10 Steps to enhance the agility, security and performance of Active Directory

ONE IDENTITY by Quest

About this document

This document provides 10 steps to optimize the security, agility and performance of your AD. This optimization will help to remediate and prevent potential user account problems in AD, including breaches. These steps use system-provided AD features and common workflow technology, such as Microsoft SharePoint. This low-code/no-code approach to AD management requires a minimal learning curve, making it very simple to implement the recommendations in this document.

However, even if you heed all the recommendations in this doc, without additional tools to help manage and automate your processes, much of the clerical and manual confirmation burden remains on your line-of-business managers, HR reps and IT staff.

The good news is you can alleviate nearly all these challenges with One Identity Active Roles. Active Roles eliminates reliance on users, managers and other staff for manual processes.

Read on to see how Active Roles facilitates each of the 10 steps.

Overview

Microsoft Active Directory (AD) and Azure AD (AAD) bring organization and standards to how identity and account data is managed and stored. One Identity Active Roles unifies your AD and AAD environments and enables identity and account data to be managed with agility, security and speed. Together, Active Roles and AD/AAD provide IT leaders and admins a solution that dramatically enhances the security and efficiency of their AD environments, reducing vulnerability. To use an analogy, it's like taking a powerful production-version of a sports car and adding a racing suspension, a turbocharger and a vastly improved cloud-connected dashboard and performance-monitoring system, as well as protecting it all with a programmable, highly secure remote-access key fob.

While the showroom vehicle is great, the aftermarket-enhanced version enables you to handle whatever the road throws at you – including massive change and threats. It's faster, more secure, corners like a gazelle, requires less ongoing maintenance and is more fuel efficient. Your additional investment is paid back quickly and the vehicle prepares you to safely take journeys that were out of the question before. It's just plain better.

That's the same with Active Directory and One Identity Active Roles. They are better together.

If you're like 95% of Fortune 1000 companies, you already have the showroom car – you're using Microsoft Active Directory as your daily driver for provisioning/deprovisioning of user permissions. However, the world is moving faster, the resources that AD and Azure AD – as well as AD LDS – manage continue to diversify. Additionally, other trends increase the complexity surrounding AD/AAD, including identity security, the migration to the cloud and the critical role that AD/AAD plays in Privileged Access Management (PAM). To further complicate matters, you are managing each of your Azure tenants and AD domains separately. Your cars are each in their own garage, with a separate resource managing and maintaining them. This environment is resource intensive, to say the least.

In this document, we will show you 10 steps to take to clean up your Microsoft AD/AAD user account data. This process is key to efficiency and security. As we break down each of the 10 steps, we will take you through each one, demonstrate why it's important and how One Identity Active Roles enables or accelerates that step. Many of the steps are common sense – such as deleting unused accounts and revoking access to applications and other resources – but as we all know, in the heat of the daily battle, prioritizing manual account maintenance tasks over acute tech and data situations is difficult.

Active Roles provides a **wealth of features** to eliminate manually-intensive processes.

Active Directory is crucial to controlling risk and ensuring compliance

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 and Azure Active Directory (AAD) for authentication, policy, entitlements and configuration management. If AD is insecure, everything is insecure.

User accounts are important for security but are difficult to maintain

Securing Active Directory/Azure AD is crucial to controlling risk and achieving compliance. However, maintaining AD in a clean, organized and secure state is a challenge, particularly for user accounts.

User accounts are the basis for authentication and access to networks, systems and applications. They are difficult to maintain without the proper tools to support tracking users' permissions across multiple platforms. When an employee is hired, a user account is created. As the user's job and assignments change, the user's AD account (such as job title, department and phone number) is updated, including when the user leaves and joins groups. Ultimately, when the user leaves the organization altogether, the account's access rights should be properly deleted.

The root cause of these problems is weak user-account lifecycle practices. Many organizations operate with a significant number of user accounts across multiple domains.

Traditionally, organizations rely on end users, managers and HR staff to recognize events that impact a user's AD account.

These busy people are then expected to inform their overworked IT team to make the changes in AD to keep the user accounts up to date. When relying purely on manual processes, these changes are too often not executed, which leads to ghost accounts and inappropriate permissions that can be targeted by bad actors to wreak havoc on an organization.

