

Cross-tenant mailbox migration (preview)

Cross-tenant mailbox migration

Commonly, during mergers or divestitures, you need the ability to move your user's Exchange Online mailbox into a new tenant. Cross-tenant mailbox migration allows tenant administrators to use well known interfaces like Remote PowerShell and MRS to transition users to their new organization.

Administrators can use the `New-MigrationBatch` cmdlet, available through the Move Mailboxes management role, to execute cross-tenant moves.

Users migrating must be present in the target tenant Exchange Online system as MailUsers, marked with specific attributes to enable the cross-tenant moves. The system will fail moves for users that are not properly set up in the target tenant.

When the moves are complete, the source user mailbox is converted to a MailUser and the `targetAddress` (shown as `ExternalEmailAddress` in Exchange) is stamped with the routing address to the destination tenant. This process leaves the legacy MailUser in the source tenant and allows for a period of co-existence and mail routing. When business processes allow, the source tenant may remove the source MailUser or convert them to a mail contact.

Cross-tenant Exchange mailbox migrations are supported for tenants in hybrid or cloud only, or any combination of the two.

This article describes the process for cross-tenant mailbox moves and provides guidance on how to prepare source and target tenants for the Exchange Online mailbox content moves.

[!NOTE] We've recently updated our setup steps to enable cross-tenant mailbox migration to no longer require Azure Key Vault! If this is the first time you are onboarding to this preview, no action is required and you can go ahead and follow the steps detailed in this document. If you have started configuring your tenants using the previous AKV method, we highly recommend you stop or remove that configuration to begin using this new method. If you have mailbox migrations in progress with the

previous AKV method, then please wait until your existing migrations are complete and follow the steps below to enable the new simplified method. Azure Key Vault required setup steps are archived but can be found [here](#), for reference.

Preparing source and target tenants

Prerequisites for source and target tenants

Before starting, be sure you have the necessary permissions to configure the Move Mailbox application in Azure, EXO Migration Endpoint, and the EXO Organization Relationship.

Additionally, at least one mail-enabled security group in the source tenant is required. These groups are used to scope the list of mailboxes that can move from source (or sometimes referred to as resource) tenant to the target tenant. This allows the source tenant admin to restrict or scope the specific set of mailboxes that need to be moved, preventing unintended users from being migrated. Nested groups are not supported.

You will also need to communicate with your trusted partner company (with whom you will be moving mailboxes) to obtain their Microsoft 365 tenant ID. This tenant ID is used in the Organization Relationship DomainName field.

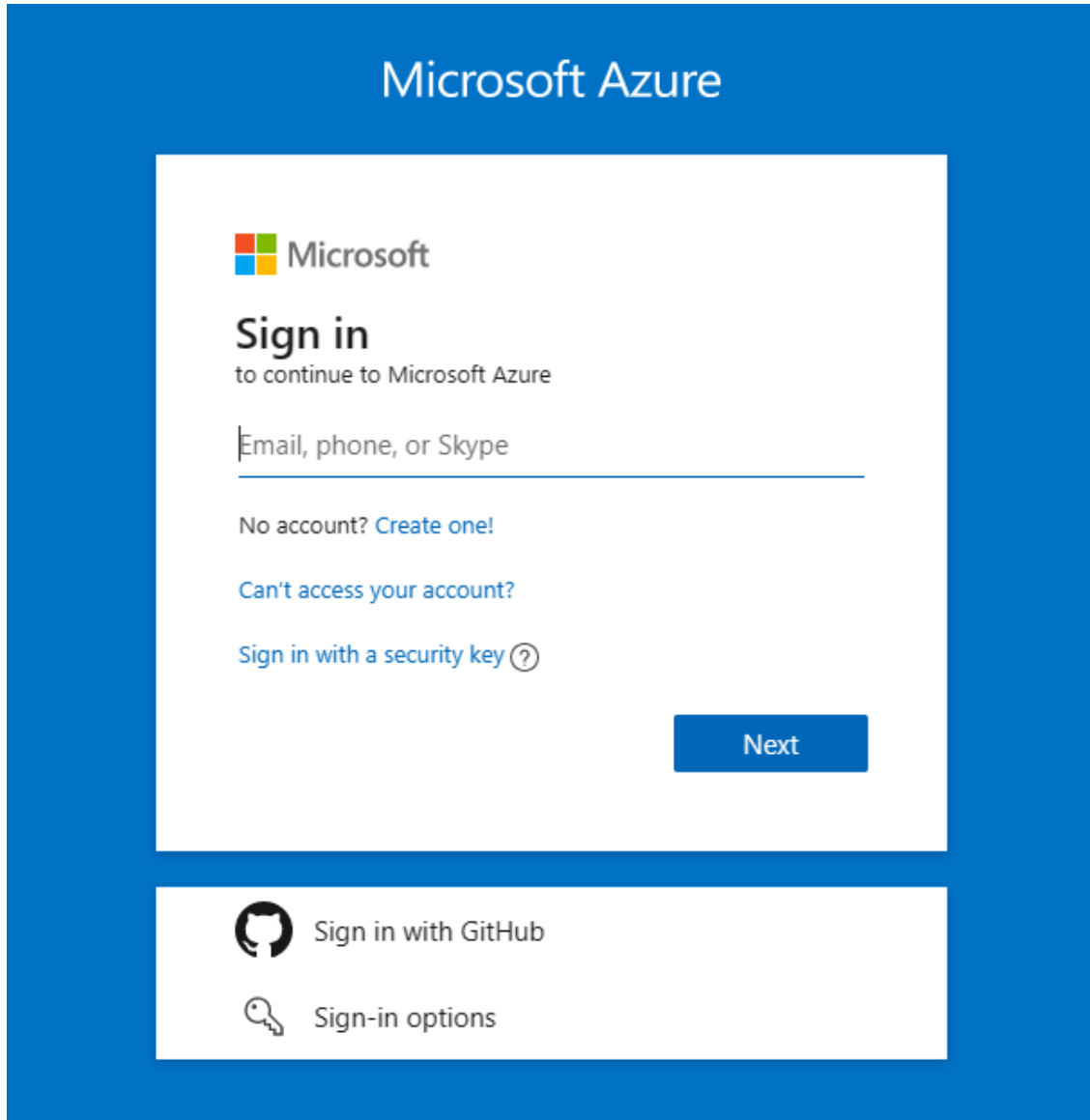
To obtain the tenant ID of a subscription, sign in to the [Microsoft 365 admin center](#) and go to https://aad.portal.azure.com/#blade/Microsoft_AAD_IAM/ActiveDirectoryMenuBlade/Properties. Click the copy icon for the Tenant ID property to copy it to the clipboard.

