Azure AD Connect Custom Settings

This guide provides setup requirements and steps to demonstrate how to extend an on-premises multi-forest Windows Server Active Directory environment to Azure Active Directory using Azure AD Connect.

Contents

[Pre-Requisites 1](#_Toc430074833)

[Setup 1](#_Toc430074834)

[Demo Steps 4](#_Toc430074835)

[Clean Up 9](#_Toc430074836)

## Pre-Requisites

This section lists the pre-requisites required for this demonstration.

* An Azure subscription
* Visual Studio 2015
  + Make sure you have the Azure Tools installed.
  + Install the PowerShell tools for Visual Studio from [here](https://visualstudiogallery.msdn.microsoft.com/c9eb3ba8-0c59-4944-9a62-6eee37294597).
* You need to be able to invoke a PowerShell script from your computer. If you’re not already able to do this, then open a PowerShell prompt and run **Set-ExecutionPolicy –ExecutionPolicy Unrestricted**
* You must have completed the Setup section for demo 1. See ***1\_Demo\_App\_Integration.docx***.
* You must have completed demo 2 – Enable Directory Integration.

## Setup

Estimated time: 30 minutes

|  |  |
| --- | --- |
| 1. Open Windows Explorer. 2. Copy the folder **.\Demos\2C\_AAD\_Connect\_Custom\_Settings** to **c:\azurecoe\demos\identity**. 3. Open Visual Studio. 4. Create a **v1 storage account (classic)** in your Azure subscription that you can use for deployment purposes. It won’t be used by the resources in the environment. This is just for deploying artifacts to during deployment so Locally Redundant is fine. |  |
| 1. Open the solution at **c:\azurecoe\demos\identity\AAD\_Connect\_Custom\_Settings\AAD\_Connect-Demo-Solution.sln**. 2. Right-click the project in Solution Explorer and select **Deploy > New Deployment**. 3. Select the option to **Create a new resource group** in the **Resource group** field. 4. Set the **Artifacts storage account** to the storage account you created in step 2 that is used for deployment purposes. |  |
| 1. Click **Deploy**. 2. In the Edit Parameters dialog, set the adVNETLocation to the location closest to you. Leave the <auto-generated> fields alone. 3. Click **Save**. |  |
| 1. Wait about 20 minutes for the deployment to finish. |  |
| 1. Open the Azure portal. 2. Go to the **adVNET** blade and then to the **DNS Servers** blade. 3. Configure DNS Servers in the adVNET resource.    1. In the AADconnect-Demo-Solution resource group blade, click on **adVNET** in the Summary part.    2. In the adVNET blade, click on **Settings** in the toolbar.    3. In the Settings blade, click on **DNS Servers**.    4. In the DNS Servers blade:       1. Set **DNS servers** to **Custom DNS**.       2. Set **Primary DNS** sever to **10.0.0.4**.       3. Set **Secondary DNS** server to **10.0.1.4**.       4. Click the **Save** button in the toolbar. |  |
| 1. **Restart** the **adVM1** and **adVM2** virtual machines.    1. In the virtual machine blade, click the Restart button in the toolbar.   Wait for **adVM1** and **adVM2** to restart before continuing. |  |
| 1. In the **adVM1** blade, click **Connect** in the toolbar. 2. Sign-in using the following credentials.    1. Set user name to **.\adminuser**    2. Set password to **P@ssword1** 3. Open Windows Explorer and navigate **to C:\Program Files\WindowsPowerShell\Modules\ADInitUsers**. 4. Right-click on **VM1-ConfigureForDemo** and select **Run with PowerShell**. 5. Keep the RDP connection to this virtual machine open. |  |
| 1. In the **adVM2** blade, click **Connect** in the toolbar. 2. Sign-in using the following credentials.    1. Set user name to **.\adminuser**    2. Set password to **P@ssword1** 3. Open Windows Explorer and navigate **to C:\Program Files\WindowsPowerShell\Modules\ADInitUsers**. 4. Right-click on **VM2-CreateUsers** and select **Run with PowerShell**. |  |
| 1. Open Server Manager if it is not already opened. 2. Click **Tools > Active Directory Domains and Trusts**. 3. Right-click **contoso.com** and select **Properties**. 4. In the contoso.com Properties window, click on the **Trusts** tab. 5. Click the **New Trust** button. 6. In the New Trust Wizard…    1. Click **Next** on the Welcome screen.    2. Set the **Trust Name** to **azurecoe.com** and click **Next**.    3. Set the **Trust Type** to **Forest trust** and click **Next**.    4. Set the **Direction of Trust** to **Two-Way** and click **Next**.    5. Set the **Sides of Trust** to **Both this domain and the specified domain** and click **Next**.    6. Set **User Name** to **azurecoe.com\adminuser**, **Password** to **P@ssword1**, and click **Next**.    7. Set **Outgoing Trust Authentication Level-Local Forest** to **Forest-wide authentication** and click **Next**.    8. Set **Outgoing Trust Authentication Level-Specified Forest** to **Forest-wide authentication** and click **Next**.    9. In the **Trust Selections Complete** page of the wizard click **Next**.    10. In the **Trust Creation Complete** screen click **Next**.    11. Set **Confirm Outgoing Trust** to **Yes, confirm the outgoing trust** and click **Next**.    12. Set **Confirm Incoming Trust** to **Yes, confirm the incoming trust** and click **Next**.    13. In the **Completing the New Trust Wizard** page click **Finish**.    14. Click **OK** to close the contoso.com properties. |  |
| 1. Go back to the RDP session for adVM1. 2. Open Windows Explorer and navigate **to C:\Program Files\WindowsPowerShell\Modules\ADConnect**. 3. Right-click on **AzureADConnect** and select **Install**. 4. In the **Welcome to Azure AD Connect** page, click the check box to Agree to the terms and then click **Continue**. 5. When you get to the **Express Settings** page ***STOP***.   **The demo will continue from here.** |  |

