10 Steps to Cleaning Up Active Directory User Accounts—and Keeping Them That Way

Written by Randy Franklin Smith, CEO of Monterey Technology Group, Inc.
and CTO of LOGbinder Software

Introduction

AD security is crucial to controlling risk and ensuring compliance
Active Directory (AD) is the foundation of identity and access management (IAM) at most organizations and, as such, is probably the most crucial technology on the network. More and more systems and applications depend on AD for authentication, policy, entitlements, and configuration management. If AD is secure, everything is secure.

User accounts are important for security but are difficult to maintain
AD security, then, is crucial to controlling risk and ensuring compliance. But maintaining AD in a clean, organized, and secure state is a challenge for organizations. This is especially true with user accounts, which are both important for security and difficult to maintain.

User accounts are important to security because they are the basis for authentication and initial access to the network, systems, and applications. They are difficult to maintain because they need to mirror the status and role of the human member of the organization that they represent during the lifecycle of the member and her user account. When an employee is hired, a user account is created. As the user's job and assignments change, the AD account's identity information (such as job title, department, and phone number) is updated. The user is made a member of different groups and hopefully removed from groups as appropriate. Finally, when the user leaves the organization, the account is (hopefully) deleted.

This process sounds simple and straightforward. But the reality is that I have audited many AD implementations over the years and have yet to find one without a significant number of user accounts that were outdated, inappropriate, or out of compliance with good security practice and organization security policy. Problematic user accounts create an environment that is more difficult and time-consuming to manage. More importantly, they expose the organization to security risks and compliance problems.

The root cause for these problems is the fact that AD is not tied directly into the user account lifecycle events described previously. Therefore, organizations rely on end users, managers, and HR staff to recognize these events and to inform or initiate requests to IT so that user accounts can be kept up to date.

About this document
In this document, I provide 10 steps that you can take to remediate user account problems in AD and to prevent them from occurring in the future. These steps use native AD features and common workflow technology such as Microsoft SharePoint, so no significant prerequisites will hinder your ability to implement my recommendations.

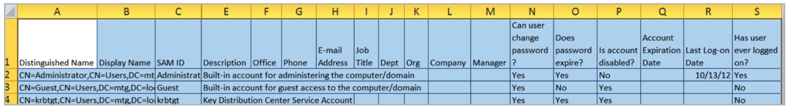
However, without additional tools, much of the clerical and manual confirmation burden on IT staff remains. And for the most part, the organization continues to rely on end users, managers, and HR staff to recognize important user lifecycle events and to inform IT about them. ActiveRoles Server (ARS) from Dell addresses these issues by automating the majority of AD maintenance. ARS provides a wealth of features for eliminating this reliance on end users, managers, and HR staff. To help you realize the full extent of how ARS can help your organization to reduce costs while improving security and compliance, this paper shows how ARS ties into each of the 10 steps. You will see how ARS can help you accomplish the goal of each step — or in some cases, obviates the need for it.

Step 1. Perform regular account analysis

The single, easiest step to maintaining a clean and secure AD is to regularly review user accounts. If you take the time to extract and review a list of your user accounts and their main properties before an audit, you can quickly find and remediate many points with which auditors take issue.

Getting a list of user accounts is easy

At one time, getting a list of user accounts was no easy task. But now, it's a simple matter of running a Windows PowerShell script and importing the results to Microsoft Excel. You can download my Output-ADUsersAsCSV script from http://www.ultimatewindowssecurity.com/tools/Output-ADUsersAsCSV and use it to produce a spreadsheet such as the one that follows. I use this script when I perform IT audits of AD.



	A	B	C	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
1	Distinguished Name	Display Name	SAM ID	Description	Office	Phone	E-mail Address	Job Title	Dept	Org	Company	Manager	Can user change password ?	Does password expire?	Is account disabled?	Account Expiration Date	Last Log-on Date	Has user ever logged on?
2	CN=Administrator,CN=Users,DC=mt		Administrat	Built-in account for administering the computer/domain									Yes	Yes	No		10/13/12	Yes
3	CN=Guest,CN=Users,DC=mtg,DC=lo		Guest	Built-in account for guest access to the computer/domain									Yes	No	Yes			No
4	CN=krbtgt,CN=Users,DC=mtg,DC=lo		krbtgt	Key Distribution Center Service Account									Yes	Yes	Yes			No

Filter the spreadsheet to find non-compliant accounts

This spreadsheet allows you to quickly filter on various user properties to find non-compliant accounts. Begin by identifying accounts with easy-to-find problems, such as a password that never expires. Then include filtering criteria on other columns, such as SAM ID or description, to eliminate service, application, and other accounts that you know to be exceptions.

These are easy problems to fix before the auditor comes and will reduce the number of risk findings on your audit. An obvious problem to look for is dormant accounts; I've designated an entire step to these accounts later in this paper.

But you can usually find other problems, such as accounts that should never have been created in the first place or that were not provisioned according to naming standards or other account creation controls.

