Lab-04-05-06

Lab 04-Configure external collaboration settings

Lab scenario

You must enable external collaboration settings for your organization for approved guests access.

Estimated timing: 5 minutes

Exercise 1 - Allowing guest users to be invited into your organization

Task - Configure external collaboration settings

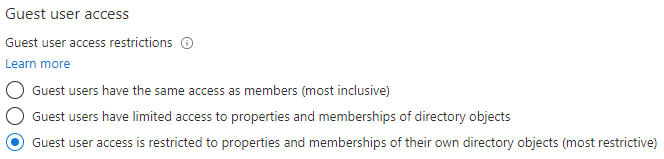
1. Sign in to the [https://portal.azure.com](https://portal.azure.com/) as a tenant administrator.
2. Select **Azure Active Directory**.
3. Select **External Identities > External collaboration settings**.
4. Select the **Email one-time passcode** notification link you see near the top of the screen.

**Note** - A one-time passcode is a very secure way to invite a user to join your organization.

1. Select on Home > Contoso Marketing > **External Identities** to return to the previous screen.
2. Select **External Collaboration Settings** on the left
3. Under **Guest user access**, review access levels that are available and then select **Guest user access is restricted to properties and memberships of their own directory objects (most restrictive)**.

**NOTE**

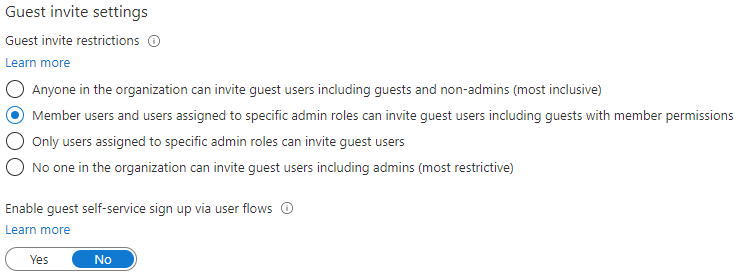
* + Guest users have the same access as members (most inclusive): This option gives guests the same access to Azure AD resources and directory data as member users.
  + Guest users have limited access to properties and memberships of directory objects: (Default) This setting blocks guests from certain directory tasks, like enumerating users, groups, or other directory resources. Guests can see membership of all non-hidden groups.
  + Guest user access is restricted to properties and memberships of their own directory objects (most restrictive): With this setting, guests can access only their own profiles. Guests are not allowed to see other users’ profiles, groups, or group memberships.

[](https://microsoftlearning.github.io/SC-300-Identity-and-Access-Administrator/Instructions/Labs/media/lp1-mod3-guest-user-access-restrictions.png)

1. Under **Guest invite settings**, select **Member users and users assigned to specific admin roles can invite guest users including guests with member permissions**!

**NOTE**

* + Anyone in the organization can invite guest users including guests and non-admins (most inclusive): To allow guests in the organization to invite other guests including those who are not members of an organization, select this radio button.
  + Member users and users assigned to specific admin roles can invite guest users including guests with member permissions: To allow member users and users who have specific administrator roles to invite guests, select this radio button.
  + Only users assigned to specific admin roles can invite guest users: To allow only those users with administrator roles to invite guests, select this radio button. The administrator roles include Global Administrator, User Administrator, and Guest Inviter.
  + No one in the organization can invite guest users including admins (most restrictive): To deny everyone in the organization from inviting guests, select this radio button.
  + If Members can invite is set to No and Admins and users in the guest inviter role can invite is set to Yes, users in the Guest Inviter role will still be able to invite guests.

[](https://microsoftlearning.github.io/SC-300-Identity-and-Access-Administrator/Instructions/Labs/media/lp1-mod3-guest-user-invite-settings.png)

1. Under **Collaboration restrictions**, review the available options and accept the default settings.

**IMPORTANT**

* + You can create either an allow list or a deny list. You can’t set up both types of lists. By default, whatever domains are not in the allow list are on the deny list, and vice versa.
  + You can create only one policy per organization. You can update the policy to include more domains, or you can delete the policy to create a new one.
  + The number of domains you can add to an allow list or deny list is limited only by the size of the policy. The maximum size of the entire policy is 25 KB (25,000 characters), which includes the allow list or deny list and any other parameters configured for other features.
  + This list works independently from OneDrive for Business and SharePoint Online allow/block lists. If you want to restrict individual file sharing in SharePoint Online, you need to set up an allow or deny list for OneDrive for Business and SharePoint Online.
  + The list does not apply to external users who have already redeemed the invitation. The list will be enforced after the list is set up. If a user invitation is in a pending state, and you set a policy that blocks their domain, the user’s attempt to redeem the invitation will fail.

1. When finished, **save** your changes.

# Lab 05: Add guest users to the directory

## Lab scenario

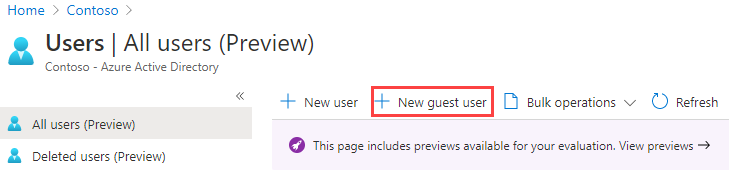
Your company works with many vendors and, on occasion, you need to add some vendor accounts to your directory as a guest.

#### Estimated time: 20 minutes

### Exercise 1 - Add guest users to the directory

#### Task - Add the guest user

1. Sign in to the [https://portal.azure.com](https://portal.azure.com/) as a user who is assigned a limited administrator directory role or the Guest Inviter role.
2. Select **Azure Active Directory**.
3. Under **Manage**, select **Users**.
4. Select **New guest user**.

[](https://microsoftlearning.github.io/SC-300-Identity-and-Access-Administrator/Instructions/Labs/media/lp1-mod3-new-guest-user-menu-selection.png)

1. On the New user page, select **Invite user** and then add your information as the guest user.

**NOTE** - Group email addresses are not supported; enter the email address for an individual. Also, some email providers allow users to add a plus symbol (+) and additional text to their email addresses to help with things like inbox filtering. However, Azure AD does not currently support plus symbols in email addresses. To avoid delivery issues, omit the plus symbol and any characters following it up to the @ symbol.

1. Enter an email address, such as **sc300externaluser1@sc300email.com**.
2. When complete, select **Invite**.
3. On the Users page, verify your account is listed and, in the **User type** column, verify **Guest** is shown.

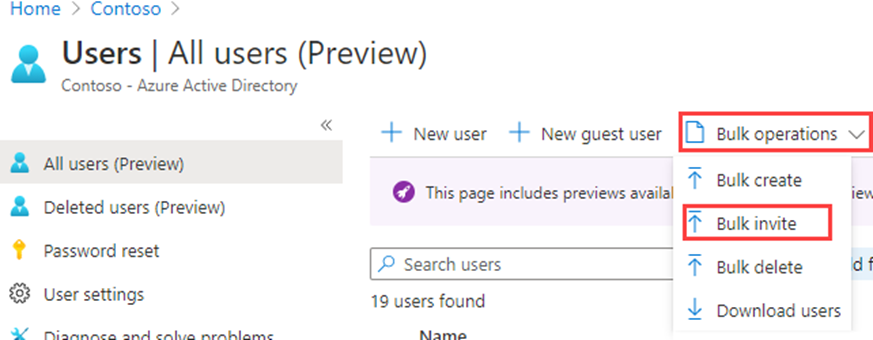
After you send the invitation, the user account is automatically added to the directory as a guest.

