Microsoft Active Directory

- Disabling Active Directory Users with PowerShell (Generalized)
- ? Transferring FSMO Roles to Another Domain Controller
- ? Login Banner via GPO- How to Set a Security Message at Login via Group Policy (GPO)
- ? How to Configure Windows Updates to Run Outside Working Hours via Group Policy
- How to Convert Windows Server 2025 Evaluation to Full Version (Standard/Datacenter)
- Entra Cloud Sync / gMSA Troubleshooting Guide

Disabling Active Directory Users with PowerShell (Generalized)

A guide to efficiently disabling user accounts in specific Organizational Units (OUs) using PowerShell.

Introduction

This document provides a step-by-step guide on how to disable Active Directory (AD) user accounts located within specific Organizational Units (OUs) using PowerShell. This method is non-destructive, meaning the accounts are disabled and not deleted, allowing for easy re-enablement if needed.

We will focus on disabling all users within a 'YourTopLevelOU' and a 'YourNestedOU', assuming 'YourNestedOU' is nested under 'YourParentOU' and 'YourTopLevelOU' is a top-level OU directly under your domain (e.g., 'contoso.com').

{{info}} **Prerequisites:** Ensure you are running PowerShell with administrative privileges on a Domain Controller or a machine with the Remote Server Administration Tools (RSAT) for Active Directory installed. {{/info}}

1. Understanding Distinguished Names (DNs)

To target specific OUs, you need their precise Distinguished Name (DN). The DN is a unique identifier that specifies the exact location of an object within the Active Directory hierarchy.

- **Your Domain:** `yourdomain.com` (e.g., `contoso.com`) (translates to `DC=yourdomain,DC=com` or `DC=contoso,DC=com`)
- **YourTopLevelOU:** This OU is directly under your domain. DN: `OU=YourTopLevelOU,DC=yourdomain,DC=com`
- **YourParentOU:** This OU is also directly under your domain. DN: `OU=YourParentOU,DC=yourdomain,DC=com`
- **YourNestedOU:** This OU is nested inside the 'YourParentOU'. DN: `OU=YourNestedOU,OU=YourParentOU,DC=yourdomain,DC=com`

> **Tip: Verifying DNs:** To get the exact DN for any object in Active Directory Users and Computers (ADUC), enable "Advanced Features" under the "View" menu. Then, right-click the object, go to "Properties," click the "Attribute Editor" tab, and find the `distinguishedName` attribute. Copy its value directly.

2. The PowerShell Cmdlet: `Disable-ADAccount`

The primary PowerShell cmdlet used for this operation is `Disable-ADAccount`. We will combine this with `Get-ADUser` to retrieve the target users.

- `Get-ADUser -Filter *`: Retrieves all user objects.
- `-SearchBase "Your_OU_DN"`: Specifies the starting point for the search.
- `-SearchScope Subtree`: Crucially, this ensures that not only users directly in the specified OU are found, but also users in any sub-OUs or containers beneath it.
- `| Disable-ADAccount`: The pipeline operator sends the retrieved user objects to the `Disable-ADAccount` cmdlet, which performs the disabling action.

3. Disabling Users in 'YourTopLevelOU'

To disable all user accounts within 'YourTopLevelOU' Organizational Unit, including any users in its sub-OUs, use the following command. Remember to replace `YourTopLevelOU`, `yourdomain`, and `com` with your actual OU and domain names.

PowerShell Command:

```
Get-ADUser -Filter * -SearchBase "OU=YourTopLevelOU,DC=yourdomain,DC=com" -
SearchScope Subtree | Disable-ADAccount
```

> **Important:** This command will disable ALL user accounts found within the specified OU and any OUs nested inside it. Confirm your `SearchBase` is correct before execution.

4. Disabling Users in 'YourNestedOU'

To disable all user accounts within 'YourNestedOU' Organizational Unit, including any users in its sub-OUs (assuming 'YourNestedOU' is nested under 'YourParentOU'), use this command. Remember to replace `YourNestedOU`, `YourParentOU`, `yourdomain`, and `com` with your actual OU and domain names.

PowerShell Command:

```
Get-ADUser -Filter * -SearchBase
"OU=YourNestedOU,OU=YourParentOU,DC=yourdomain,DC=com" -SearchScope Subtree |
Disable-ADAccount
```

> **Important:** This command will disable ALL user accounts found within the specified OU and any OUs nested inside it. Double-check the `SearchBase` for accuracy.