Without knowing your naming conventions, account creation controls, and other standards, I can't provide specific guidance, but here's a simple example: I know that Acme Corp's naming standard mandates all end-user accounts begin with "u-", admin accounts with "p-" (for privilege), and service accounts with "s-". So I filter out all accounts beginning with those prefixes to find the remaining questionable accounts.

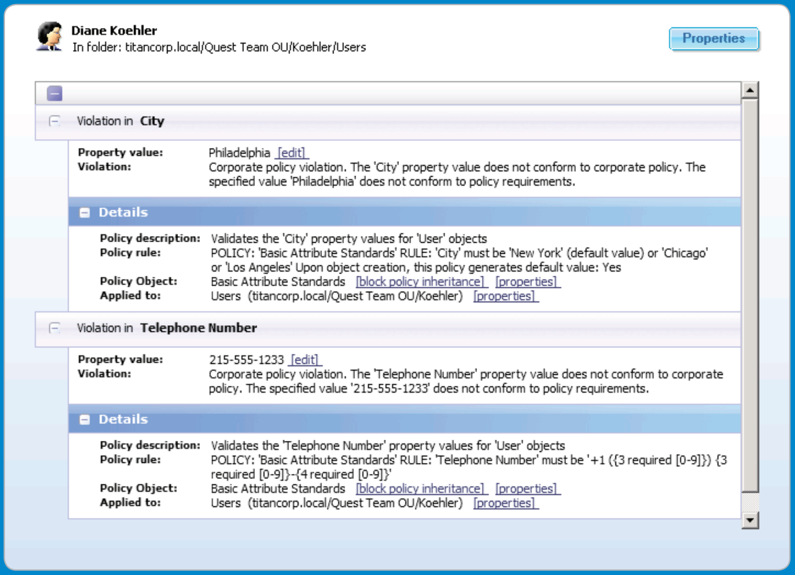
Some of those remaining accounts might be legitimate exceptions, which I address in a later step. But many of these accounts will turn out to be mystery accounts that you need to track down to determine purpose and status.

You certainly want to perform this step prior to an audit. But you really should do this every month to stay on top of your AD. After all, you probably aren't employed to just pass audits; the goal should be to keep AD secure and organized at all times.

Be aware that this step is a detective or reactive control, not a preventive or proactive control. Your goal should be to prevent the problems that I described from happening in the first place. Step 2 is the first way to accomplish that goal.

How ActiveRoles Server helps

ActiveRoles Server has the ability to compare your intended AD object standards (called policies) to your actual AD objects. The results of this comparison (called a Check Policy request) are delivered ad hoc, on screen with two clicks or via regularly scheduled reports.



Diane Koehler
In folder: titancorp.local/Quest Team OU/Koehler/Users

Violation in **City**

Property value: Philadelphia [edit]
Violation: Corporate policy violation. The 'City' property value does not conform to corporate policy. The specified value 'Philadelphia' does not conform to policy requirements.

Details

Policy description: Validates the 'City' property values for 'User' objects
Policy rule: POLICY: 'Basic Attribute Standards' RULE: 'City' must be 'New York' (default value) or 'Chicago' or 'Los Angeles' Upon object creation, this policy generates default value: Yes
Policy Object: Basic Attribute Standards [block policy inheritance] [properties]
Applied to: Users (titancorp.local/Quest Team OU/Koehler) [properties]

Violation in **Telephone Number**

Property value: 215-555-1233 [edit]
Violation: Corporate policy violation. The 'Telephone Number' property value does not conform to corporate policy. The specified value '215-555-1233' does not conform to policy requirements.

Details

Policy description: Validates the 'Telephone Number' property values for 'User' objects
Policy rule: POLICY: 'Basic Attribute Standards' RULE: 'Telephone Number' must be '+1 ({3 required [0-9]}) {3 required [0-9]}-{4 required [0-9]}'
Policy Object: Basic Attribute Standards [block policy inheritance] [properties]
Applied to: Users (titancorp.local/Quest Team OU/Koehler) [properties]

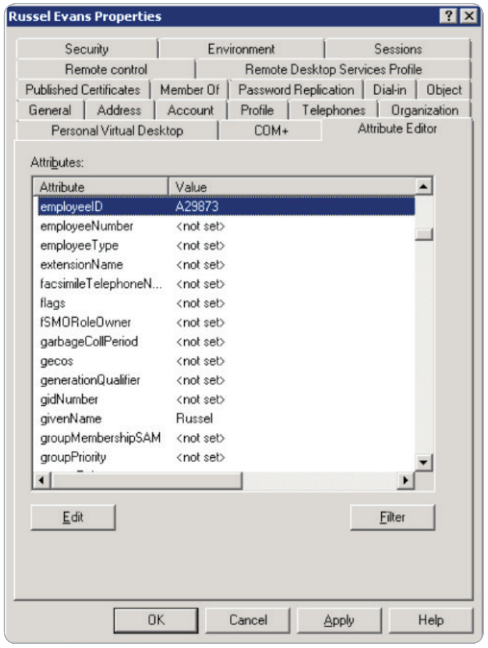
Step 2. Link accounts to employee records

The most fundamental way to keep AD accounts clean and secure is to link all accounts to an actual human. This includes non-human accounts such as those created for services and applications; I'll explain that later in Step 7. First let's talk about accounts that are created for individual persons including end users, contractors, administrators, and others.