### Exercise 2 - Invite guest users in bulk

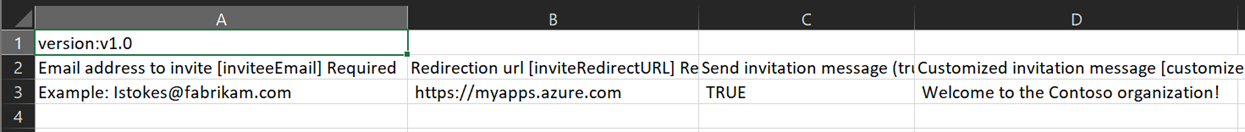
#### Task 1 - Bulk user invite

A recent partnership has been established with another company. For now, employees of the partner company will be added as guests. You need to ensure you can import multiple guest users at one time.

1. Sign in to the [https://portal.azure.com](https://portal.azure.com/) as your Global Administrator.
2. In the navigation pane, select **Azure Active Directory**.
3. Under **Manage**, select **Users**.
4. On the Users page, on the menu, select **Bulk operations > Bulk invite**.

[](https://microsoftlearning.github.io/SC-300-Identity-and-Access-Administrator/Instructions/Labs/media/lp1-mod3-bulk-invite-option.png)

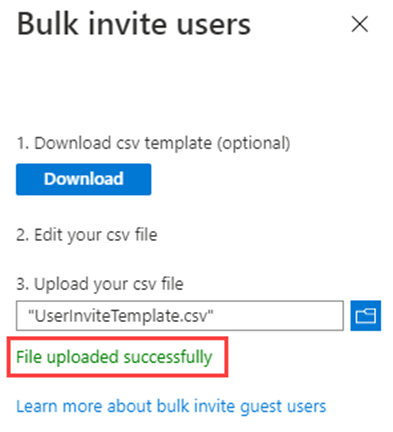
1. In the Bulk invite users pane, select **Download** to a sample CSV template with invitation properties.
2. Using an editor to view the CSV file, review the template.
3. Open the .csv template and add a line for each guest user. Required values are:
   * **Email address to invite** - the user who will receive an invitation
   * **Redirection url** - the URL to which the invited user is forwarded after accepting the invitation.

[](https://microsoftlearning.github.io/SC-300-Identity-and-Access-Administrator/Instructions/Labs/media/lp1-mod3-template-csv.png)

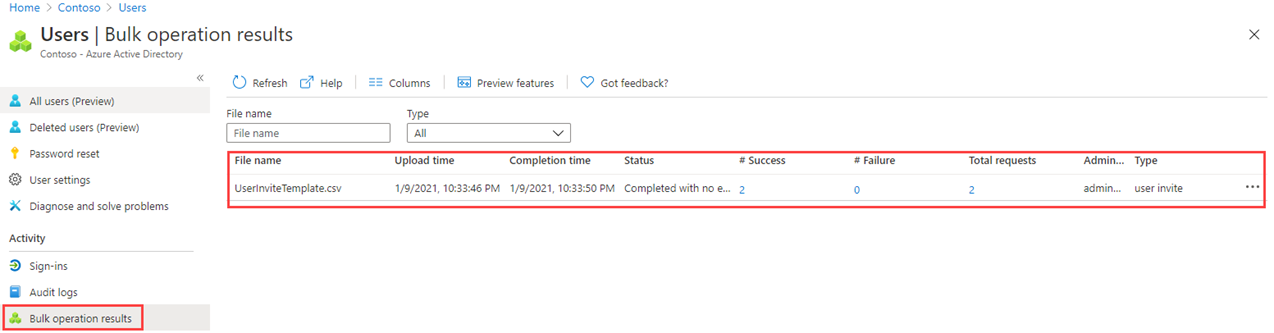
1. Save the file.
2. On the Bulk invite users page, under **Upload your csv file**, browse to the file.

**Note** - When you select the file, validation of the .csv file starts.

1. After the file contents are validated, you will see **File uploaded successfully**. If there are errors, you must fix them before you can submit the job.

[](https://microsoftlearning.github.io/SC-300-Identity-and-Access-Administrator/Instructions/Labs/media/lp1-mod3-bulk-invite-users-upload-csv.png)

1. When your file passes validation, select **Submit** to start the Azure bulk operation that adds the invitations.
2. To view the job status, select **Select here to view the status of each operation**. Or, you can select **Bulk operation results** in the Activity section. For details about each line item within the bulk operation, select the values under the **# Success**, **# Failure**, or **Total Requests** columns. If failures occurred, the reasons for failure will be listed.

[](https://microsoftlearning.github.io/SC-300-Identity-and-Access-Administrator/Instructions/Labs/media/lp1-mod3-bulk-operations-results.png)

1. When the job completes, you will see a notification that the bulk operation succeeded.

#### Task 2 - Invite guest users with PowerShell

1. Open PowerShell as an administrator. This can be done by searching for PowerShell in Windows and choosing Run as administrator.
2. You will need to add the Azure AD PowerShell module, if you have not used it before. Run the command: Install-Module AzureAD. When prompted, select “Y” to continue.

CodeCopy

Install-Module AzureAD

1. Confirm that the module installed correctly by running the command:

CodeCopy

Get-Module AzureAD

1. Next, you will need to login to Azure by running:

CodeCopy

Connect-AzureAD

1. The Microsoft login window will appear for you to login to Azure AD.
2. To verify that you are connected and to see existing users, run:

CodeCopy

Get-AzureADUser

1. You are ready to invite a guest user. The following command will be populated with the user information and run. If you have more than one user to add, you can use a notepad txt file to add the user information and copy/paste into PowerShell.

CodeCopy

New-AzureADMSInvitation -InvitedUserDisplayName "Display" -InvitedUserEmailAddress name@emaildomain.com -InviteRedirectURL https://myapps.microsoft.com -SendInvitationMessage $true

You now know how to invite users within the Azure AD portal, Microsoft 365 Admin center, Bulk invitations with a csv file, and inviting users with PowerShell commands.

# Lab 06: Add a federated identity provider

## Lab scenario

Your company works with many vendors and, on occasion, you need to add some vendor accounts to your directory as a guest and allow them to use their Google account to sign-in.

#### Estimated time: 25 minutes

### Exercise 1 - Configure identity providers

#### Task 1 - Configure Google to be used as an identity provider

**Note** - For this exercise, you will need a Gmail account on Google. Create a new Google account and then follow the steps for the exercise.

1. Go to the Google APIs at https://console.developers.google.com, and sign in with your Google account. We recommend that you use a shared team Google account.
2. Accept the terms of service if you’re prompted to do so.
3. Create a new project: At the top of the page, select the project menu to open the Select a project page. Choose New Project. Leave the remaining fields with the default settings.
4. On the New Project page, give the project a name (for example, **MyB2BApp**), and then select **Create**.
5. Open the new project by selecting the link in the Notifications message box or by using the project menu at the top of the page.
6. In the left menu, select **APIs & Services**, and then select **OAuth consent screen**.
7. Under User Type, select **External**, and then select **Create**.
8. On the **OAuth consent screen**, under App information, enter an App name, such as **Azure AD**.
9. Under User support email, select an email address. This should include the email address that you used to log into Google.
10. Under Authorized domains, select **Add domain**, and then add the microsoftonline.com domain.