5. Verification (Optional but Recommended)

Before running the `Disable-ADAccount` part, you can test the `Get-ADUser` portion to see which users will be affected. Remove the `| Disable-ADAccount` part to just list the users:

PowerShell Command:

```
Get-ADUser -Filter * -SearchBase "OU=YourTopLevelOU,DC=yourdomain,DC=com" -
SearchScope Subtree | Select-Object Name, DistinguishedName, Enabled
Get-ADUser -Filter * -SearchBase
"OU=YourNestedOU,OU=YourParentOU,DC=yourdomain,DC=com" -SearchScope Subtree |
Select-Object Name, DistinguishedName, Enabled
```

After running the `Disable-ADAccount` commands, you can run the verification commands again to confirm that the `Enabled` status for the affected users has changed to `False`.

> **Conclusion**
>
> By following these steps, you can efficiently disable user accounts in specific Active Directory Organizational Units using PowerShell. This approach provides a quick and non-destructive way to manage user access.

? Transferring FSMO Roles to Another Domain Controller

FSMO (Flexible Single Master Operations) roles are critical for Active Directory functionality. This guide shows how to transfer all FSMO roles to a new Domain Controller (`HQ-DC01`) using both GUI and PowerShell.

□□ FSMO Roles Overview

- **Schema Master**
- **Domain Naming Master**
- **PDC Emulator**
- **RID Master**
- **Infrastructure Master**

□□ Method 1: Transfer FSMO Roles via PowerShell

1. Open PowerShell as Administrator on **any DC**.
2. Run the following command to transfer all FSMO roles to `HQ-DC01`:

```
Import-Module ActiveDirectory
Move-ADDirectoryServerOperationMasterRole -Identity "HQ-DC01" -
OperationMasterRole 0,1,2,3,4 -Confirm:$false
```

This command transfers all five roles at once:

- 0 – PDC Emulator
- 1 – RID Master
- 2 – Infrastructure Master
- 3 – Schema Master
- 4 – Domain Naming Master

□ Verify the Transfer

```
Get-ADForest | Select-Object SchemaMaster, DomainNamingMaster
Get-ADDomain | Select-Object PDCEmulator, RIDMaster, InfrastructureMaster
```

□□ Method 2: Transfer FSMO Roles via GUI

1. Transfer RID, PDC, Infrastructure Master

1. Open **Active Directory Users and Computers (dsa.msc)**
2. Right-click the domain ? click **Operations Masters**
3. Go through the **RID**, **PDC**, and **Infrastructure** tabs
4. Click **Change** on each tab to transfer the role to `HQ-DC01`

2. Transfer Domain Naming Master

1. Open **Active Directory Domains and Trusts (domain.msc)**
2. Right-click **Active Directory Domains and Trusts** at the top left
3. Select **Operations Master**
4. Click **Change**

3. Transfer Schema Master

1. Run the following to register the Schema snap-in:

```
regsvr32 schmmgmt.dll
```

2. Run `mmc` ? Add Snap-in ? **Active Directory Schema**
3. Right-click **Active Directory Schema** ? **Change Active Directory Domain Controller…**
4. Select **HQ-DC01**
5. Then right-click again ? **Operations Master** ? Click **Change**

□□ Notes

- You must be a **Domain Admin** and a **Schema Admin** to transfer all roles.

- The Schema Master snap-in only connects to DCs that are **writable** and have a **replica of the schema** .
- If a DC is unreachable, roles must be **seized** rather than transferred.

Final Tip

Use `netdom query fsmo` to check current FSMO role holders at any time.

```
netdom query fsmo
```

? Login Banner via GPO- How to Set a Security Message at Login via Group Policy (GPO)

This tutorial explains how to display a login banner or legal notice when users sign into Windows devices on a domain. This is often used to show security warnings, acceptable use policies, or legal disclaimers.

Step 1: Open Group Policy Management Console

On a domain controller or a machine with the GPMC installed:

```
Start ? Run ? gpmc.msc
```

Step 2: Create or Edit a GPO

1. Navigate to your domain or an Organizational Unit (OU).
2. Right-click and select **"Create a GPO in this domain, and Link it here…"** or edit an existing GPO.

