and various other parameters. Since the first analysis is almost never correct, we look at the results, tweak some parameters, and try again. This process iterates until we are either satisfied with the results or give up trying.

While role-mining tools look great in a demo, the value is extremely difficult to predict. Sometimes they will provide a great starting point for mapping privileges into roles, while in other companies, nearly all of the work of the

role-mining tool needs to be re-created. In a highly centralized environment with many common classes of users (like helpdesk, customer support), they can typically perform better. When performing a cost-benefit analysis for role mining tools, consider some of the following points:

- How well did the demo/evaluation perform with your data?
- Do internal application owners already know what their application roles should be?
- Do you have limited support outside of your project team where this type of automation tool could help fill in the gaps?
- Role mining is typically a one-time task and the tool is not heavily used after the project is complete.

Identity management tools

Many books can, and have, been written about the various aspects of identity management (IdM) solutions. The commercial solutions have various levels of support for role-based access control. Some have no concept of a role while others have full integration with a role-mining tool and provide interfaces to manage the ongoing maintenance of the roles.

The decision to use an IdM tool is largely independent of your RBAC project since multiple types of solutions exist to manage the role definitions besides IdM products.

The costs of an IdM deployment typically run in the millions of dollars because of the high consulting effort required, but the benefits of an IdM solution are likewise potentially high. A solid cost-benefit analysis should be completed outside of the RBAC efforts.

As they relate to RBAC, the most important considerations of an IdM tool include:

- How hard is it to define cross-application enterprise roles?
- Does the tool support the concept of a role hierarchy, where roles contain other roles?
- Is the automated provisioning fully “RBAC-aware”?
- How well can the IdM tool help with the ongoing tuning and compliance efforts related to RBAC?

Audit and compliance tools

A new category of compliance tools has emerged in recent years that can also assist with managing your role-based access controls. Burton Group has named this category Identity Audit (IdA). IdA tools, such as SCC’s Access Auditor, were developed to automate security and compliance efforts including the periodic certification of user access rights, enforcing separation of duties (SOD), and alerting and reporting to access rights data across the organization.

Because these tools do not perform the provisioning like an IdM solution would, the cost and deployment efforts are a fraction of IdM solutions’ outlays.

These compliance tools usually have the support for managing roles as well. All of the certification, SOD, alerting, and reporting functions can be based on a combination of roles and distinct privileges, and exceptions are easily spotted. IdA solutions can operate independently of IdM products or interoperate to leverage combined strengths.

Do it yourself

The final option for managing a role-based access control deployment is to build your own system to keep track of the role definitions and perform the required security and compliance reporting.

While this is usually too large of an endeavor to be cost-justified, some companies have been extremely successful in building their own RBAC management system. The common success factors in these cases were a clear business case and limited scope.

Summary

I presented a four-step process for leading a role-based access control project, and gave consideration to various tools to help you succeed. We need to remember two key points.

First, the ideal end-state of RBAC nirvana is not going to happen, but we can still make significant improvements in security, compliance, and automation.

Second, nothing is more important than an honest and accurate needs analysis to keep the project focused and ensure that we achieve a rapid return on investment. These items will become our rudder as we keep our project on course.

Dr. Steve Slater, CISSP, is the Founder and CEO of Security Compliance Corporation (www.securitycompliancecorp.com), a leader in the identity management market focused on user access rights, role management, attestation, and separation of duties. Over the past 15 years, Steve has provided a range of expert consulting including web application vulnerability assessments, penetration testing, regulatory compliance (SOX/GLBA/HIPAA), and PCI assessments for some of the world’s top companies, such as Bank of America and Visa.

Dr. Slater has written and taught Information Security classes for leading training organizations on topics including auditing techniques, LAMP, web application security, and secure development. In addition to security, Steve also holds a PhD in Nuclear Engineering from UC Berkeley and has several publications relating to high-performance computing and advanced numerical analysis. His scientific expertise earned the recognition of both the National Science Foundation and the Department of Energy.