CodeCopy

microsoftonline.com

1. Under Developer contact information, enter the email address for the lab account that you used to sign into the portal.
2. Select **Save and continue**.
3. In the left menu, select **Credentials**.
4. Select **+ Create credentials**, and then select **OAuth client ID**.
5. In the Application type menu, select Web application. Give the application a suitable name, like Azure AD B2B. Under **Authorized redirect URIs**, add the following URIs:

CodeCopy

- https://login.microsoftonline.com

- https://login.microsoftonline.com/te/\*\*tenant ID\*\*/oauth2/authresp

(where <tenant ID> is your tenant ID)

- https://login.microsoftonline.com/te/\*\*tenant name\*\*.onmicrosoft.com/oauth2/authresp

(where <tenant name> is your tenant name)

1. Select Create. Copy your **client ID** and **client secret**. You’ll use them when you add the identity provider in the Azure portal.
2. You can leave your project at a publishing status of Testing and add test users to the OAuth consent screen. Or you can select the Publish app button on the OAuth consent screen to make the app available to any user with a Google Account.

#### Task 2 - Configure Azure AD for Google federation

1. Sign in to the [https://portal.azure.com](https://portal.azure.com/) as a user who is assigned a limited administrator directory role or the Guest Inviter role.
2. Select **Azure Active Directory**.
3. Under **Manage**, select **External Identities**.

|  |  |
| --- | --- |
| Microsoft provides a direct federation for **Google** as an identity provider.  This can be initiated by selecting **+ Google** from the \*\*External Identities | All identity providers\*\* page |

1. After selecting + Google, another page will open with additional information that is required to configure Google as an identity provider.
2. Enter the client ID and client secret you obtained earlier. Select Save.
3. This completes the configuration of Google as an identity provider.

**Note** - You may receive a failed to add Google as a social identity provider message. Navigate out of that screen and back to the External Identities and Google should be available in the list.

|  |  |
| --- | --- |
| If you used an existing Gmail account, remember to delete the account with \*\*External Identities | All identity providers\*\*. You can also return to the Google developer console and delete the project that you created. |

1. **NOTE** - Use this link to complete the steps for finding the Google Client ID and Client secret. https://docs.microsoft.com/azure/active-directory/external-identities/google-federation
2. **Note** - Facebook can also easily be configured as an identity provider. These steps can be accessed here: https://docs.microsoft.com/azure/active-directory/external-identities/facebook-federation. If you prefer to use a Facebook account and not Google for this exercise, you may complete this option.
3. Once the setup is complete, you can open a private browser and enter the following web address:

CodeCopy

login.microsoftonline.com

1. Enter the **Google** email address and password that you created. You should then enter the Microsoft online portal for app access.

# Lab 07: Add Hybrid Identity with Azure AD Connect

**Note** - This lab requires an Azure Pass. Please see lab 00 for directions.

**Note 2** - This lab should be treated as optional, if you have ever worked with Azure AD Connect before. This lab require at least 30 minutes of setup time to deploy and configure the Domain Controler with the script. If you are going to do the lab, it is recommended that you launch the script and work on other activities while it runs in the background.

## Lab scenario

Your company works has Active Directory Domain Services on-premises. They would like to continue to use on-premises Active Directory as their identity and access management solution, but also require the ability for users to access cloud applications with the same username and password.

#### Estimated time: 60 minutes

### Exercise 1 - Setup On-Premises infrastructure

#### Task 1 - Create the on-premises Active Directory infrastructure

1. Deployment template can be accessed at this link: [On-Premises Test Lab Guide](https://github.com/maxskunkworks/TLG/tree/master/tlg-base-config_3-vm).

**Note to learners and MCTs** - The deployment of this template can take 30-60 minutes, so be ready to take a break at this step or run the deployment before a lecture section of the course.

**Note to Lab providers** - If possible, it would be helpful to students to complete and deploy as part of the lab environment setup.

1. On the **TLG (Test Lab Guide) - 3 VM Base Configuration (v1.0)** page, select **Deploy to Azure**.

**Note** - The 3 VM Base Configuration provisions a Windows Server 2016 Active Directory domain controller named DC1 using the domain name you specify, and a domain member server named APP1 running Windows Server 2016. It also offers an option to provision a client VM running Windows 10, however we will not be using it in our lab (primarily due to licensing requirements applicable when running Windows 10 VMs in Azure). The domain member server (APP1) has automatically installed .NET 4.5 and IIS.

**Note** - The VM that is required for this lab is **DC1**. If you are using an Azure Pass, there is a limitation of 2 VMs, so the Client VM may fail. This is not needed for this lab.

1. On the **Custom deployment** page, specify the following settings, then select **Review + Create** then **Create**.
   * Subscription: The name of the target Azure subscription where you want to provision the lab environment Azure VMs.
   * Resource group: (Create new) **hybrididentity-RG**
   * Location: The name of the Azure region that will host the lab environment Azure VMs.
   * Config Name: **TlgBaseConfig-01**
   * Domain Name: **corp.contoso.com**
   * Server OS: **2016-Datacenter**
   * Admin Username: **demouser**
   * Admin Password: **demo@pass123**
   * **It is strongly recommended that you enter a secure password that you know and can remember.**
   * Deploy Client VM: **No**
   * Client VHD URI: **leave blank**
   * VM Size: **Standard\_D2s\_v3**

**Note** - Use a similar VM size if your subscription does not support the listed size. Documentation is linked here: <https://docs.microsoft.com/en-us/azure/virtual-machines/windows/sizes>.

* + DNS Label Prefix: **Any valid, globally unique DNS name (a unique string consisting of letters, digits, and hyphens, starting with a letter and up to 47 characters long).**
  + \_artifacts Location: **Accept the default**
  + \_artifacts Location Sas Token: **leave blank**

1. Select **Review + Create**.
2. After validation has passed, select **Create**.
3. Wait for the deployment to complete. This might take about 60 minutes.

### Task 2 - Configure the lab environment Azure VMs

1. In the browser window displaying the Azure portal, navigate to the **DC1** Azure VM and connect to it via Remote Desktop. When prompted, sign in by using the following credentials:
   * Username: **demouser**
   * Password: **demo\**[**@pass123**](https://github.com/pass123)
   * **It is strongly recommended that you enter a secure password that you can remember.**
2. Within the Remote Desktop session to **DC1**, start **Windows PowerShell ISE**, add the following script to the script pane, and run it to disable Internet Explorer enhanced security configuration and User Access Control on both **DC1** and **APP1** Azure VMs:

CodeCopy

$vmNames = @('dc1','app1')

Invoke-Command -ComputerName $vmNames {Set-ItemProperty -Path "HKLM:\SOFTWARE\Microsoft\Active Setup\Installed Components\{A509B1A7-37EF-4b3f-8CFC-4F3A74704073}" -Name "IsInstalled" -Value 0}

Invoke-Command -ComputerName $vmNames {Set-ItemProperty -Path "HKLM:\SOFTWARE\Microsoft\Active Setup\Installed Components\{A509B1A8-37EF-4b3f-8CFC-4F3A74704073}" -Name "IsInstalled" -Value 0}

Invoke-Command -ComputerName $vmNames {Set-ItemProperty "HKLM:\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System" -Name "ConsentPromptBehaviorAdmin" -Value 00000000}

**Note:** To run multiple PowerShell scripts in the same file, you can highlight a specific script and select **Run Selection** next to the green play button.