10 steps to enhance the agility, security and performance of Active Directory

Step 1. Perform regular account analysis

The most effective way to maintain a clean and secure AD/ AAD is to regularly review user accounts. If an organization can review the account 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

In the past, getting a list of user accounts was no easy task. Now, it's a simple matter of running a Windows PowerShell script and importing the results to Microsoft Excel. See this script (Output-ADUsersAsCSV) available at http://www. ultimatewindowssecurity.com/tools/Output-ADUsersAsCSV.

It will output as a spreadsheet, such as the one that follows:

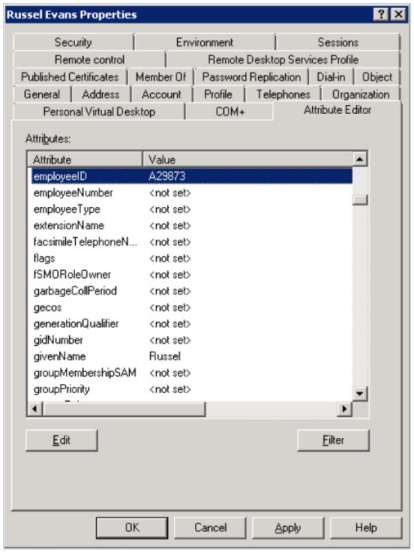
	A	B	C	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
1	Distinguished Name	Display Name	SAM ID	Description	Office	Phone	E-mail Address	Job Title	Dept	Org	Company	Manager	Can user change password ?	Does password expire?	Is account disabled?	Account Expiration Date	Last Log-on Date	Has user ever logged on?
2	CN=Administrator,CN=Users,DC=mt		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

With a script and the resulting spreadsheet, you can filter 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; an entire step focuses on this topic later in this paper. Other problems abound, 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.

For example: Acme Corp's naming standard mandates all end-user accounts begin with 'u-', admin accounts with 'p-' (for privilege), and service accounts with 's-'. First, filter out all accounts beginning with those prefixes to find the remaining questionable accounts. Some of those remaining accounts might be legitimate exceptions, which can be addressed in a later step. Many of these accounts will turn out to be mystery accounts that you need to track down to determine purpose and status.



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

You certainly want to perform this step prior to an audit. However, this should be done 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.

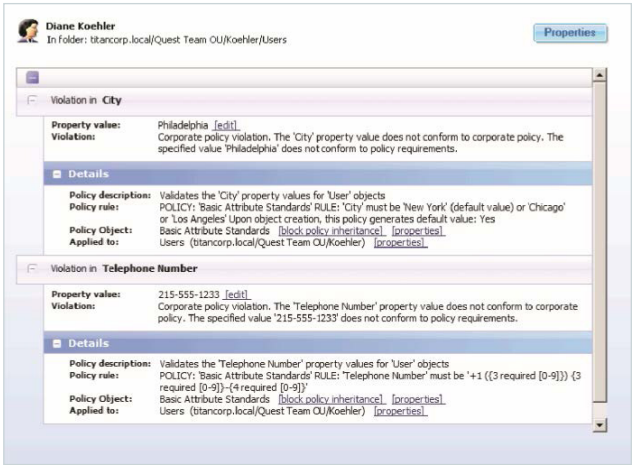
Be aware that this step is a detective or reactive control, not a preventive or proactive control. Your goal should be to prevent the problems from happening. Step 2 is the first way to accomplish that goal.

How Active Roles Helps

Active Roles 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. This functionality can help an organization get their house in order.

With a relatively small administrative investment in creation of policies, the process to regain control can begin.

Active Roles has the ability to compare your intended AD object standards (called policies) to your actual AD objects



ONE IDENTITY by Quest

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 user. This includes non-human accounts, such as those created for services and applications, which will be explained later in Step 7. First, focus on accounts that are created for people, including end users, contractors, administrators and others.

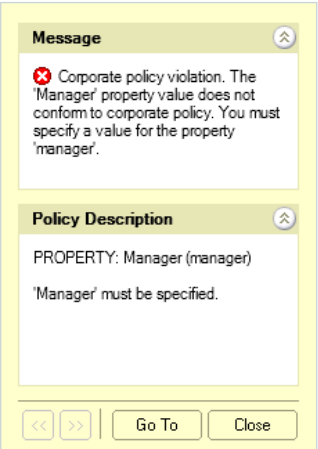
Most importantly, any employee account should be linked to the employee's master record in your HR system.