Step 3: Configure the Security Message

In the GPO editor, go to:

```
Computer Configuration
 ??? Policies
     ??? Windows Settings
         ??? Security Settings
             ??? Local Policies
                 ??? Security Options
```

Find and configure the following two settings:

- **Interactive logon: Message title for users attempting to log on**
- **Interactive logon: Message text for users attempting to log on**

Recommended General Message

Title:

```
Authorized Use Only
```

Text:

```
You are accessing a secured system.

This system is for authorized users only. By continuing, you agree to comply
with organizational policies and security guidelines.

All actions may be monitored and recorded. Unauthorized access is prohibited
and may lead to disciplinary action or legal consequences.

If you are not authorized, please disconnect immediately.
```

Step 4: Apply the GPO

To apply the policy immediately, run the following command on a client machine:

```
gpupdate /force
```

Then log off or reboot to verify that the message appears before login.

Note: Always test GPO changes in a controlled environment before deploying them network-wide.

? How to Configure Windows Updates to Run Outside Working Hours via Group Policy

This guide configures Windows devices to download and install updates only **outside of business hours** (between 4:30 PM and 6:30 AM) using Group Policy.

Step 1: Open Group Policy Management Console

```
Start ? Run ? gpmc.msc
```

Step 2: Create or Edit a GPO

1. Navigate to the appropriate **OU** or domain.
2. Right-click and select **"Create a GPO in this domain, and Link it here..."** or edit an existing one.
3. Right-click the GPO and choose **Edit**.

Step 3: Configure Windows Update Settings

Go to:

```
Computer Configuration
 ??? Policies
     ??? Administrative Templates
         ??? Windows Components
             ??? Windows Update
                 ??? Manage end user experience
```

Set Active Hours

Find the policy:

```
Turn off auto-restart for updates during active hours
```

Enable this setting and configure:

- **Start time:** 6:30 AM
- **End time:** 4:30 PM

This prevents auto-restart during business hours.

Configure Automatic Updates

Open the policy:

```
Configure Automatic Updates
```

Enable this policy and set it to:

```
4 - Auto download and schedule the install
```

Then set the scheduled install time outside business hours, for example:

- **Every day at 5:00 PM**

Step 4: Configure Automatic Maintenance (Optional)

To schedule general maintenance tasks (including updates):

```
Computer Configuration
 ??? Administrative Templates
     ??? Windows Components
         ??? Maintenance Scheduler
```

- **Enable** and set the time for automatic maintenance to run after hours (e.g., 5:00 PM).

Step 5: Apply and Test

Run this command on a target machine to apply the new settings:

```
gpupdate /force
```

Then verify by opening **Windows Update Settings** on a client machine and checking the active hours and scheduled install time.

Note: Clients must be running Windows 10 1607 or newer for Active Hours GPO to work.

How to Convert Windows Server 2025 Evaluation to Full Version (Standard/Datacenter)

Got Windows Server 2025 Evaluation installed and ready to move to the full Standard or Datacenter edition? Don’t worry; I’ve got you covered. This guide breaks it down step-by-step, so you can upgrade smoothly without any hiccups along the way.

winserv2025-2.PNG

Why Bother Upgrading the Evaluation Edition?

The evaluation version of Windows Server 2025 is brilliant for testing and getting a feel for things, but it has its drawbacks:

- It’s free, but only for 180 days.
- Some of the enterprise features you might need are locked unless you have a proper licence.
- And let’s not forget those persistent activation nags popping up.

Switching to the full version unlocks your server's full power and ensures everything runs like clockwork in production.

Before You Start: The Essentials

This should go without saying, especially if you’ve got important data on the server—but let’s not take any chances. Before diving into the conversion, make sure you’ve got the following covered:

- **Back Up Your Data**: This process is meant to keep everything intact, but it's always better to be safe than sorry. A solid backup is your safety net.
- **Get a Valid Licence Key**: You'll need a proper product key for either Windows Server 2025 Standard or Datacenter. (I grabbed mine from cjs-cdkeys.com for less than £25).
- **Admin Access**: You'll need admin privileges to run the commands and make changes to the system.

winserv2025-3.PNG

Converting the Edition

Here's the command you'll need to make the switch. Just replace XXXXX-XXXXX-XXXXX-XXXXX-XXXXX with your licence key:

Convert to the standard edition

Code:

```
DISM /online /Set-Edition:ServerStandard /ProductKey:XXXXX-XXXXX-XXXXX-XXXXX-
XXXXX /AcceptEula
```

Convert to datacentre edition

Code:

```
DISM /online /Set-Edition:ServerDatacenter /ProductKey:XXXXX-XXXXX-XXXXX-
XXXXX-XXXXX /AcceptEula
```

Heads up: This process will take a few minutes, and your server will restart automatically during the conversion. Make sure to plan for any necessary downtime beforehand to avoid surprises.