1. Within the **Windows PowerShell ISE** window add the following script to the script pane, and run it to install Remote Server Administration Tools on both **DC1\* and \*\*APP1** Azure VMs:

CodeCopy

$vmNames = @('dc1','app1')

Invoke-Command -ComputerName $vmNames {Install-WindowsFeature RSAT -IncludeAllSubFeature}

1. Within the **Windows PowerShell ISE** window add the following script to the script pane, and run it to enable TLS 1.2 on both **DC1\* and \*\*APP1** Azure VMs:

CodeCopy

$vmNames = @('dc1','app1')

Invoke-Command -ComputerName $vmNames {New-Item 'HKLM:\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols\TLS 1.2\Server' -Force}

Invoke-Command -ComputerName $vmNames {New-Item 'HKLM:\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols\TLS 1.2\Client' -Force}

Invoke-Command -ComputerName $vmNames {New-ItemProperty -path 'HKLM:\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols\TLS 1.2\Client' -name 'Enabled' -value 1 –PropertyType DWORD}

Invoke-Command -ComputerName $vmNames {New-ItemProperty -path 'HKLM:\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols\TLS 1.2\Client' -name 'DisabledByDefault' -value 0 –PropertyType DWORD}

Invoke-Command -ComputerName $vmNames {New-ItemProperty -Path 'HKLM:\SOFTWARE\Microsoft\.NETFramework\v4.0.30319' -name 'SchUseStrongCrypto' -value 1 –PropertyType DWORD}

1. Within the **Windows PowerShell ISE** window add the following script to the script pane, and run it to configure Windows Integrated Authentication on the Default Web Site hosted on the **APP1** Azure VM:

CodeCopy

$vmNames = @('app1')

Invoke-Command -ComputerName $vmNames {Enable-WindowsOptionalFeature -Online -FeatureName IIS-WindowsAuthentication}

Invoke-Command -ComputerName $vmNames {Set-WebConfigurationProperty -Filter "/system.webServer/security/authentication/anonymousAuthentication" -Name Enabled -Value False -PSPath IIS:\ -Location "Default Web Site"}

Invoke-Command -ComputerName $vmNames {Set-WebConfigurationProperty -Filter "/system.webServer/security/authentication/windowsAuthentication" -Name Enabled -Value True -PSPath IIS:\ -Location "Default Web Site"}

### Task 3 - Restart the Azure VMs

1. Within the **Windows PowerShell ISE** window, from the console pane, run the following to restart **APP1**:

CodeCopy

Restart-Computer -ComputerName 'APP1'

1. Within the **Windows PowerShell ISE** window, from the console pane, run the following to restart **DC1**:

CodeCopy

Restart-Computer -ComputerName 'DC1'

### Task 5 - Configure contoso.local Active Directory

1. Connect again to the **DC1** Azure VM via Remote Desktop. When prompted, sign in by using the following credentials:
   * Username: **demouser**
   * Password: **demo\**[**@pass123**](https://github.com/pass123) - **It is strongly recommended that you enter a secure password that you can remember.**
2. Within the Remote Desktop session to **DC1**, start Internet Explorer and navigate to the link below.

CodeCopy

https://github.com/microsoft/MCW-Hybrid-identity/tree/main/Hands-on%20lab/studentfiles

1. On the **Create Users/Group for Active Directory Demo/Test Environment** page, select the **CreateDemoUsers.ps1** link, accept the licensing terms, and save the corresponding script to the local file system.
2. On the **Create Users/Group for Active Directory Demo/Test Environment** page, select the **CreateDemoUsers.csv** link (directly above the PowerShell code section) and save the corresponding csv file to the same location as the **CreateDemoUsers.ps1** file.
3. Within the Remote Desktop session to **DC1**, start File Explorer, navigate to the folder where you downloaded both files, right-Select on the file **CreateDemoUsers.ps1**, select **Properties**, in the **CreateDemoUsers.ps1 Properties** dialog box, check the **Unblock** checkbox and select **OK**.
4. Within the File Explorer window, right-Select on the file **CreateDemoUsers.ps1** again and select **Edit**.
5. In the **Administrator: Windows PowerShell ISE** window, change line **148** from:

CodeCopy

$UserCount = 1000 #Up to 2500 can be created

to

CodeCopy

$UserCount = 2500 #Up to 2500 can be created

1. In the **Windows PowerShell ISE** window, save the change and run the **CreateDemoUsers.ps1** script to create a lab environment organizational unit hierarchy and populate it with test user accounts.
2. Within the **Windows PowerShell ISE** window, add the following script to the script pane, and run it to modify settings of the AD user accounts you will use in this lab:

CodeCopy

$adUser1 = Get-ADUser -Filter {samAccountName -eq "AGAyers"}

$adUser1groups = $adUser1 | Get-ADPrincipalGroupMembership

$adUser1groups | foreach { if ($\_.name -ne 'Domain Users') {Remove-ADPrincipalGroupMembership -MemberOf $\_.name -Identity $adUser1.DistinguishedName} }

Add-ADPrincipalGroupMembership -MemberOf 'Engineering' -Identity $adUser1.DistinguishedName

Move-ADObject -Identity $adUser1.DistinguishedName -TargetPath 'OU=NJ,OU=US,OU=Users,OU=Demo Accounts,DC=corp,DC=contoso,DC=com'

Set-ADAccountPassword -Identity 'CN=Ayers\, Ann,OU=NJ,OU=US,OU=Users,OU=Demo Accounts,DC=corp,DC=contoso,DC=com' -Reset -NewPassword (ConvertTo-SecureString -AsPlainText "demo@pass123" -Force)

$adUser2 = Get-ADUser -Filter {samAccountName -eq "TFBell"}

$adUser2groups = $adUser2 | Get-ADPrincipalGroupMembership

$adUser2groups | foreach { if ($\_.name -ne 'Domain Users') {Remove-ADPrincipalGroupMembership -MemberOf $\_.name -Identity $adUser2.DistinguishedName} }

Add-ADPrincipalGroupMembership -MemberOf 'Engineering' -Identity $adUser2.DistinguishedName

Move-ADObject -Identity $adUser2.DistinguishedName -TargetPath 'OU=VT,OU=US,OU=Users,OU=Demo Accounts,DC=corp,DC=contoso,DC=com'

Set-ADAccountPassword -Identity 'CN=Bell\, Teresa,OU=VT,OU=US,OU=Users,OU=Demo Accounts,DC=corp,DC=contoso,DC=com' -Reset -NewPassword (ConvertTo-SecureString -AsPlainText "demo@pass123" -Force)

Get-ADGroup -Identity 'Domain Admins' | Add-ADGroupMember -Members 'CN=Ayers\, Ann,OU=NJ,OU=US,OU=Users,OU=Demo Accounts,DC=corp,DC=contoso,DC=com'