First and foremost, any account that is assigned to an employee should be tagged in such a way as to positively link that account to the employee's master record in your HR system.

This link is crucial because employees' access to your network and entitlements within it must be tied to their status and role within the organization. The official record of this is the master record in HR, which also has the best chance of being up to date.

When an employee's status or role changes, you must be able to find the employee's accounts and change the status or entitlements accordingly. Documenting the employee ID on AD accounts is the key. (Of course, you also need to implement procedures to facilitate a response to these events; I'll cover this in a later step).



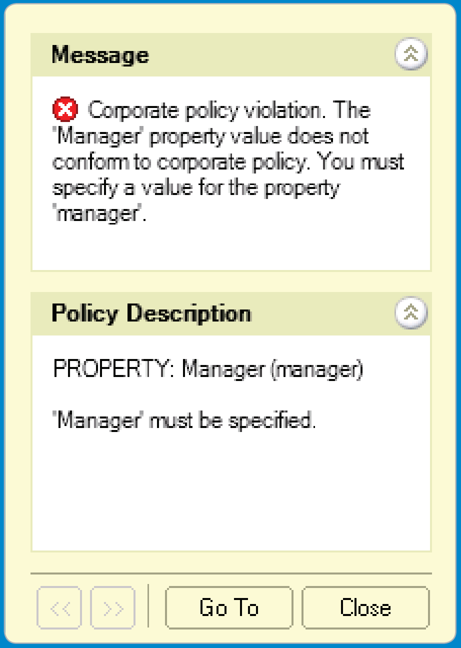
Attributes:

Attribute	Value
employeeID	A29873
employeeNumber	<not set>
employeeType	<not set>
extensionName	<not set>
facsimileTelephoneN...	<not set>
flags	<not set>
fSMORoleOwner	<not set>
garbageCollPeriod	<not set>
gecos	<not set>
generationQualifier	<not set>
gidNumber	<not set>
givenName	Russel
groupMembershipSAM	<not set>
groupPriority	<not set>

There are many ways to link AD accounts to employee records: (1) Using the Employee ID or Employee Number attribute in AD; (2) Via the Attribute Editor tab, as shown in the figure above; (3) Entering the employee ID in the Description or Notes field; (4) Embedding the employee number in the logon name.

How ActiveRoles Server helps

Through account creation policies, ActiveRoles Server can mandate that all non-human accounts must be created with a Manager or EmployeeID value (or any attribute, custom or otherwise, for that matter).



Message

Corporate policy violation. The 'Manager' property value does not conform to corporate policy. You must specify a value for the property 'manager'.

Policy Description

PROPERTY: Manager (manager)

'Manager' must be specified.

Step 3. Monitor new accounts

In IT audits of AD, I frequently find a surprising number accounts that should never have been created or that were created without following the organization's standards for naming convention or other policies. One reason that this happens is because too many people in the IT department have authority to create accounts; this issue is addressed in a later step.

Intruders often create backdoor accounts

Successful intruders, both human and automated, often create backdoor accounts to ensure continued access and to obfuscate their activity. Flame, a recent weaponized malware, specifically attempted to create such an account whenever it discovered that it was running under the authority of a domain admin.

Stop them when the account is created

So tracking down new accounts is crucial — but also time-consuming and often inconclusive. The best time to track down provenance of a new but non-compliant account is when it's first created:

- You can identity who created the account.
- The account creator is still at your company.
- The creator remembers why the account was created.

How to monitor and review new accounts

There are two ways to review and respond to new accounts:

- Monitor AD domain controller security logs for event ID 4720 (you need to enable the User Account Management audit subcategory).
- Run the Output-ADUsersAsCSV script and sort on the WhenCreated column.

As you review each account, do your best to answer the following questions:

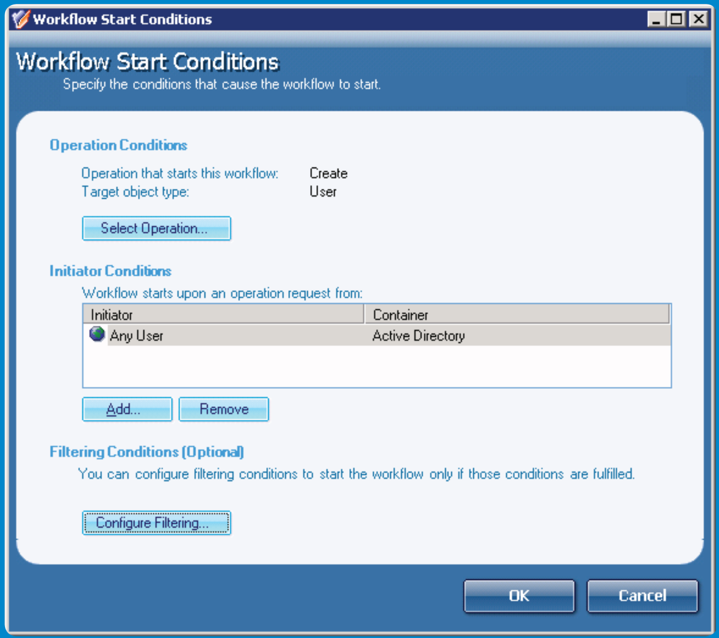
- Is there a work ticket or other corroborating documentation for this account?
- Does the account match established naming conventions?
- Does the account comply with your organization's other account-creation standards and policies?