## Demo Steps

Estimated time: 15 minutes

|  |  |
| --- | --- |
| 1. In the **Azure AD Connect** installation tool, click **Customize**. 2. In the Install required components click on a couple of options to show that you can customize the installation of Azure AD Connect. Make sure to **uncheck** any options **before continuing**. 3. Click **Install**. |  |
| 1. While Azure AD Connect is installing, show the trust relationship between the azurecoe.com and contoso.com domains.    1. Open **Server Manager** if it is not already opened.    2. Click **Tools > Active Directory Domains and Trusts**.    3. Right-click **azure.com** and select **Properties**.    4. In the **azurecoe.com Properties** window, click on the **Trusts** tab. Explain the trust relationship between the two domain controllers.    5. Click **Cancel**. 2. If Azure AD Connect is still installing, show the users and groups in the azurecoe.com and contoso.com forests. |  |
| 1. Go back to Azure AD Connect. 2. In the User sign-in page, show the sign-in methods that were discussed in the slides. Click on **Federation with AD FS** and explain this configuration option. This is a huge benefit of Azure AD Connect. 3. Select in **Password Synchronization** and click **Next**. |  |
| 1. In the Connect to Azure AD screen, enter the following credentials for the Azure CoE directory:    1. Set **USERNAME** to **clark@azurecoe01.onmicrosoft.com**    2. Set **PASSWORD** to **P@ssword1**    3. Click **Next**. |  |
| 1. In the Connect your directories screen, enter the following:    1. Set **FOREST** to **azurecoe.com**    2. Set **USERNAME** to **AZURECOE.COM\adminuser**    3. Set **PASSWORD** to **P@ssword1**    4. Click **Add Directory**.    5. Set **FOREST** to **contoso.com**    6. Set **USERNAME** to **CONTOSO.COM\adminuser**    7. Set **PASSWORD** to **P@ssword1**    8. Click **Add Directory**.    9. Click **Next**. |  |
| 1. In the Uniquely identifying your users screen, explain some of the options. As an example, Microsoft uses a custom attribute to identify users based on employee Id. Accept the default settings and click **Next**. |  |
| 1. In the Filter users and devices screen, set **GROUP** to **Development** for the **contoso.com** forest and click **Resolve**. This will restrict synchronization for this forest to just that group. The azurecoe.com forest will still synchronize all users, groups and devices (the default). 2. Click **Next**. |  |
| 1. In the Optional features screen, select the following options:    1. Azure AD App and attribute filtering    2. Password writeback   Note: The options that are grayed out require Azure AD Premium. But, you can still explain what they are for. Don’t select Directory extension attribute sync. If you do, the demo will still work but you will likely get an error message about directory extensions not getting installed which you will have to explain. So, until this bug is fixed, it’s better to just avoid it. But, you can still explain the feature.   1. Click **Next**. |  |
| 1. In the Azure AD apps screen, explain that this page exists in the wizard because you selected the optional feature for Azure AD App and attribute filtering. By default, the attributes/claims needed by these applications are synchronized to Azure AD. If you wanted to restrict the attributes that are synchronized to support a specific set of applications, then you could do that here. 2. Click **Next**. |  |
| 1. In the Azure AD attributes screen, explain that these are the attributes that would by default be synchronized based on your selection of the Azure AD apps in the previous screen. If you wanted to further restrict the attributes that are synchronized then you could do that here. 2. Click the link to view the list of attributes (using Notepad). 3. Close Notepad. 4. Click **Next**. |  |
| 1. In the Ready to configure page, review the settings with the audience. Explain the checkmark indicating to start the synchronization process after configuring Azure AD Connect. 2. Click **Install**. |  |
| 1. The install will take about 2-3 minutes. While this is working field a couple of questions from the audience. | Questions??? |
| 1. Click **Exit**. |  |