Get-ADGroup -Identity 'Enterprise Admins' | Add-ADGroupMember -Members 'CN=Ayers\, Ann,OU=NJ,OU=US,OU=Users,OU=Demo Accounts,DC=corp,DC=contoso,DC=com'

1. Within the **Windows PowerShell ISE** window, add the following script to the script pane, and run it to create additional organizational units named **Servers** and **Clients** and move the **APP1** computer account to the first of them:

CodeCopy

New-ADOrganizationalUnit -Name 'Servers' -Path 'OU=Demo Accounts,DC=corp,DC=contoso,DC=com'

New-ADOrganizationalUnit -Name 'Clients' -Path 'OU=Demo Accounts,DC=corp,DC=contoso,DC=com'

Move-ADObject -Identity 'CN=APP1,CN=Computers,DC=corp,DC=contoso,DC=com' -TargetPath 'OU=Servers,OU=Demo Accounts,DC=corp,DC=contoso,DC=com'

1. Sign out from **DC1**.

## Exercise 2: Integrate an Active Directory forest with an Azure Active Directory tenant

### Task 1: Create an Azure Active Directory tenant and activate an EMS E5 trial

In this task, you will create an Azure Active Directory tenant with the following settings:

* Organization name: **Contoso**
* Initial domain name: Any valid, unique domain name.
* Country or region: **United States**

1. From the lab computer, start a new Web browser window and navigate to the Azure portal at [https://portal.azure.com](https://portal.azure.com/) if you haven’t already.
2. When prompted, sign into the Azure subscription into which you deployed resources in the Before Hands-On Lab exercises.
3. On the lab computer, in the Azure portal, select **+ Create a resource**.
4. On the **New** page, in the **Search the Marketplace** text box, type **Azure Active Directory** and, in the list of results, select **Azure Active Directory**.
5. On the **Azure Active Directory** page, select **Create**.
6. On the **Create directory** page, specify the following settings and select **Create**:
   * Organization name: **Contoso**
   * Initial domain name: Any valid, unique domain name.
   * Country or region: **United States**
7. Once it’s created, navigate to your subscription page. Select **Change directory**.
8. In the **Change the directory** page on the right, select **Contoso** in the dropdown and select **Change**.
9. In the portal’s left navigation, select **Azure Active Directory**.
10. In the **Azure Active Directory** page, select **Switch tenant** then select the **Contoso** box and select **Switch**.

**Note**: It may take a few minutes for everything to display properly.

1. On the **Contoso - Overview** page, select **Licenses** under **Manage** on the left navigation.
2. On the **Contoso - Licenses**, page, select **All Products** and select **+ Try/Buy**.
3. On the **Activate** page, in the **ENTERPRISE MOBILITY + SECURITY E5** section, select **Free trial** and then select **Activate**.

**Note**: Activation typically takes about 5 minutes.

### Task 2: Create and configure Azure AD users

In this task, you will configure Azure AD user accounts in the newly created Azure AD tenant with the following settings. This will include assigning EM+S E5 licenses to the user account you are using for this lab as well as creating a new Azure AD user account with the following settings and assigning to it the Global Administrator role as well as the EM+S E5 license.

* Name: **john.doe**
* First name: **John**
* Last name: **Doe**
* Password: **Auto-generate password**
* Show Password: **Enabled**
* Groups: **0 group selected**
* Roles: **Global Administrator**
* Block sign in: **No**
* Usage location: **United States**
* Job title: **Leave blank**
* Department: **Leave blank**

1. From the lab computer, in the Azure portal, navigate back to the **Contoso - Overview** page.
2. On the **Contoso - Overview** page, select **Users** under **Manage** in the left navigation.
3. On the **Users - All users** page, select the entry representing your user account.
4. On the **Profile** page of your user account, select **Edit**.
5. In the **Settings** section, in the **Usage location** drop-down list, select the **United States** entry and select **Save**.
6. On the **Profile** page of your user account, select **Licenses** under **Manage** on the left.
7. On the **Licenses** page, select **+ Assignments**.
8. On the **Update license assignments** page, enable the **Enterprise Mobility + Security E5** checkbox, ensure that all the corresponding license options are enabled, and select **Save**.
9. On the **Users - All users** page, select **+ New user**.
10. On the **New user** page, ensure that the **Create user** option is selected, specify the following settings, and select **Create**:
    * User name: **john.doe\@your Azure AD tenant domain name** where **your Azure AD tenant domain name** is the domain name you specified when creating the Contoso Azure AD tenant.
    * Name: **john.doe**
    * First name: **John**
    * Last name: **Doe**
    * Password: **Auto-generate password**
    * Show Password: **Enabled**
    * Groups: **0 group selected**
    * Roles: **Global Administrator**
    * Block sign in: **No**
    * Usage location: **United States**
    * Job title: **Leave blank**
    * Department: **Leave blank**

**Note**: Copy the **User name** and **Password** values into Notepad. You will need them later in this lab.

1. On the **Users - All users** page, select the entry representing the newly created user account.
2. On the **john.doe - Profile** page, select **Licenses** under **Manage** on the left.
3. On the **john.doe - Licenses** page, select **+ Assignments**.
4. On the **Update license assignments** page, enable the **Enterprise Mobility + Security E5** checkbox, ensure that all the corresponding license options are enabled, and select **Save**.

### Task 3: Purchase a custom domain name

In this task, you will purchase a custom DNS domain name by leveraging the functionality described at <https://docs.microsoft.com/en-us/azure/app-service/manage-custom-dns-buy-domain>.

1. On the lab computer, in the browser displaying the Azure portal, navigate to your subscription page and select **Change directory**. In the **Change the directory** page on the right, select **Default Directory** in the dropdown and select **Change**.
2. Return to the **Azure Active Directory** overview page. Select **Switch tenant** and select the **Default Directory** associated with the Azure subscription into which you deployed resources in the Before Hands-On Lab exercises then select **Switch**.
3. In the Azure portal’s left navigation, select **+ Create a resource**.
4. On the **New** page, select **Create** within **Web App**.
5. On the **Basics** tab of the **Web App** page, specify the following settings and select **Next: Deployment** and then **Next: Monitoring**:
   * Subscription: The name of the Azure subscription into which you deployed resources in the Before Hands-On Lab exercises.
   * Resource Group: **(Create new) contosohilab-RG**.
   * Name: Any valid, globally unique name.
   * Publish: **Code**
   * Runtime stack: **.NET Core 3.1 (LTS)**
   * Operating system: **Windows**
   * Region: Any Azure region in which you can create Azure Web Apps in the target subscription.
   * App Service plan: Accept the default.
   * SKU and size: **Shared D1** (If necessary, select **Change size**, select Dev/Test, select **D1** and select **Apply**)
6. On the **Monitoring** tab of the **Web App** page, specify the following setting and select **Review + create** then **Create**:
   * Enable Application Insights: **No**
7. In the Azure portal, search for and select **App Service Domains** on the top search bar.
8. On the **App Service Domains** page, select **Create App Service Domain**.
9. On the **Create App Service Domains** page, select the **contosohilab-RG** resource group. Then in the **Search for domains…** text box, type the domain name you want to purchase and select the box next to one of the available domain names listed below the text box. Make sure you make note of the domain you choose.
10. Select **Next: Contact information**, type required information.
11. Select **Next: Advanced**, ensure that **Enable privacy protection** is set to **Disable**.
12. Select **Review + Create** then **Create**.