If the account turns out to be unauthorized or non-compliant, you will need to follow up with whoever created it. The advantage with using the first method is that the security log event 4720 tells you who created the account.

```
Event ID 4720 - A user account was created
Subject:
        Security ID: ACME-FR\administrator
        Account Name: administrator
        Account Domain: ACME-FR
        Logon ID: 0x20f9d
New Account:
        Security ID: ACME-FR\John.Locke
        Account Name: John.Locke
        Account Domain: ACME-FR
Attributes:
        SAM Account Name: John.Locke
        Display Name: John Locke
        User Principal Name: John.Locke@acme-fr.local
```

How ActiveRoles Server helps

ActiveRoles Server acts as a virtual firewall around AD and enables you to create workflow for any operation, such as the creation of any account in the domain.



Workflow Start Conditions

Workflow Start Conditions

Specify the conditions that cause the workflow to start.

Operation Conditions

Operation that starts this workflow: Create

Target object type: User

Initiator Conditions

Workflow starts upon an operation request from:

Initiator	Container
Any User	Active Directory

Filtering Conditions (Optional)

You can configure filtering conditions to start the workflow only if those conditions are fulfilled.

Step 4. Automate account maintenance

Steps in creating a new account

To help ensure that new accounts are created according to your standards, automate as much as possible of the account creation process so that the potential for human error is eliminated. Creating a new account includes the following steps:

1. Create the account in AD.
2. Set identity attributes (job title, phone numbers, and so on).
3. Create the account's mailbox in Microsoft Exchange.
4. Make the account a member of groups that are appropriate to the user's role.
5. Register the AD account in other applications, as necessary.

Automate the process with PowerShell scripts

Many of these steps can be automated through PowerShell scripts. The following script performs steps 1 through 4.

```
New-ADUser –Name "randysmith" –SamAccountName randysmith  –
AccountExpirationDate 01/01/2014 -GivenName "Randy" -Surname "Smith"
-DisplayName "RandySmith" -Path 'CN=Users,DC=acme,DC=local' –
EmployeeID "93299" –OfficePhone "27884" –Title "CEO"

Enable-Mailbox -Identity acme\randysmith -Database Database01

Add-ADGroupMember Group1 acme\randysmith
Add-ADGroupMember Group2 acme\randysmith
```

You can make a customized version of this script for roles in your organization that have high turnover. Or you can enhance this script to accept input and build the account according to choices made at execution time.

How ActiveRoles Server helps

ActiveRoles Server provides numerous interfaces, including PowerShell, ADSI scripting, SPML, MMC and Web. The significance is that you can enforce standards (called policies) on any AD object CRUD operations regardless of the interface. This layer of management enables you to ensure that all activity to your AD is controlled by your standards. Failure to adhere to your standards can result in an allowed violation (reportable) or cause an error response; the choice is yours.

Step 5. Handle departed users and role changes

A frequent source of risk in AD is user accounts that have not been disabled even though the person is no long associated with the organization (e.g., an employee who has been terminated). It is crucial for HR or managers to inform IT when employees are terminated or when other relationships (such as a contractor relationship) end.

IT staff responsible for account management also need to know when users change jobs or other roles so that the users' group memberships and other entitlements can be revised.

Looking for dormant accounts does not address this problem

As simple as this might sound, organizations commonly fail to implement a working process to disable user accounts or to change entitlements when a user's status changes. During IT audit interviews, when asked what the procedure is for disabling departed users, I have observed staff answering that they regularly check for dormant user accounts and disable accounts that have not logged on recently. This is not an effective control for the risk at issue. After all, if someone is still accessing the network after being terminated, their account will never show up as being dormant and hence will never be disabled.

Effective ways to handle departing users and role changes

The following are three ways that some organizations use to fulfill this vital requirement, beginning with the most preferable:

- Most organizations have a clearly defined and strictly executed process for removing a user's physical access to the building; make account disabling part of this process.
- If your HR application includes workflow, configure it to automatically send an email to account administrators when a user is terminated or when a user's job title or manager changes.
- Most HR applications allow you to schedule automatic report delivery; schedule a daily report of terminations and job changes to be delivered to account admins.

The bottom line is that to comply with any regulation framework, an organization must disable accounts and adjust entitlements whenever a user's status changes. Whichever process is selected, management should understand its importance and responsibility should be clearly defined.

How ActiveRoles Server helps

ActiveRoles Server includes “account termination policies,” which allow you to designate what happens to a user object when its corresponding “carbon equivalent” has been terminated. Options include disabling, moving OU location, password and logon name scrabbling, renaming with operation variables, and more. You can also remove the user from all groups, re-permission the user’s home directory and mailbox, and much more. It is important to note that these policies can be kicked off manually or programmatically.