Switching from Windows Server 2025 Evaluation to Standard or Datacenter is a pretty straightforward task as long as you follow these steps carefully. Once upgraded, you'll have the full power of enterprise-grade features at your fingertips to keep everything running smoothly.

If this guide helped or you hit a snag along the way, don't hesitate to share your experience. Let's keep the TechSEO community strong by swapping tips and solutions!

Entra Cloud Sync / gMSA Troubleshooting Guide

How we fixed the Entra Provisioning Agent error on **HQ-DC01** when Active Directory could not resolve the **Managed Service Accounts** container correctly.

Summary: The final root cause was not just permissions or the existence of the `CN=Managed Service Accounts` container. The real issue was that the domain's `otherWellKnownObjects` attribute still pointed the Managed Service Accounts GUID to a **deleted object** under `CN=Deleted Objects`. Removing the stale `\0ADEL` reference and restoring the live mapping fixed the problem.

1. Environment

- **Domain:** aspirapa.org
- **Server used for repair:** HQ-DC01
- **Issue surface:** Entra Provisioning Agent / Entra Cloud Sync setup
- **Symptom:** The wizard failed while trying to create or locate the Managed Service Account container

2. Original Symptoms

During the Entra Cloud Sync setup, the wizard reported that it could not find the **Managed Service Accounts** container. Earlier troubleshooting also uncovered several foundational Active Directory issues that had to be corrected before the final fix would succeed.

> **Important:** This repair was the final step in a larger cleanup. Earlier problems included an outdated domain functional level, stale domain controller metadata, and DNS / domain discovery problems. Those needed to be addressed first.

3. Earlier Problems That Were Addressed First

Before the final container mapping repair, the following issues were identified and worked through:

1. **Domain functional level was too old.**
   The domain was initially identified as `Windows2008Domain`. This was raised to **Windows Server 2012 R2 domain functional level**, which is necessary for modern gMSA and Entra-related workflows.
2. **Old DC metadata needed cleanup.**
   Legacy domain controllers such as `CYBERDC01` and `ASPIRADC1` were no longer present and required metadata cleanup, including NTDSUTIL cleanup, DNS cleanup, and review of Active Directory Sites and Services.
3. **KDS root key and gMSA prerequisites were reviewed.**
   KDS keys were checked and a new KDS root key was added as part of the process.
4. **DNS / secure channel discovery was broken.**
   `nltest /sc_verify` failed with `1355 ERROR_NO_SUCH_DOMAIN`. The NIC DNS configuration was corrected so the domain controller pointed to internal AD DNS only, and DNS / SRV discovery tests were rerun successfully.
5. **Managed Service Accounts container checks were performed.**
   Eventually, `CN=Managed Service Accounts,DC=aspirapa,DC=org` was confirmed to exist, which proved that the final error was not simply “missing container.”

4. What Did Not Work

- Attempting to treat the issue as only a permissions problem
- Assuming the error meant the container did not exist
- Trying to add the mapping to `wellKnownObjects` instead of the correct attribute
- Trying to force the repair through LDP while the wrong target attribute was being used

> **Key lesson:** In this case, the Managed Service Accounts GUID was already present in the directory metadata, but it pointed to a deleted object. That is why the live container existed while the Entra agent still failed.

5. How the Real Root Cause Was Found

PowerShell inspection of the domain root object showed that the built-in `wellKnownObjects` attribute did not contain the MSA mapping, but the `otherWellKnownObjects` attribute did. The problem was that the MSA GUID `1EB93889E40C45DF9F0C64D23BBB6237` pointed to a deleted object path instead of the live container.

Command used to inspect the live domain root mapping

```
Get-ADObject "DC=aspirapa,DC=org" -Properties otherWellKnownObjects |
Select-Object -ExpandProperty otherWellKnownObjects
```

Problematic output

```
B:32:683A24E2E8164BD3AF86AC3C2CF3F981:CN=Keys,DC=aspirapa,DC=org
B:32:1EB93889E40C45DF9F0C64D23BBB6237:CN=Managed Service
Accounts\0ADEL:5b87c493-325b-45b2-9c8c-bd17424d981b,CN=Deleted
Objects,DC=aspirapa,DC=org
```

That second line was the smoking gun. The MSA GUID existed, but it referenced a deleted object instead of:

```
CN=Managed Service Accounts,DC=aspirapa,DC=org
```

6. Final Repair That Worked

The successful repair was to remove the stale deleted-object mapping from `otherWellKnownObjects` and replace it with the correct live container mapping.