### Task 4: Assign a custom domain name to the Contoso Azure AD tenant

In this task, you will assign a newly purchased custom DNS domain name to the Contoso Azure AD tenant.

1. On the lab computer, in the Azure portal, select the **Directory + Subscription** icon in the toolbar of the Azure portal (to the right of the **Cloud Shell** icon) and switch to the Contoso Azure AD tenant.
2. In the Azure portal’s left navigation, select **Azure Active Directory** to navigate to the **Contoso - Overview** page.
3. On the **Contoso - Overview** page, select **Custom domain names** under **Manage** on the left.
4. On the **Contoso - Custom domain names** page, select **+ Add custom domain**.
5. On the **Add custom domain** page that appears on the right, in the **Custom domain name** text box, type the domain name you purchased in the previous task and select **Add domain**. You will be redirected to a new page displaying your custom domain name settings.
6. Identify the value of the **TXT** record on the custom domain name page.
7. On the lab computer, start another browser tab and navigate to the Azure portal.
8. In the Azure portal, select the **Directory + Subscription** icon in the toolbar of the Azure portal (to the right of the **Cloud Shell** icon) to switch to the Azure AD tenant associated with the Azure subscription into which you deployed resources in the Before Hands-On Lab exercises (the **Default Directory**).
9. In the Azure portal, select **All services** in the portal’s left navigation. In the **Search All** textbox, type **DNS zones**, and then select the **DNS zones** entry in the listing of search results.
10. On the **DNS zones** page, select the entry with the name matching the custom domain name you purchased in the previous task.
11. On the DNS zone page, select **+ Record set**.
12. On the **Add record set** page, specify the following settings and select **OK**:
    * Name: **\@**
    * Type: **TXT**
    * TTL: **1**
    * TTL unit: **Hours**
    * Value: The value of **DESTINATION OR POINTS TO ADDRESS** entry you identified on the **Custom domain name** page.
13. Switch back to the browser window displaying the custom domain name page and select **Verify**. Ensure that the verification was successful.
14. Select **Make primary** and confirm the change when prompted.

### Task 5: Configure DNS suffix in the Contoso Active Directory forest

In this task, you will configure the DNS suffix of the Contoso Active Directory forest to match the newly verified Azure AD custom domain name.

1. On the lab computer, in the Azure portal, verify that you are signed into the Azure AD tenant associated with the Azure subscription into which you deployed resources in the Before Hands-On Lab exercises (the **Default Directory**). If not, select the **Directory + Subscription** icon in the toolbar of the Azure portal (to the right of the **Cloud Shell** icon) to switch to that Azure AD tenant.
2. In the Azure portal, navigate to the page of the **DC1** virtual machine.
3. On the **DC1** virtual machine page, connect to **DC1** via Remote Desktop. When prompted to sign in, use the **demouser** name and the **demo\**[**@pass123**](https://github.com/pass123) password.
4. Within the Remote Desktop session to **DC1**, on the **Server Manager** window, start the **Active Directory Domains and Trusts** console under **Tools**.
5. In the **Active Directory Domains and Trusts** console, right-Select **Active Directory Domains and Trusts [DC1.corp.contoso.com]** on the left and select **Properties**.
6. On the **UPN Suffixes** tab of the **Active Directory Domains and Trusts [DC1.corp.contoso.com]** window, in the **Alternative UPN suffixes** textbox, type the name of the custom domain you verified in the previous task, select **Add**, and then select **OK**.
7. Within the Remote Desktop session to **DC1**, on the **Server Manager** window, start the **Active Directory Users and Computers** console under **Tools**.
8. In the **Active Directory Users and Computers** console, expand **corp.contoso.com** on the left and examine the organizational unit hierarchy of the domain and the group membership of the domain groups.
9. Within the Remote Desktop session to **DC1**, start Windows PowerShell ISE and, on the Script pane, run the following to replace the UPN suffix of all users who are members of the **Engineering** group with the one matching the custom verified domain name of the Contoso Azure AD tenant (replace the placeholder <custom\_domain\_name> with the actual name of the custom verified domain name you assigned to the Contoso Azure AD tenant).

CodeCopy

$domainName = '<custom\_domain\_name>'

$users = Get-ADGroupMember -Identity 'Engineering' -Recursive | Where-Object {$\_.objectClass -eq 'user'}

foreach ($user in $users) {

$user = Get-ADUser -Identity $User.SamAccountName

$userName = $user.UserPrincipalName.Split('@')[0]

$upn = $userName + "@" + $domainName

$user | Set-ADUser -UserPrincipalName $upn

}

### Task 6: Install Azure AD Connect

In this task, you will install Azure AD Connect.

1. Within the Remote Desktop session to **DC1**, in Server Manager, select **Local Server**, and ensure that **IE Enhanced Security Configuration** is disabled. If not, then select the **On** link next to **IE Enhanced Security Configuration**, set the **Administrators** settings to **Off**, and select **OK**.
2. Within the Remote Desktop session to **DC1**, open the **Windows PowerShell ISE** window and run this command to install the Chrome browser.

CodeCopy

$LocalTempDir = $env:TEMP; $ChromeInstaller = "ChromeInstaller.exe"; (new-object System.Net.WebClient).DownloadFile('http://dl.google.com/chrome/install/375.126/chrome\_installer.exe', "$LocalTempDir\$ChromeInstaller"); & "$LocalTempDir\$ChromeInstaller" /silent /install; $Process2Monitor = "ChromeInstaller"; Do { $ProcessesFound = Get-Process | ?{$Process2Monitor -contains $\_.Name} | Select-Object -ExpandProperty Name; If ($ProcessesFound) { "Still running: $($ProcessesFound -join ', ')" | Write-Host; Start-Sleep -Seconds 2 } else { rm "$LocalTempDir\$ChromeInstaller" -ErrorAction SilentlyContinue -Verbose } } Until (!$ProcessesFound)

1. Within the Remote Desktop session to **DC1**, start the Chrome browser and navigate to the Azure portal at [https://portal.azure.com](https://portal.azure.com/).
2. When prompted to sign in, enter the credentials of the **john.doe** Azure AD user account, which you copied into Notepad earlier in this exercise.
3. When prompted, change the password for the **john.doe** user account.

**Note**: If you receive the message **We’ve seen that password too many times before. Choose something harder to guess**, you’ll need to modify the password until it is unique enough to be accepted.