Step 6. Handle dormant accounts

Given the difficulty that most organizations face with reliably disabling accounts of departed users, the next step in keeping AD clean and secure is to regularly check for dormant accounts (i.e., user accounts that have not recently logged on). It bears repeating, though, that this step does not qualify as a substitute for Step 5.

Finding dormant accounts used to be hard but now it's easy
In Windows 2000, it was difficult to find a dormant user because the account's last logon date and time was not replicated between domain controllers. Therefore it was necessary, for each account, to query each domain controller for the last logon date and time on record for the account. The most recent date and time was then used to determine when the user had last authenticated.

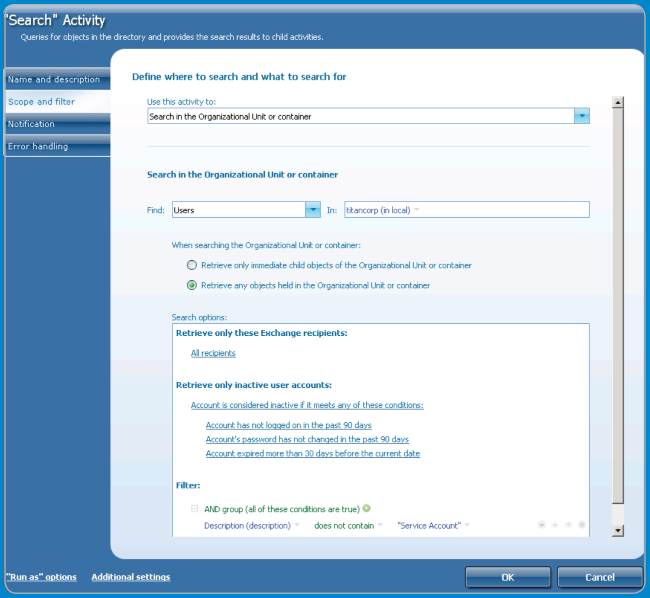
Thankfully, beginning with Windows 2003, Microsoft added a new lastLogonTimestamp attribute to AD user accounts; this attribute is replicated every 7 days. This replication ensures that you can query any domain controller and get last logon times that are recent enough to identity dormant users.

LastLogonTimestamp is exposed by Get-ADUser with the LastLogonDate property, as shown in the Output-ADUsersAsCSV script in Step 1. With that script, you simply need to sort on the Last Logon column in descending order to easily identity accounts that have not recently logged on.

You should also check for user accounts that have never logged on. In spreadsheets produced from Output-ADUsersAsCSV, these accounts are indicated by rows in which the Last Logon column is blank.

How ActiveRoles Server helps

ActiveRoles Server automates dormant account classification, discovery, and remediation (such as deprovisioning) and offers configurable notification.



'Search" Activity

Queries for objects in the directory and provides the search results to child activities.

Define where to search and what to search for

Use this activity to: Search in the Organizational Unit or container

Search in the Organizational Unit or container

Find: Users In: titancorp (in local)

When searching the Organizational Unit or container:

○ Retrieve only immediate child objects of the Organizational Unit or container

● Retrieve any objects held in the Organizational Unit or container

Search options:

Retrieve only these Exchange recipients:

All recipients

Retrieve only inactive user accounts:

Account is considered inactive if it meets any of these conditions:

Account has not logged on in the past 90 days
Account's password has not changed in the past 90 days
Account expired more than 30 days before the current date

Filter:

AND group (all of these conditions are true)

Description (description) does not contain "Service Account"

Step 7. Manage non-human accounts

Not all accounts directly correspond to a person. For instance, many applications require one or more accounts for services to log on. These accounts often have privileged access to servers and data and therefore need to be secured.

Why highly privileged accounts are at risk

However, application and other non-human accounts are difficult to track. In IT audits, my consulting practice frequently finds highly privileged accounts that are at risk for the following reasons:

- No one is sure about the account's purpose or why it should continue to exist.
- The account password has not been changed, despite the departure of many administrators, for fear of breaking an application somewhere on the network.
- The account has authority to log on interactively. Non-human accounts should be prohibited from logging on interactively at the console or via Remote Desktop to prevent administrators (who know the account's password) from logging on as that account to perform actions that can't be tied to their identity.

Identify non-human accounts

The first step in managing non-human accounts is to identity all such accounts. You can do so by using a prefix in the naming convention of the logon name, putting the accounts in a specific Non-Human Accounts organizational unit (OU), or tagging them as such via some other attribute in AD.

Document the purpose and owner of each account

Next, the purpose of the account and the systems on which it is used should be documented in the Description or Notes fields of the account.

An owner needs to be designated for each non-human account and the information needs to be documented in AD. The owner can be an individual human user account, but it is usually better to select a group that corresponds to the team that is responsible for the application or other technology that uses the account. The owner can also be documented in the Description or Notes field.

Password maintenance

One of the most difficult risks associated with non-human accounts is password maintenance. The password of a non-human account needs to be changed whenever an administrator (who knows the password) leaves the organization. Unless accounts are documented correctly, determining to which non-human accounts a given administrator had access is difficult. But changing an account password entails risk: any services, scheduled tasks running as that account, or applications that store that account's password must be updated or they will break the next time they start or attempt to log on.