PowerShell repair commands

```
$dn = "DC=aspirapa,DC=org"

$bad = "B:32:1EB93889E40C45DF9F0C64D23BBB6237:CN=Managed Service
Accounts\0ADEL:5b87c493-325b-45b2-9c8c-bd17424d981b,CN=Deleted
Objects,DC=aspirapa,DC=org"
$good = "B:32:1EB93889E40C45DF9F0C64D23BBB6237:CN=Managed Service
Accounts,DC=aspirapa,DC=org"

Set-ADObject -Identity $dn -Remove @{otherWellKnownObjects=$bad}
Set-ADObject -Identity $dn -Add @{otherWellKnownObjects=$good}

Get-ADObject $dn -Properties otherWellKnownObjects |
Select-Object -ExpandProperty otherWellKnownObjects
```

Expected good result

```
B:32:1EB93889E40C45DF9F0C64D23BBB6237:CN=Managed Service
Accounts,DC=aspirapa,DC=org
```

Replication after repair

```
repadmin /syncall /AdeP
```

7. Verification Steps

1. Confirm the stale `\0ADEL` reference is gone from `otherWellKnownObjects`.
2. Confirm the live mapping exists for the MSA GUID.
3. Run replication.
4. Return to the Entra Provisioning Agent wizard and click **Confirm** again.
5. Verify that the agent now proceeds without the Managed Service Accounts container error.

Useful verification commands

```
Get-ADObject "CN=Managed Service Accounts,DC=aspirapa,DC=org"

Get-ADObject "DC=aspirapa,DC=org" -Properties otherWellKnownObjects |
Select-Object -ExpandProperty otherWellKnownObjects

repadmin /syncall /AdeP
```

8. Full Recommended Troubleshooting Flow for Similar Cases

1. **Check domain functional level** and raise to at least 2012 / 2012 R2 if required.
2. **Clean stale DC metadata** for removed domain controllers.
3. **Check DNS client settings** on domain controllers and ensure they point only to internal AD DNS servers.
4. **Validate domain discovery** with `nltest`, `dcdiag`, and DNS SRV lookups.
5. **Confirm KDS root key** and other gMSA prerequisites.
6. **Confirm the live MSA container exists**.
7. **Inspect both** `wellKnownObjects` **and** `otherWellKnownObjects`.
8. **If the MSA GUID points to a deleted object, repair the mapping**.
9. **Sync replication** and rerun the Entra provisioning workflow.

9. Why This Matters

This issue can mislead administrators because the `Managed Service Accounts` container may exist and still not be resolvable by Entra or gMSA-related tools. The issue is not always the container itself. It can be the **directory reference to that container**.

> **Final outcome:** After removing the stale deleted-object entry and adding the correct live `otherWellKnownObjects` mapping, the Entra Cloud Sync / Provisioning Agent error was resolved successfully.

10. Copy/Paste Quick Reference

```
# Inspect the current mapping
Get-ADObject "DC=aspirapa,DC=org" -Properties otherWellKnownObjects |
Select-Object -ExpandProperty otherWellKnownObjects

# Repair the stale deleted-object reference
$dn = "DC=aspirapa,DC=org"
$bad = "B:32:1EB93889E40C45DF9F0C64D23BBB6237:CN=Managed Service
Accounts\0ADEL:5b87c493-325b-45b2-9c8c-bd17424d981b,CN=Deleted
Objects,DC=aspirapa,DC=org"
$good = "B:32:1EB93889E40C45DF9F0C64D23BBB6237:CN=Managed Service
Accounts,DC=aspirapa,DC=org"

Set-ADObject -Identity $dn -Remove @{otherWellKnownObjects=$bad}
Set-ADObject -Identity $dn -Add @{otherWellKnownObjects=$good}

# Verify the fix
Get-ADObject $dn -Properties otherWellKnownObjects |
Select-Object -ExpandProperty otherWellKnownObjects

# Replicate
repadmin /syncall /AdeP
```

Prepared for BookStack HTML use. This page is designed to be pasted into a BookStack HTML editor or imported as a standalone HTML reference.