1. If prompted whether to **Stay signed in?”** select **No**. You will be redirected to the Azure portal interface.
2. If presented with the **Welcome to Microsoft Azure** dialog box, select **Maybe later**.
3. In the Azure portal, select **Azure Active Directory** on the portal’s left navigation to navigate to the **Contoso - Overview** page.
4. On the **Contoso - Overview** page, select **Azure AD Connect** under **Manage** on the left.
5. On the **Azure AD Connect** page, select the **Download Azure AD Connect** link.
6. On the **Microsoft Azure Active Directory Connect** web page of the Microsoft Downloads site, select **Download**.
7. When prompted whether to run or save **AzureADConnect.msi**, select **Run**. This will download the file and automatically start the **Microsoft Azure Active Directory Connect** wizard.
8. On the **Welcome to Azure AD Connect** page, check the **I agree to the license terms and privacy notice** box and select **Continue**.
9. On the **Express Settings** page, select the **Customize** button.
10. On the **Install required components** page, leave all optional configuration options deselected and select **Install**.
11. On the **User sign-in** page, select the **Pass-through authentication** option and the **Enable single sign-on** checkboxes, and select **Next**.
12. On the **Connect to Azure AD** page, sign in by using the credentials of the **john.doe** account and select **Next**.
13. On the **Connect your directories** page, ensure that the **corp.contoso.com** entry appears in the **FOREST** drop-down list and select **Add Directory**. In the **AD forest account**, ensure that the **Create new AD account** option is selected, in the **ENTERPRISE ADMIN USERNAME** textbox, type **CORP.CONTOSO.COM\demouser**, in the **PASSWORD** textbox, type **demo\**[**@pass123**](https://github.com/pass123), and select **OK**.
14. Back on the **Connect your directories** page, select **Next**.
15. On the **Azure AD sign-in configuration** page, ensure that your custom domain name is listed as the verified **Active Directory UPN Suffix**, and that the **userPrincipalName** entry appears in the **USER PRINCIPAL NAME** drop-down list. Note the warning stating **Users will not be able to sign into Azure AD with on-premises credentials if the UPN suffix does not match a verified domain name**. Check the **Continue without matching all UPN suffixes to verified domain** box and select **Next**.

**Note**: This is expected, since some users are still configured with the **contoso.local** UPN suffix, which is not routable and cannot be configured as a verified custom domain name of an Azure AD tenant.

1. On the **Domain and OU filtering** page, ensure that only the **DemoAccounts** OU and all its children OUs are selected and select **Next**.
2. On the **Uniquely identifying your users** page, accept the default settings and select **Next**.
3. On the **Filter users and devices** page, accept the default settings and select **Next**.
4. On the **Optional features** page, accept the default settings and select **Next**.
5. On the **Enable single sign-on** page, select **Enter credentials**, in the **Forest credentials** dialog box, sign in with the **CORP\demouser** username and **demo\**[**@pass123**](https://github.com/pass123) password, and select **Next**.
6. On the **Ready to configure** page, ensure that the **Start the synchronization process when configuration completes** checkbox is **NOT** selected and select **Install**.

**Note**: You will configure attribute-level filtering before enabling the synchronization process.

**Note**: Installation should take about 2 minutes.

1. On the **Configuration complete** page, select **Exit**.

### Task 7: Enable Active Directory Recycle Bin

In this task, you will enable Recycle Bin in the Contoso Active Directory domain.

1. Within the Remote Desktop session to **DC1**, on the Tools menu in the Server Manager console, start **Active Directory Administrative Center**.
2. In the **Active Directory Administrative Center** console, right-Select **corp (local)** on the left and select **Enable Recycle Bin**. When prompted to confirm, select **OK**.
3. When prompted to refresh AD Administrative Center, select **OK**.

**Note**: For information regarding benefits of the Recycle Bin in hybrid scenarios, refer to <https://docs.microsoft.com/en-us/azure/active-directory/hybrid/how-to-connect-sync-recycle-bin>

### Task 8: Configure Azure AD Connect attribute-level filtering

In this task, you will configure Azure AD Connect attribute level filtering that will limit synchronization of user accounts to those with the UPN suffix matching the custom domain name of the target Azure AD tenant.

**Note**: The positive filtering option requires at least two sync rules. One of them determines the correct scope of objects to synchronize. The second catch-all sync rule filters out all objects that have not yet been identified as an object that should be synchronized.

1. Within the Remote Desktop session to **DC1**, start **Synchronization Rules Editor** under **Azure AD Connect** in the Start menu.
2. In the Synchronization Rules Editor window, on the **View and manage your synchronization rules** page, ensure that **Inbound** appears in the **Direction** drop-down list and select **Add new rule**. This will launch the **Create inbound synchronization rule** wizard.
3. On the **Create inbound synchronization rule - Description** page, specify the following settings and select **Next**:
   * Name: **Custom In from AD - UPN Filter**
   * Description: **Custom Inbound Rule - includes users with UPN set to match the Azure AD custom domain**
   * Connected System: **corp.contoso.com**
   * Connected System Object Type: **user**
   * Metaverse Object Type: **person**
   * Link Type: **join**
   * Precedence: **50**
   * Tag: **Leave empty**
   * Enable Password Sync: **Leave empty**
   * Disabled: **Leave empty**
4. On the **Create inbound scoping filter** page, select **Add Group**, select **Add clause** specify the following, and select **Next**:
   * Attribute: **userPrincipalName**
   * Operator: **ENDSWITH**
   * Value: **\@<your custom domain name>**
5. On the **Join Rules** page, select **Next**.
6. On the **Transformations** page, select **Add transformation** specify the following and select **Add**:
   * FlowType: **Constant**
   * Target Attribute: **cloudFiltered**
   * Source: **False**
7. When presented with a **Warning** dialog box displaying that message stating that **A full import and full synchronization will be run on ‘corp.contoso.com’ during your next synchronization cycle**, select **OK**.

**Note**: This should bring you back to the View and manage your synchronization rules interface, with the new rule listed at the top of the rule list.

1. Back in the **Synchronization Rules Editor** window, on the **View and manage your synchronization rules** page, ensure that **Inbound** appears in the **Direction** drop-down list and select **Add new rule** again. This will launch the **Create inbound synchronization rule** wizard.
2. On the **Description** page, specify the following settings and select **Next**:
   * Name: **Custom In from AD - Catch-all Filter**
   * Description: **Custom Inbound Rule - excludes all users with UPN not set to match the Azure AD custom domain**
   * Connected System: **corp.contoso.com**
   * Connected System Object Type: **user**
   * Metaverse Object Type: **person**
   * Link Type: **join**
   * Precedence: **51**
   * Tag: **Leave empty**
   * Enable Password Sync: **Leave empty**
   * Disabled: **Leave empty**
3. On the **Scoping filer** page, select **Next**.
4. On the **Join Rules** page, select **Next**.
5. On the **Transformations** page, select **Add transformation** specify the following and select **Add**:
   * FlowType: **Constant**
   * Target Attribute: **cloudFiltered**
   * Source: **True**
6. When presented with a **Warning** dialog box displaying a message stating that **A full import and full synchronization will be run on ‘corp.contoso.com’ during your next synchronization cycle**, select **OK**.

**Note**: This should bring you back to the **View and manage your synchronization rules** interface, with the new rules listed at the top of the rule list.

