Realtime publishers

The Essentials Series: Tackling Active Directory's Four Biggest Challenges

Automatically Provisioning New Users

sponsored by

QUEST SOFTWARE®

by Greg Shields

Copyright Statement

Automatically Provisioning New Users

Automating the complex tasks of Active Directory (AD) user provisioning is a necessity of many IT organizations. When new employees get access rights to files, folders, printers, email, applications, and databases based on their role from day one, they're immediately able to start work and provide benefit to the company.

Yet getting to that account management nirvana involves more than simply writing a few scripts. It requires the creation of carefully managed automation that ensures the right people get the right access every time. It must include the built-in workflow to validate privileges before they're assigned while preventing the accidental distribution of inappropriate rights. It must also be developed with strong controls so that the whole of IT can successfully and securely leverage its services.

That alone is a pretty heavy set of requirements. Yet your needs don't stop with simply creating new AD users. Your IT infrastructure includes other services and applications where access requires more than a simple AD object. Users need Exchange mailboxes. They need access into SharePoint document libraries. Users require accounts and permissions on applications such as Siebel, Oracle, and any number of other business-specific applications. Creating a home-grown system for provisioning any or all of these accounts requires scripting knowledge in multiple languages. It also requires the development knowledge and experience to encode the necessary workflows that make such a system useable by a business organization.

Even more critical, custom solutions require an entire infrastructure of support that wraps around the code itself. If you build it, you're responsible for maintaining it. You're the one who gets the call when it breaks. And if you ever leave your organization, you take with you the sum total of support for its continued use. That situation in and of itself creates a major business risk, one that could be completely obviated through the use of vendor-created and vendor-supported solutions.

With this in mind, however, let's take a look at some of the ways that such solutions could be created today. Although these might look simple at the outset, you'll find they get disturbingly complex as you dig even a little deeper.

The Native Solution

For example, creating a simple AD user account with the PowerShell scripting language requires only a few lines of code. In the following code snippet, an account is created for the user Jane Doe in the domain contoso.com:

```
$domain = [ADSI] "LDAP://main:389/dc=contoso,dc=com"
$usersOU = [ADSI] "LDAP://CN=Users,DC=contoso,DC=com"
$newUser = $usersOU.Create("jdue","cn=Jane Doe")
$newUser.put("description", "Jane Doe's User Account")
$newUser.SetInfo()
```

With its new focus on PowerShell, Microsoft has dramatically improved the tools available for administering accounts and access via the command line. Yet learning PowerShell doesn't happen overnight, and its use requires a fairly deep understanding of Microsoft technologies in addition to mere script syntax. Continuing with the example, the following code snippet creates an Exchange mailbox for Ms. Doe on the server \\ex2007-srv1:

```
Enable-Mailbox -Identity 'contoso.com/Users/Jane Doe' -Alias 'jane.doe' -Database
CN=InformationStore, CN=Ex2007-Srv1, CN=Servers, CN=Exchange Administrative
Group (GZEIBOHF34SPDLT), CN=Administrative Groups, CN=446A1087-6242-4D45-
AD2C-3C6C7F76F5EC, CN=Microsoft Exchange, CN=Services, CN=Configuration,
DC=contoso, DC=com'
```

Knowing the PowerShell cmdlet for creating a new user's mailbox is one thing; successfully targeting the precise Exchange server mail store to house that mailbox is yet another. The syntax above, locating that correct mailbox isn't easily done with a single PowerShell command. In a dynamic environment in which mailboxes and servers are always on the move, encoding this script in such a way that it can be used by multiple people and multiple stores requires quite a bit more verification and error handling.

Making this problem even more challenging is that Microsoft's focus on PowerShell isn't fully realized within every product. Other products such as SharePoint do not have the ready-to-use PowerShell capabilities that make unified automated user administration possible. SharePoint uses its own shell command STSADM to create new accounts. Thus, any automated user provisioning script that adds SharePoint access requires extra work to add Jane to SharePoint. Such a command might resemble:

```
STSADM –o adduser –url http://spserver1/sites/site1 -userlogon contoso\jdoe -
useremail jdoe@contoso.com -group Finance -username "Jane Doe"
```

You aren't done yet. Virtually every business organization uses applications that aren't developed by Microsoft. Thus, provisioning new accounts within other applications such as Siebel or Oracle requires understanding even more languages if you're to build a comprehensive solution.

The Need for Better Tools

Creating an automated provisioning solution is a massive development project that, once complete, brings great benefit to the business. The problem is that most organizations are not in the business of creating user provisioning infrastructures. In any build-versus-buy decision, organizations that don't have developer resources on-hand to complete such a project usually need to look elsewhere for solutions.

Organizations realize that the dynamic nature of the IT environment requires user provisioning and management solutions that can adapt as they solve the problem. If you are looking towards products that assist with the user provisioning and AD management challenges in your organization, consider those with support across domains.

Creating a unified workflow that enables IT to manage accounts requires integration with AD. It also requires integrations for each of the other areas where accounts are necessary: Exchange mailboxes, SharePoint document libraries, enterprise applications such as Oracle and Siebel, and others. If your organization leverages the use of Linux, UNIX, and mainframe servers, account management integrations into these areas are similarly critical for a complete solution.

Creating the workspace for an IT user requires more than just accounts. A best-in-class solution will enable the automated creation of home folders, instant messaging, mobile devices and their configurations, security and distribution groups, and access to whatever resources are necessary for employees to do their jobs.

Such a solution must be able to enable these capabilities in a secure manner to all needed account operators. It must include the built-in controls that prevent the inadvertent or malicious spread of access rights to the wrong individuals. It must include activity tracking capabilities that enable security officers and auditors to understand which individual completed which action. And, above all, it must include the necessary workflows, approval mechanisms, and notifications that ensure the right people get timely access to the right resources.