This link is crucial because employes' access to the network 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 emploee'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 (this is covered in a later step). 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 (this is covered in a later step).

How Active Roles helps

Using account creation policies, Active Roles can mandate that all non-human accounts are to be created with a Manager or EmployeeID value. In fact, Active Roles can manage account provisioning.



Successful intruders, both human and automated, often **create backdoor accounts** to ensure continued access and **to mask their activity**.

Step 3. Monitor new accounts

In IT audits of AD, it's common to find useless and nonstandard accounts, including those that stray from corporate naming conventions. This happens when too many people in the IT department have authority to create accounts. This will be 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 mask their activity. Trojans are an example of malware that often create a backdoor on a targeted system.

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 a noncompliant account is when it's created:

- Identify who created the account
- Are they still working at your company?
- Why was the account 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 When Created column

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

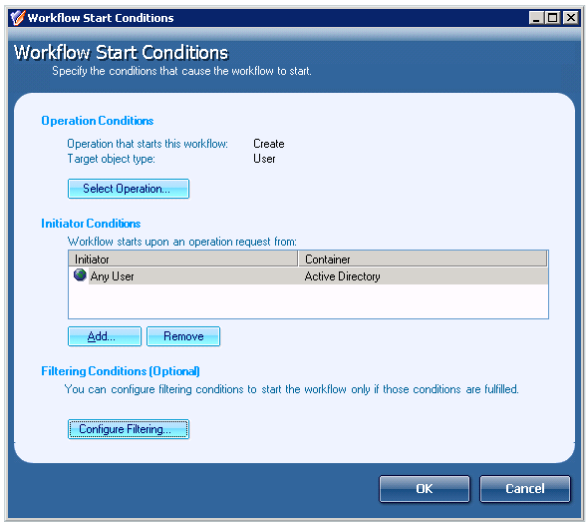
- Is there a work ticket or other corroborating documentation for this account?
- Does the account match established naming conventions?
- Does the account comply with your organization's other account creation standards and policies?

```
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
```

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

How Active Roles helps

Active Roles acts as a virtual firewall around Active Directory, ensuring that access based on the least privileged model is enforced. The ability to use workflows for any operation, such as the creation, modification or deletion of any account in the domain, means that all processes that typically are performed manually can be automated.



Step 4. Automate account maintenance

Steps to create 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 to reduce the potential for human error.

Account creation 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/O365

4. Add the account to groups that are appropriate to the user's role

5. Register the AD account in other applications as necessary

Automate with PowerShell Scripts

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

```
New-ADUser –Name 'randyjones'

–SamAccountName randyjones
– AccountExpirationDate

01/01/2014

–GivenName 'Randy' –Surname

'Jones'

–DisplayName 'RandyJones' –Path

'CN=Users,DC=acme,DC=local'
– EmployeeID '93299' –

OfficePhone

'27884' –Title 'CEO'

Enable-Mailbox –Identity acme\
randyjones –Database

Database01

Add-ADGroupMember Group1 acme\randyjones

Add-ADGroupMember Group2 acme\randyjones
```

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

How Active Roles helps
Active Roles provides numerous interfaces, including PowerShell, ADSI scripting, SPML, SCIM, 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 within your AD environment can be fully controlled by your standards. Failure to adhere to your standards can either result in an allowed violation (reportable) or cause an error response; the choice is yours.

Step 5. Handle departed users and role changes

For organizations that use traditional AD management tools, **ghost or orphaned user accounts are an ongoing source of risk**.

Without automation and/or a single source of truth of identities and permissions, you are likely to have people who are no longer employed or contracted with the organization in your identity data. It is crucial for whoever is responsible for updating status – whether it is an HR or IT – that they be notified when someone leaves the organization or moves a new role.

