Forest and Domain Function Upgrade from 2012r2 to 2016

Testing Notes-

Windows Server 2016:

Raising the domain & forest function level from 2012R2 to 2016 through the AD Domain and Trusts only raised the domain functional level.

After a quick google search, it proved that both domain and functional levels would not be raised by themselves,

Instead, once the domain and functional level were raised, the forest was to be raised separately.

For windows server 2016 to check the synchronization service manager, the user must have ADSyncAdmins.

I was able to create and sync users without it but to gain more administrative control and view of what is happening in the background this is a

A good tool to have to accomplish that.

Test Status-

Windows Server 2016:

All of the basic testing functions have been done and were successfully created & synced into our Azure AD.

Adding, modifying, and disabling trials done for:

Users/User Attributes

Groups

Roles (Within On-Prem AD)

Roles (Within Azure AD)

Images-

Windows Server 2016:

**Domain/Forest Functional Level**

A picture containing text

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

**User Add / Successful Sync**

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated



**User Disable**

Graphical user interface, text

Description automatically generated



Graphical user interface, text, application, email

Description automatically generated



**Groups Add**

Graphical user interface, text, application

Description automatically generated

Graphical user interface

Description automatically generated



**Group Removal:**

Graphical user interface, text, application

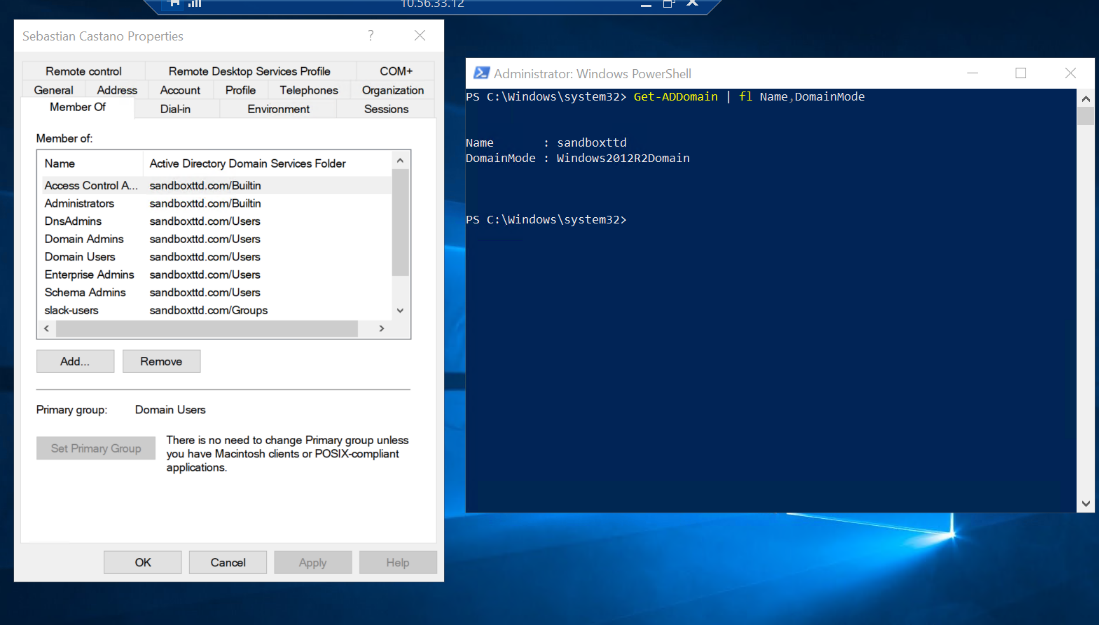
Description automatically generated

Graphical user interface, text, application, email

Description automatically generated



**Admin Roles(On-Prem AD)**



**Admin Roles(Azure AD)**

Graphical user interface, text, application, email

Description automatically generated

AAD Connect Break

How'd I got here?