Determine which systems an account is being used on

If you are attempting to clean up an existing set of non-human accounts, you can determine which systems an account is being used on by consulting the Windows security log. Assuming that you have enabled the Kerberos Service Ticket Operations audit subcategory in your Default Domain Controller Policy Group Policy Object (GPO), your domain controllers will log event ID 4769 whenever a user account requests a Kerberos service ticket to any system in the domain. By searching domain controller security logs for all occurrences of 4769 where Account Name is the service account in question, you can obtain a list of all computers on which that account is being used; look at the Service Name field in those events. The Service Name field in event ID 4769 identifies the computer for which the user account is requesting authentication.

Step 7. con't

Limit the logon rights of non-human accounts
One final step for securing non-human accounts is to limit their logon rights on computers throughout the domain. This helps to prevent non-human accounts from being abused by someone logging on with the account interactively at a computer's console or via Remote Desktop. This step also serves as a defense-in-depth measure in case password changes are missed when an administrator leaves. Five logon types in Windows have both an allow and deny right:

To log on in a given way, you must have the corresponding allow logon right. Even then, if you have also been assigned the deny logon right, you will not be allowed to log on; the deny logon right overrides the allow right. You can find these rights in a GPO under Computer Settings\Windows Settings\Security Settings\Local Policies\User Right Assignments.

Usually, non-human accounts should have only the "Log on as a service" right. It would advisable to explicitly deny Interactive and Remote Desktop logon rights to prevent the account from being misused. If you add all non-human accounts to a specific group for that purpose, you can then assign that group the "Deny log on locally" and "Deny log on through Remote Desktop Services" rights in a GPO such as Default Domain Policy, which is applied to all domain computers.

Be careful about denying the Network logon right. The application using the account might need to access resources on other networks.

Logon type	Logon rights
Interactive	Allow log on locally Deny log on locally
Remote Desktop	Allow log on through Remote Desktop Services Deny log on through Remote Desktop Services
Service	Log on as a service Deny log on as a service
Scheduled Task	Log on as a service Deny log on as a service
Network (e.g., shared folder access)	Log on as a batch job Deny log on as a batch job
FIPS 140-2 RDP transport encryption	Access this computer from the network Deny log on through Remote Desktop Services

How ActiveRoles Server helps

ActiveRoles Server can mandate and validate (through comparison reporting) that all non-human accounts are configured with the naming convention, attribute settings, objection location, and group membership (tied to GPO) that meet your company's standards. Additionally, if-then workflows can be enabled to require (tiered) approval for all (service) accounts created in a certain OU location and/or for those accounts with a particular naming prefix, etc.

Step 8. Control exceptions

Document legitimate, approved exceptions

The old adage says that "rules are made to be broken." There are definitely legitimate exceptions to standards for user accounts. For instance, you might have an application that requires a user account with a specific name that violates your normal naming convention. For situations such as this one, you need a way to document legitimate, approved exceptions. The best way is with an OU named Exceptions or by flagging exception accounts as such in the Description or Notes fields.

But simply labeling an account as an exception is not enough; the account's purpose and owner should be documented, as described in Step 7.

Don't allow exceptions to become common

A word of caution: I have seen AD implementations in which a large percentage of accounts were exceptions. Staff had gotten into the habit of just flagging an account as an exception whenever it was inconvenient to follow account maintenance standards and procedures. The provision for exceptions should not be abused.

How ActiveRoles Server helps

ActiveRoles Server can easily accommodate and control exceptions through policies to enforce that exception accounts be created only in certain locations. When an exception is created in the exception location, ActiveRoles will ensure that all necessary configuration standards, attributes or otherwise are met.

In addition, you can create a workflow to require approval whenever a creation request is made (manually or programmatically) for a new exceptions, thus preventing exceptions from becoming the “rule.”

Step 9. Control admin authority

Limit the number of people who can create accounts

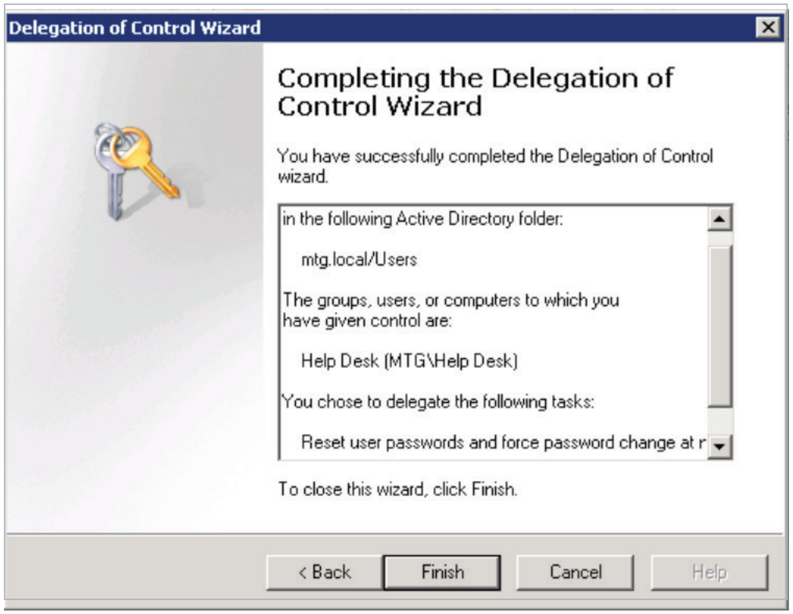
I noted earlier that one reason AD tends to get littered with unneeded or mystery accounts is because organizations frequently give too many people authority to create user accounts.