Looking for dormant accounts does not address this problem
As simple as this might sound, organizations often fail to disable user accounts or to change entitlements when a user's status changes. Here is a common response to audit questions about how an organization handles disabling departed users: ***Typically, organizations handle disabling departed users by looking for dormant accounts (accounts that have not logged on recently)***. This approach is flawed because if a terminated person is still accessing the network, their account will not show up as dormant and will not be included in the dormant account report.

Searching for dormant accounts is treating the symptoms rather than the cause. With an approach that considers the full AD account lifecycle, from hiring to departure and all steps in between, this problem can be eliminated. The same could be applied to redundant data. This is just as important as the accurate creation of new entries. Without purging redundant and unwanted data, your AD will fill with cluttered data.

Effective ways to handle departing with users and role changes
The following are three ways to effectively deal with status changes, in descending order of preference:

- Most organizations have a clearly defined and strictly executed process to remove a user's physical access to the building. Make disabling AD account part of this process
- If your HR application includes workflow, automate it to send an email to administrators when a user is terminated, moves to a new role or reports to a different manager
- Most HR applications allow you to schedule automatic report delivery. Schedule a daily report of terminations and job changes that is delivered to account admins

The bottom line is: account disablement and permissions status updates are required for compliance with industry and government regulations, and cyber insurance imperatives as well. Whatever your process, management should understand its importance, and responsibility should be clearly defined.

How Active Roles helps
Active Roles workflow capabilities includes tasks and entire processes triggered by changes to the directory. This includes account-termination policies that enables your organization to designate exactly what happens to a user account when a person has been terminated.

Options may include disabling the account, moving Organization Unit (OU) location, scrambling the password and altering the logon name, renaming with operation variables, assigning delegates for mail and home folders, etc.

Most importantly, Active Roles can remove the user from all security groups, re-permission the user's home directory, free up assigned O365 licenses, and much more. It is important to note that these policies can be triggered manually, programmatically or automatically.

For organizations that use traditional AD management tools, **ghost or orphaned user accounts are an ongoing source of risk**.

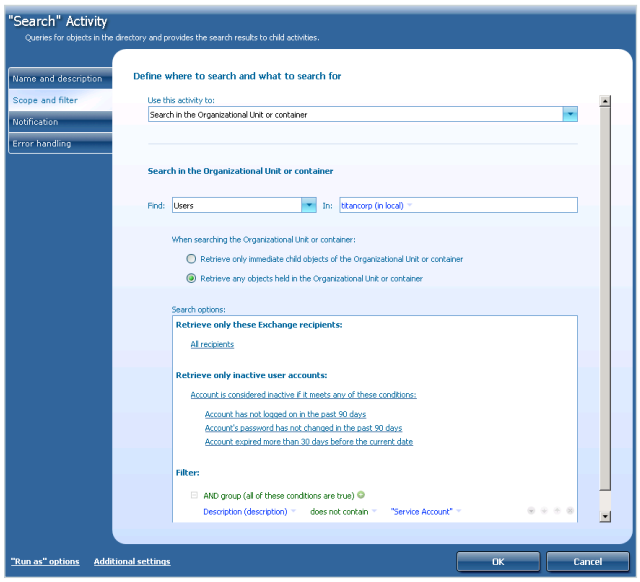
Step 6. Handle dormant accounts

The next step is to regularly check for dormant accounts (i.e., user accounts that have not recently logged on). Again, please note that this step does is not a substitute for Step 5.

Finding dormant accounts is easy

Before Windows 2003, it was challenging to find dormant accounts. Thanks to the lastLogonTimestamp attribute, it is relatively easy. This replication (every seven days) enables you to query domain controllers and see the last logon times, which helps to identify dormant users.

LastLogonTimestamp is exposed by Get-ADUser with the LastLogonDate property, as shown in the OutputADUsersAsCSV 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 Active Roles helps

Active Roles automates the processes that identify and manage dormant accounts, including classification, discovery and remediation. This facilitates the account clean-up process. When used in conjunction with appropriate policies around account lifecycle management (such as deprovisioning), not only can legacy issues be resolved but future problems can be avoided.

ONE IDENTITY by Quest

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

Application accounts and other non-human accounts are difficult to track. In IT audits, it's not uncommon to discover privileged accounts that are at risk for the following reasons:

- No one is sure about an account's purpose or why it exists
- Despite the departure of many administrators, an account's password has not been updated for fear of breaking an application somewhere on the network
- The account has authority to log on interactively
    - Nonhuman 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 anonymously as that account, and without individual accountability

Identify non-human accounts

The first step in managing non-human accounts is to identify 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 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.