### Task 9: Initiate and verify directory synchronization

1. Within the Remote Desktop session to **DC1**, double-Select the **Azure AD Connect** desktop shortcut.
2. On the **Welcome to Azure AD Connect** page, select **Configure**.
3. On the **Additional tasks** page, select **Customize synchronization options** and select **Next**.
4. On the **Connect to Azure AD** page, sign in by using the credentials of the **john.doe** account and select **Next**.
5. On the **Connect your directories** page, select **Next**.
6. On the **Domain and OU filtering** page, select **Next**.
7. On the **Optional features** page, accept the default settings and select **Next**.
8. On the **Enable single sign-on** page, select **Next**.
9. On the **Ready to configure** page, select the **Start the synchronization process when configuration completes** checkbox and select **Configure**.
10. On the **Configuration complete** page, select **Exit**.
11. Within the Remote Desktop session to **DC1**, in the Edge browser window displaying the Azure portal, navigate to the **Users - All users** page of the Contoso Azure AD tenant.
12. On the **Users - All users** page, note that the list of user objects includes all user accounts with the UPN suffix matching the custom domain name of the Azure AD tenant. You may need to refresh the page or wait a few minutes to see the change.
13. In the Azure portal, navigate to the **Groups - All groups** page of the Contoso Azure AD tenant and note that all the corp.contoso.com domain groups have been synchronized as well.
14. In the Azure portal, navigate to the **Contoso - Azure AD Connect** page and select **Azure AD Connect** on the left. Verify that the following settings are set:
    * Azure AD Connect Sync Status: **Enabled**
    * Last Sync: **This should be a timestamp of some sort**.
    * Password Hash Sync: **Disabled**
    * Federation: **Disabled**
    * Seamless single sign-on: **Enabled for 1 domain**
    * Pass-through authentication: **Enabled with 1 agent**

**Note**: In a production environment, you would install additional agents for redundancy. For more information, refer to <https://docs.microsoft.com/en-us/azure/active-directory/hybrid/how-to-connect-pta-quick-start>.

### Task 10: Configure Hybrid Azure AD join

In this task, you will configure Azure AD Connect device synchronization options.

1. Within the Remote Desktop session to **DC1**, double-Select the **Azure AD Connect** desktop shortcut.
2. On the **Welcome to Azure AD Connect** page, select **Configure**.
3. On the **Additional tasks** page, select **Configure device options** and select **Next**.
4. On the **Overview** page, review the information regarding **Hybrid Azure AD join** and **Device writeback**, and select **Next**.
5. On the **Connect to Azure AD** page, sign in by using the credentials of the **john.doe** account and select **Next**.
6. On the **Device options** page, ensure that the **Configure Hybrid Azure AD join** option is selected and select **Next**.
7. On the **Device operating system** page, select the **Windows 10 or later domain-joined devices** and **Supported Windows down-level domain-joined devices** checkboxes, and select **Next**.

**Note**: Windows down-level devices are supported only if you are using Seamless SSO for managed domains or a federation service such as AD FS for federated domains.

1. On the **SCP configuration** page, check the **corp.contoso.com** Active Directory forest box, select the **Azure Active Directory** entry in the **Authentication Service** dropdown list, and select **Add**.
2. When prompted for Enterprise Admin Credentials for corp.contoso.com, in the **Windows Security** dialog box, sign in with the **CORP\demouser** user name and **demo\**[**@pass123**](https://github.com/pass123) password.
3. Back on the **SCP configuration** page, select **Next**.
4. On the **Ready to configure** page, select **Configure**.
5. On the **Configuration complete** page verify that the task completed successfully and select **Exit**.

**Note**: For more information regarding configuring hybrid Azure Active Directory join for managed domains, refer to <https://docs.microsoft.com/en-us/azure/active-directory/devices/hybrid-azuread-join-managed-domains#configure-hybrid-azure-ad-join>.

### Task 11: Perform Hybrid Azure AD join

1. On the lab computer, in the Azure portal, verify that you are signed into the Azure AD tenant associated with the Azure subscription into which you deployed resources in the Before Hands-On Lab exercises (the **Default directory**). If not, select the **Directory + Subscription** icon in the toolbar of the Azure portal (to the right of the **Cloud Shell** icon) to switch to that Azure AD tenant.
2. In the Azure portal, navigate to the page of the **APP1** virtual machine.
3. On the **APP1** virtual machine page, connect to **APP1** via Remote Desktop. When prompted to sign in, use the \*\*AGAyers\@\*\* user name with the \*\*demo@pass123\*\* password (where \*\*\*\* placeholder represents the custom DNS domain name you assigned to the Contoso Azure AD tenant earlier in this exercise.
4. Within the Remote Desktop session to **APP1**, on the **Server Manager** window, start **Task Scheduler** under **Tools**.
5. In the **Task Scheduler** console, navigate to **Task Scheduler Library** > **Microsoft** > **Windows** > **Workplace Join**. From there, enable then run the **Automatic-Device-Join** task.
6. Switch to the Remote Desktop session to **DC1** and, from the console pane of the Windows PowerShell ISE window, start Azure AD Connect delta synchronization by running the following:

CodeCopy

Import-Module -Name 'C:\Program Files\Microsoft Azure AD Sync\Bin\ADSync\ADSync.psd1'

Start-ADSyncSyncCycle -PolicyType Delta

1. Switch back to the Remote Desktop session to **APP1** and start a **Command Prompt**.
2. From the Command Prompt window, check the Azure AD registration status of APP1 by running the following:

CodeCopy

dsregcmd /status

1. Verify that the output of the command resembles the following:

CodeCopy

+----------------------------------------------------------------------+

| Device State |

+----------------------------------------------------------------------+

AzureAdJoined : YES

EnterpriseJoined : NO

DeviceId : 61eea2b8-efbe-43d9-b267-126433c8ee34

Thumbprint : BBAAA0FB4A55E880388851BED955A2669A961A96

KeyContainerId : 2eb75eb8-0a1d-437b-99d9-9dd161ca0d90

KeyProvider : Microsoft Software Key Storage Provider

TpmProtected : NO

KeySignTest: : PASSED

Idp : login.windows.net

TenantId : xxxxxxx-xxxx-xxx-xxxx-xxxxxxxxxx

TenantName : xxxxxxx-xxxx-xxx-xxxx-xxxxxxxxxx

AuthCodeUrl : https://login.microsoftonline.com/xxxxxxx-xxxx-xxx-xxxx-xxxxxxxxxx/oauth2/authorize

AccessTokenUrl : https://login.microsoftonline.com/xxxxxxx-xxxx-xxx-xxxx-xxxxxxxxxx/oauth2/token

MdmUrl :

MdmTouUrl :

MdmComplianceUrl :

SettingsUrl :

JoinSrvVersion : 1.0

JoinSrvUrl : https://enterpriseregistration.windows.net/EnrollmentServer/device/

JoinSrvId : urn:ms-drs:enterpriseregistration.windows.net

KeySrvVersion : 1.0

KeySrvUrl : https://enterpriseregistration.windows.net/EnrollmentServer/key/

KeySrvId : urn:ms-drs:enterpriseregistration.windows.net

DomainJoined : YES

DomainName : CORP

+----------------------------------------------------------------------+

| User State |

+----------------------------------------------------------------------+

NgcSet : NO

WorkplaceJoined : NO

WamDefaultSet : NO

AzureAdPrt : NO

+----------------------------------------------------------------------+

| Ngc Prerequisite Check |

+----------------------------------------------------------------------+

IsUserAzureAD : NO

PolicyEnabled : NO

DeviceEligible : YES

SessionIsNotRemote : NO

X509CertRequired : NO

PreReqResult : WillNotProvision

1. Switch back to the Remote Desktop session to **DC1**, in the Edge browser window displaying the Azure portal, navigate to the **Devices - All devices** page of the Contoso Azure AD tenant and verify that there is an entry representing the APP1 server, with the **Join Type** set to **Hybrid Azure AD joined**.