To enforce new account creation controls that are crucial for security and compliance, we must limit the number of people who can create accounts to just a few people who are trained and responsible for new-account policy compliance.

Use the Delegation of Control Wizard

AD supports least privilege by allowing domain admins to delegate selected permission over specific OUs. When properly implemented, AD's delegation of control ability allows people to do their jobs without giving them more authority than needed. For instance, rather than making the Help desk members of Domain Admins, you might grant the Help Desk group the Reset password permission on the OU that contains your end-user accounts.

To begin the Delegation of Control Wizard, simply right-click the desired OU and select "Delegate Control." The following figure shows password reset authority being delegated to the Help Desk group.



Delegation of Control Wizard

Completing the Delegation of Control Wizard

You have successfully completed the Delegation of Control wizard.

in the following Active Directory folder:

mtg.local/Users

The groups, users, or computers to which you have given control are:

Help Desk (MTG\Help Desk)

You chose to delegate the following tasks:

Reset user passwords and force password change at r

To close this wizard, click Finish.

How ActiveRoles Server helps

The “roles” in “ActiveRoles Server” are access templates that you can apply to any location in your AD infrastructure or even to virtual locations that you can create and dynamically maintain within the tool. Access templates are a collection of AD permissions, categorized by target object, that allow you to easily delegate a least privilege to your administrators. These permission sets can be as simple as “reset password” or as detailed as read/write/list permissions to any/all AD object’s attributes. ARS includes over 300 access templates out of the box.

Step 10. Leverage workflow technology

SharePoint is better than email alone for account management

I find many organizations trying to handle new account requests, job terminations, job changes, and various approvals using nothing else than email. This makes it difficult to follow account management standards or to prove compliance. Workflow technology, such as lists in SharePoint, will never be a full automation option for account management, but it is definitely an improvement over email alone. SharePoint, as an example of workflow technology, allows you to give Announcement lists an email address that turns incoming emails into new list items and carries any attached documents over to list item attachments. You can customize the list with Status fields to track the processing steps of the list item.

Example: Using SharePoint to manage termination-related account changes

For instance, you can use an email-enabled SharePoint list to organize job termination notifications and to document compliance with your departed user procedure. If you use option 2 or 3 in Step 5, configure the HR application to send its emails to your SharePoint list, and add Status and Notes columns to the list. As new job termination notifications or reports are delivered to the list, you can disable the associated accounts in AD and edit the list item to document that it was processed and which accounts were disabled in response. You can even subscribe to alerts on the list so that you know as soon as an item is created. Similar lists can be created for new account requests and job change notifications.

The point is that you need to leverage workflow technology to reduce the clerical and documentation burden on account administrators while improving your compliance.

How ActiveRoles Server helps

The benefit of ActiveRoles Server's architecture is that all CRUD operations that flow through it are audited and can be reported on. You will be able to provide reports for all new account creations or modifications, group modifications, account deprovisionings, etc. Reports include the five W's (Who, What, When, Where and Why) and can automatically be sent to auditors; a web portal is also provided.

Maintain a clean and secure AD – automatically

Native tools alone leave a lot of manual work – and the risk of human error

The 10 recommendations in this document will help you clean up the user accounts in your AD as well as prevent problems from being repeated in the future. However, if you simply follow the recommendations without investing in additional tools, much of the clerical and manual confirmation burden on IT staff will remain, along with you reliance on end users, managers, and HR staff for notification of and information about important user lifecycle events.

Within IT, most organizations spend far too much time creating and terminating user accounts in AD. Native tools are inefficient and time-consuming, and the manual processes that they require can introduce human error that compromises both the security and stability of the environment. In addition, many organizations have equally inefficient but completely separate processes for creating accounts in their non-Windows systems, adding to administrative overhead and introducing even more security risks.

ActiveRoles Server automates user account maintenance, reducing work and enhancing security

As you have seen in the "How ActiveRoles Server helps" section in each step, ARS automates the majority of AD maintenance and provides a wealth of features for eliminating reliance on end users, managers, and HR staff. ARS helps you to accomplish—or even prevent the need for—each of the steps in this paper.

When integrated with QuickConnect, ARS enables AD to synchronize with external databases and directories, including SharePoint Server, line-of-business applications, and many more, by using add-on connectors. Every system on almost any modern operating system can now enjoy two-way identity synchronization on premise or in the cloud. Best of all, by integrating with your HR application, identity account creation can be used to drive automated access management.

ARS automates AD-based account creation and administration. Users are assigned to job roles that map directly to their organizational responsibilities, ensuring that they always have the right permissions to the right resources—nothing more and nothing less. Users are happier because they can get to the resources they need to do their jobs; administrators are happier because everything is automated, minimizing the need for tedious, manual button-clicking.