I realized that Azure AD Connect was set to Auto Upgrade (doh!)

The premise of Project Malibu was to provide a step-by-step demonstration of upgrading the on-prem Active Directory domain and functional level, Azure AD Connect, and eventually a cloud-only environment.

By setting Azure AD Connect to Auto Upgrade I skipped a crucial step in the process and therefore I wanted to retrace my steps and present the information as intended.

Return to testing phase uno!

-Removed AADC

-Downgrade on-prem AD forest and domain functional level back to 2012

-Install the older version 1.6.16.0 of AADC

Problems that I ran into:

Removing AADC from on-prem AD broke the sync between on-prem and Azure. I tried multiple tests to get AADC connected to Azure, InteractiveAuth script, and removing MFA from Azure users (Microsoft's suggestion).

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application

Description automatically generated

After the connection was broken, I was unable to sign in to the Azure tenant.

The log-in failed attempts were receiving an error number 80014(Authentication Agent timed out) in the sign-in logs.

-I was only able to log in with my user account to the Azure tenant on the server itself.

-The admin account could log in as well, but it's a cloud-only user.

Graphical user interface, text, application

Description automatically generated



What do I think is causing this?

The break in Azure AD & on-prem AD would not allow Azure to authenticate the users' passwords (Pass-through Authentication).

With that break, Azure is unable to authenticate, therefore it is unable to make the connection to Azure so Azure AD Connect can't connect to reestablish the connection with Azure AD and on-prem AD.

I believe there might be something cached somewhere where it is trying to utilize the old connection instead of establishing a new connection to Azure.

Moving forward:

The most important aspect of this project and the overall sandbox is to have a fully functioning environment that demonstrated the process of the upgrades and eventual migration to cloud-only. The upgrades done have shown that upgrading the on-prem domain and forest functional level from 2012r2 to 2016 and the Azure AD Connect auto-upgrade to its latest possible version was seamless. Moving forward I will try to get the environment back working to where it once was.

Final Result

After many attempts to connect Azure AD Connect from on-prem AD to Azure AD, finally, after upgrading the environment back to the 2016 forest and domain functional level, and version 2.1.16.0 of Azure AD Connect, the Azure AD Connect has once again started working, and is now syncing with our cloud environment. MFA was removed from the Azure tenant and the Azure AD Connect was run through an InteractiveAuth CMD. All changes to our on-prem AD are now visible on our Azure AD.