| 1. Open Windows Explorer and navigate to **C:\Program Files\Microsoft Azure AD Sync\UIShell**. 2. Double-click on **miisclient.exe**. 3. Click on the **Connectors** button in the toolbar. 4. Explain the concept of connectors and point out there is one for Azure AD and each of the forests. |  |
| 1. Click on the **Operations** button in the toolbar. 2. Show the operations for each of the connectors. Point out the Full **Synchronization** operations and then the links at the bottom of each operation showing statistics for the operation. 3. Click on **contoso.com**. 4. In the statistics section, show the links to the two **Provisioning Adds**. This is indicating that two objects were synchronized to Azure AD. 5. Click on the **Provisioning Adds** link to show the object details for the two objects. 6. Double-click through each to reveal that the following two users were synchronized from contoso.com to Azure AD.    1. Larry Goza    2. David Devready   Recall that when we ran Azure AD Connect we filtered synchronization to only users in the **Development** group. There are actually 4 users in the contoso.com directory, but Larry and David are the only two in the Development group.   1. Click **Close**. 2. Click **Close**.   A couple of key things to point out here:   * 1. The Azure AD Connect sync schedule. The connectors will sync every 3 hours by default. This can be changed, but that is the default.   2. The Synchronization Service Manager only shows directory objects getting synced (added or deleted for example). Password syncs are not shown in this tool. For those, you need to use the event log if you want to see verify. |  |
| 1. In Server Manager, click **Tools > Active Directory Users and Computers**. 2. Right-click on user **Adam** and select **Reset Password**.    1. Change the password to **P@ssword2**    2. Click **OK**. |  |
| 1. In Server Manager, click **Tools > Event Viewer**. 2. Click on the **Application** event log. 3. Locate events 650, 656, 651, and 657. These are the result of changing Adams password. 4. Go back to the Synchronization Service Manager and point out that this password change didn’t result in any new operations. Reiterate that Password Sync and Directory Sync are two separate concepts in this configuration. |  |
| 1. Open the Azure Management Portal in your browser. 2. Go to the **USERS** page for the Azure CoE directory. 3. Show the users that were synchronized from the azurecoe.com and contoso.com directories. These users will show Local Active Directory in the SOURCED FROM column. |  |
| 1. Go to the **GROUPS** page and show the security groups that were synchronized. The Finance group from the azurecoe.com directory was synchronized as well as the membership for the group. Point out that there were additional groups created that are used to manage directory integration with the on-premises directories. |  |

## Clean Up

To clean up after this demo perform the following steps:

1. In the Azure Management Portal, go to the Azure CoE Directory.
   1. In the DIRECTORY INTEGRATION page, set Directory Sync to DEACTIVATED.
   2. Click SAVE. Wait for this change to complete before continuing.
   3. In the USERS page, delete all the users except Clark Griswold (Admin) and your service account user which cannot be deleted anyway.
   4. In the GROUPS page, delete all the groups.
2. On your local computer, run PowerShell ISE as Administrator.
   1. Run the following script. When prompted, sign-in using [clark@azurecoe01.onmicrosoft.com](mailto:clark@azurecoe01.onmicrosoft.com) and password P@ssword1.

Connect-MsolService

Get-MsolGroup | Remove-MsolGroup -Force

Get-MsolServicePrincipal | Remove-MsolServicePrincipal

* 1. You will see some error output. This is a bug and can be ignored.

1. Go back to the USERS page in the Azure Management portal.
2. Delete the Clark Griswold user.
3. Click the back arrow to show all the directories in your subscription.
4. Highlight Azure CoE and click the DELETE button at the bottom of the screen. This may not actually work. It’s hit or miss. Until the PG decides to finally fix this nonsense this is a best effort. More information here. <https://support.microsoft.com/en-us/kb/2967860>