More about ActiveRoles Server

ActiveRoles Server solves your security issues and meets those never-ending compliance requirements by securing and protecting Active Directory simply and efficiently. ActiveRoles Server delivers automated tools for user and group account management that overcome the native shortcomings of Active Directory, so you can do your job faster. ActiveRoles Server is designed with a modular architecture, so your organization can afford to meet your business requirements today and in the future.

ARS provides out-of-the-box user and group account management, strictly enforced role-based security, day-to-day identity administration, and built-in auditing and reporting for Windows-centric environments.

ARS includes these features:

- **Secure access** – ActiveRoles Server acts as a virtual firewall around Active Directory, enabling you to control access through delegation using a least privilege model. Based on defined administrative policies and associated permissions generates and strictly enforces access rules, eliminating the errors and inconsis10cies common with native approaches to AD management. Plus, robust and personalized approval procedures establish an IT process and oversight consistent with business requirements, with responsibility chains that complement the automated management of directory data.
- **Automated account creation** – ARS automates a wide variety of tasks, including:
  - Creating user and group accounts in AD
  - Creating mailboxes in Exchange
  - Populating groups
  - Assigning resource in Windows

ARS also automates the process of reassigning and removing user access rights in AD and AD-joined systems (including user and group terminations) to ensure an efficient and secure administrative process over the user and group lifetimes. When a user's access needs to be changed or removed, updates are made automatically in AD, Exchange, SharePoint, OCS, Lync and Windows, as well as any AD-joined systems such as Unix, Linux, and Mac OS X.

- Day-to-day directory management – ARS enables you to easily manage all of the following:
  - Exchange recipients, including mailbox/OCS assignment, creation, movement, deletion, permissions and distribution list management
  - Groups
  - Computers, including shares, printers, local users and groups
  - Active Directory, including AD LDS

  ARS also includes intuitive interfaces for improving day-to-day administration and help desk operations via both an MMC snap-in and a Web interface.
- Manage groups and users in a hosted environment – ActiveRoles Server works with Quest One Quick Connect in a hosted environment where accounts from client AD domain are synchronized with a host AD domain. ARS enables user and group account management from the client domain to the hosted domain, while also synchronizing attributes and passwords.Utilize out-of-the-box connectors to synchronize your on-premises AD accounts to cloud-based services such as Salesforce.com, Google Apps, Microsoft Office 365, Lync Online, and SharePoint Online.
- Consolidate management points through integration – ActiveRoles Server complements your existing technology and identity and access management strategy. Its Extend All feature simplifies and consolidates management points by ensuri...ng easy integration with many Dell products, including Quest One Quick Connect, Quest One Identity Manager, Privilege Password Manager, Desktop Virtualization, Authentication Services, Defender, Password Manager, Webthority, and ChangeAuditor. ActiveRoles Server also automates and ex10ds the capabilities of PowerShell, ADSI, SPML and customizable Web interfaces.

For more information or to see ActiveRoles Server in action through a virtual trial, please visit http://www.quest.com/activeroles-server/questdrive.aspx.

About the author

Randy Franklin Smith is an internationally recognized expert on the security and control of Windows and Active Directory.

Randy regularly performs IT audits of Active Directory and each year teaches dozens of IT auditors how to do the same with his Audit and Assessment of Active Directory and related webinars.

Randy publishes UltimateWindowsSecurity.com and is CEO of Monterey Technology Group, Inc. and CTO of LOGbinder Software.

For More Information

The information in this document is provided in connection with Dell products. No license, express or implied, by estoppel or otherwise, to any intellectual property right is granted by this document or in connection with the sale of Dell products. EXCEPT AS SET FORTH IN DELL'S TERMS AND CONDITIONS AS SPECIFIED IN THE LICENSE AGREEMENT FOR THIS PRODUCT, DELL ASSUMES NO LIABILITY WHATSOEVER AND DISCLAIMS ANY EXPRESS, IMPLIED OR STATUTORY WARRANTY RELATING TO ITS PRODUCTS INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. IN NO EVENT SHALL DELL BE LIABLE FOR ANY DIRECT, INDIRECT, CONSEQUENTIAL, PUNITIVE, SPECIAL OR INCIDENTAL DAMAGES (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF PROFITS, BUSINESS INTERRUPTION OR LOSS OF INFORMATION) ARISING OUT OF THE USE OR INABILITY TO USE THIS DOCUMENT, EVEN IF DELL HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Dell makes no representations or warranties with respect to the accuracy or completeness of the contents of this document and reserves the right to make changes to specifications and product descriptions at any time without notice. Dell does not make any commitment to update the information contained in this document.

About Dell

Dell Inc. (NASDAQ: DELL) listens to customers and delivers worldwide innovative technology, business solutions and services they trust and value. For more information, visit www.dell.com.

If you have any questions regarding your potential use of this material, contact:

Dell Software
5 Polaris Way
Aliso Viejo, CA 92656
www.dell.com
Refer to our Web site for regional and international office information.