Designate an owner for each non-human account and document it 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.

The use of Managed Service Accounts (MSA) was introduced in Windows Server 2008 R2 (and subsequently Group Managed Service Accounts (gMSAs), to automatically manage the passwords of service accounts. Using MSA/gMSA, you can considerably reduce the risk of system accounts being compromised.

Active Roles automates the processes that identify and manage dormant accounts, including classification, discovery and remediation.

Password maintenance

One of the biggest challenges of nonhuman 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 an administrator had access to is difficult. However, changing an account password entails to risk because any services or scheduled tasks that run 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 nonhuman accounts, you can determine the systems on which an account is being used 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. 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.

Limit the logon rights of non-human accounts

The 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 serves as a defense-in-depth measure in case password changes are missed when an administrator leaves. Five logon types in Windows have both and can allow and deny rights.

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 denylogon 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 is 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.

How Active Roles helps

Active Roles can mandate and validate (through comparison reporting) that all non-human accounts are configured with the naming convention, attribute settings, object location and group membership (tied to GPO) that meet your company's standards. Additionally, if-then workflows can be enabled to enforce (tiered) approval for all service accounts created in a certain OU location and/or for those accounts with a particular naming prefix, etc., with all such actions being fully audited and tied to the individual responsible.

The old adage says that '**rules are made to be broken**'. There are definitely legitimate exceptions to standards for user accounts.

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 in the Description or Notes fields.

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

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

Don't allow exceptions to become common
A word of caution regarding 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 Active Roles helps
Active Roles can accommodate and control exceptions through policies that ensure that exception accounts are only permitted in certain locations. When an exception is created in the exception location, Active Roles ensures that all necessary configuration standards, attributes or other policy constraints are met and enforced.

In addition, approval workflows can be employed where escalation occurs when a creation request is made (manually or programmatically) for a newly attempted exceptions, to prevent exception. This prevents exceptions from becoming the rule.

Step 9. Control admin authority

Limit who can create accounts
One reason AD is often littered with unneeded or mystery accounts is because too many people have authority to create user accounts. To enforce new account creation controls crucial for security and compliance, the number of people who can create accounts must be kept to just a few trained people.

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 necessary. 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.

Active Roles includes **more than 300** commonly used, tried and tested **access templates**, which makes Active Roles one of the fastest tools to get up and running and deliver ROI.

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

How Active Roles helps
The 'roles' in Active Roles are known as Access Templates. These represent collections of permissions to a very high degree of granularity, which can be applied to any location in your Active Directory infrastructure. It is even possible to apply them to virtual locations that can be customized and dynamically maintained within the tool.

Access templates are a collection of AD permissions, categorized by target object, that enable you to easily delegate admin permissions based on a least privilege model. 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. Active Roles includes more than 300 commonly used, tried and tested access templates, which makes Active Roles one of the fastest tools to get up and running – and deliver - ROI. And creating new templates is simple and quick.

Step 10. Leverage workflow technology

SharePoint is better than email alone for account management
Many organizations try to handle new account requests, job terminations, job changes and various approvals using nothing other than email. This approach 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
Email-enabled SharePoint lists can organize job termination notifiations and 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, as well as 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. This example demonstrates how to leverage workflow technology to reduce the paperwork burden on admins while enhancing your compliance.

How Active Roles helps
Active Roles' architecture enables reporting and auditing capabilities to be applied to all CRUD operations. This means that reports are available for all new account creations or modifications; all group creations, modifications and account deprovisioning and in fact, everything that happens through Active Roles is audited.

Reports include the five W's (Who, What, When, Where and Why) and can be sent to auditors automatically. Additionally, reports can be accessed online via a web portal.

A useful byproduct of the high level of audit is the ability to securely undo actions. An accidental deprovisioning action, for example, can be rolled back with a couple of clicks, and no loss of business continuity.