Graphical user interface, text

Description automatically generated

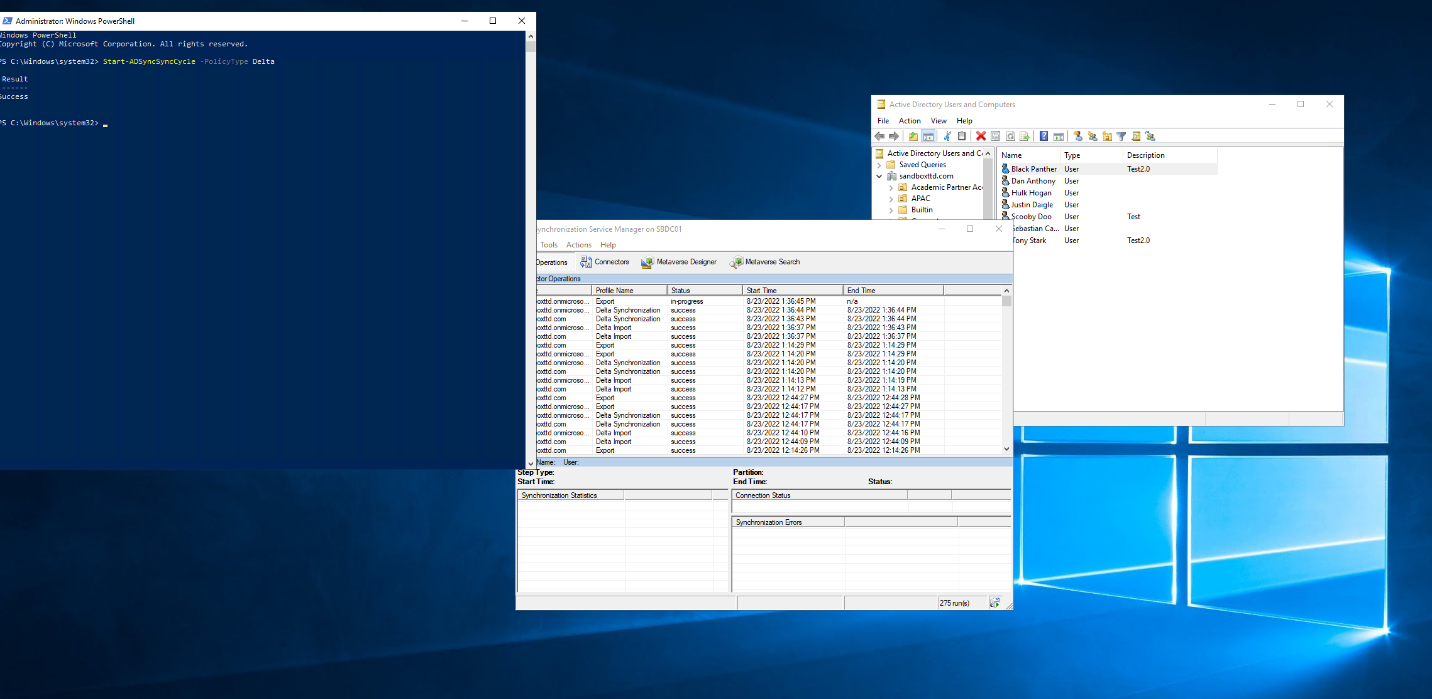
Text

Description automatically generated

Graphical user interface, application, table

Description automatically generated

A New Sync test was done to demonstrate that our environment is fully functioning and is connected again with Azure AD.



Graphical user interface, application

Description automatically generated



Azure OU Break for Cloud-Only Test

Procedure-

-Created an on-prem OU within our domain.

* Cloud-OnlyOU

-Changed the Azure sync settings:

* Customize synchronization options
* Connect to Azure AD
* Select directories (Should already be selected)
* Select sync selected domains and OUS
* Uncheck our cloud-only OU.

Graphical user interface, application

Description automatically generated

The first test is done with the user Peter Parker, in the image below we can see that the user is in our Azure tenant and is synced with our on-premises AD. We will also note that we have a total of 17 users within our Azure tenant.

Graphical user interface, application, table

Description automatically generated

Next, move the user to the Cloud-OnlyOU that we created and changed the Azure Sync settings to not sync with our Azure AD tenant.

Graphical user interface, text, application, email

Description automatically generated

Moving the user to the Cloud-OnlyOU has successfully removed the user from our Azure tenant.

Graphical user interface, application, Word

Description automatically generated

Lastly, we connect Powershell with our Azure tenant and then restore the user to our tenant.

Graphical user interface, text

Description automatically generated

We successfully restored the user that was deleted with our sync break with the Cloud-OnlyOU and it shows that the user is not syncing with our on-premises AD.



After restoring the user to Azure, we were able to successfully log in to our Azure tenant.

Graphical user interface, text, application

Description automatically generated

Test #2:

Add the user to the Cloud-OnlyOU, and restore the user from the Azure UI.

Graphical user interface, text, application, email

Description automatically generated

After restoring the user utilizing the Azure UI, I was able to successfully log the user in.

Graphical user interface, text, application

Description automatically generated

Password Reset:

The next test was to reset the password of one of the cloud-only users.



Graphical user interface, text, application, email

Description automatically generated



Microsoft allowed the user to reset their password and sign into their account.

Graphical user interface, application

Description automatically generated