Maintain a clean and secure AD – automatically

Extending and automating the capabilities of system-provided tools to reduce risk
The 10 recommended steps in this document will help you clean up the user accounts in your AD, and prevent problems from being repeated. 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 a 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. System-provided tools are inefficient and time consuming. The manual processes they require introduce the possibility of human error that can compromise the security and stability of your Windows 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.

Active Roles automates user-account maintenance, reduces work and enhances security
As you have seen in the 'How Active Roles helps' section, in each step, Active Roles automates the majority of AD maintenance and provides a wealth of features to eliminate reliance on end users, managers and HR staff. Active Roles helps you to accomplish each of the steps in this paper.

Active Roles enables AD to synchronize with external databases and directories, including SharePoint Server, line-ofbusiness applications and many more. 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.

Active Roles automates AD-based account creation and administration. Users are assigned to job roles that map their responsibilities, ensuring that they have exactly the right permissions to the right resources—nothing more and nothing less. Users are happier because they can get to the resources they require in order to do their jobs and administrators are happier because everything is automated, minimizing the time spent executing tedious, manual tasks.

Active Roles 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.

Active Roles includes these features:

- **Secure access** – Active Roles 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, Active Roles strictly enforces access rules, eliminating the errors and inconsistencies common with native approaches to AD management. Plus, robust and personalized approval procedures establish IT processes and oversight consistent with business requirements, with responsibility chains that complement the automated management of directory data.
- **Automated account creation** - Automates a wide variety of tasks, including:
  - Creating user and group accounts in AD/AAD
  - Creating mailboxes in Exchange/ Exchange Online
  - Populating groups
  - Assigning resource in Windows

Active Roles also automates the process of reassigning and removing user access rights in AD/AAD 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 enables you to easily manage all of the following:

- Exchange/Exchange Online recipients, including mailbox assignment, creation, movement, deletion, permissions and distribution list management
- Groups
- Computers, including shares, printers, local users and groups
- Active Directory, including AD LDS

Day-to-day directory management also includes intuitive interfaces for improving day-to-day administration and Help Desk operations via an MMC snap-in and a web interface.

Manage groups and users in a hosted environment
Active Roles works in a hosted environment where accounts from client AD domains are synchronized with a host AD domains. This enables user and group account management from the client domain to the hosted domain, while also synchronizing attributes and passwords. Utilize out-ofthe- box connectors to synchronize your on-premises AD accounts to other platforms and applications. Leverage a fast growing range of over 30 connectors (https://www.cloud.oneidentity.com/products/connect/connectors) to multiple cloud-based services and applications, such as Salesforce, G-Suite and ServiceNow, through One Identity Starling Connect.

Consolidate management points through integration

Active Roles complements existing technology and IAM strategy. It extends all features and simplifies and consolidates management points by ensuring easy integration with many One Identity products, including Identity Manager, Privilege Password Manager, Authentication Services, Defender, Password Manager, and Quest Change Auditor. Active Roles also automates and extends the capabilities of PowerShell, ADSI, SPML and customizable Web interfaces.

10 steps to performance, agility and security

Step 1. Perform regular account analysis

Step 2. Link accounts to employee records

Step 3. Monitor new accounts

Step 4. Automate account maintenance

Step 5. Handle departed users and role changes

Step 6. Address dormant accounts

Step 7. Manage non-human accounts

Step 8. Restrict exceptions

Step 9. Control admin authority

Step 10. Leverage workflow technology

These 10 steps can clean up your AD/AAD data, which is critical to performance and security. One Identity Active Roles helps execute these steps and will keep your data clean moving forward. So, take that cool showroom car, power it with clean data and enjoy the performance, speed and handling that Active Roles can install into your AD/AAD strategy.

Microsoft Active Directory and One Identity Active Roles: Better Together

About One Identity

One Identity delivers unified identity security solutions that help customers strengthen their overall cybersecurity posture and protect the people, applications and data essential to business. Our Unified Identity Security Platform brings together best-in-class Identity Governance and Administration (IGA), Access Management (AM), Privileged Access Management (PAM) and Active Directory Management (AD Mgmt) capabilities to enable organizations to shift from a fragmented to a holistic approach to identity security. One Identity is trusted and proven on a global scale – managing more than 500 million identities for more than 11,000 organizations worldwide. For more information, visit www.oneidentity.com.

ONE IDENTITY by Quest