Configuration steps to enable your tenants for cross-tenant mailbox migrations

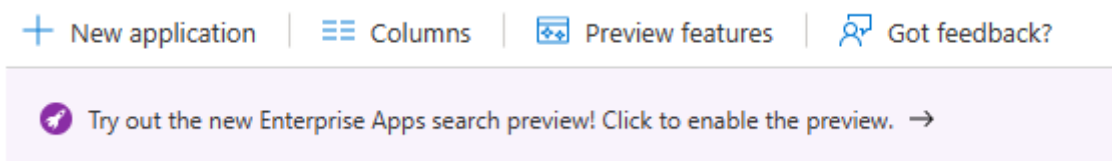
[!NOTE] You must configure the target (destination) first. To complete these steps, you are not required to have or know the tenant admin credentials for both source or target tenant. Steps can be performed individually for each tenant by different administrators.

Prepare the target (destination) tenant by creating the migration application and secret

1. Log into your Azure AD portal (<https://portal.azure.com>) with your target tenant admin credentials



2. Under Azures services, click on Azure Active Directory.
3. On the left nav, select Enterprise applications.
4. Select New application



5. Select Create your own application

Browse Azure AD Gallery ...

[+ Create your own application](#) | [Request new gallery app](#) | [Got feedback?](#)

6. Enter a name for your application (can be specific to your organization's naming conventions), and select the Register an application to integrate with Azure AD, then Create.

Create your own application



[Got feedback?](#)

What's the name of your app?

✓

What are you looking to do with your application?

- Configure Application Proxy for secure remote access to an on-premises application
- Register an application to integrate with Azure AD (App you're developing)
- Integrate any other application you don't find in the gallery (Non-gallery)

7. On the Register an application page, under Supported account types, Select Accounts in any organizational directly (Any Azure AD directory – Multitenant). Then under Redirect URI (optional) select Web and enter

<https://office.com>. Last, select Register.

[Home](#) > [Enterprise applications](#) > [Browse Azure AD Gallery](#) >

Register an application

* Name

The user-facing display name for this application (this can be changed later).

MyCrossTenantMailboxMigrationApp ✓

Supported account types

Who can use this application or access this API?

- Accounts in this organizational directory only (test_test_msftofetesttenant-AdvancedEncryption only - Single tenant)
- Accounts in any organizational directory (Any Azure AD directory - Multitenant)
- Accounts in any organizational directory (Any Azure AD directory - Multitenant) and personal Microsoft accounts (e.g. Skype, Xbox)
- Personal Microsoft accounts only

[Help me choose...](#)

Redirect URI (optional)

We'll return the authentication response to this URI after successfully authenticating the user. Providing this now is optional and it can be changed later, but a value is required for most authentication scenarios.

Web ✓

8. On the top right corner of the page, you will see a pop up that states the app was successfully created.
9. Go back to Home, Azure Active Directory and click on App registrations.
10. Under Owned applications, find the app you just created by name, click on it.
11. Now, on the left nav bar, click on API permissions to view permissions assigned to your app.
12. By default User.Read permissions are assigned to the app you just created, but we do not require them for mailbox migrations, you can remove that permission.

Configured permissions

Applications are authorized to call APIs when they are granted permissions by users/admins as part of the consent process. The list of configured permissions should include all the permissions the application needs. [Learn more about permissions and consent](#)

+ Add a permission ✓ Grant admin consent for test_test_msftofetesttenant-AdvancedEncryption

API / Permissions name	Type	Description	Admin consent requ...	Status	
Microsoft Graph (1)					
User.Read	Delegated	Sign in and read user profile	No		Remove permission

13. Now we need to add a permission for mailbox migration, select Add a permission
14. In the Request API permissions windows, select APIs my organization uses, and search for office 365 exchange online, select it.

Request API permissions

Select an API

Microsoft APIs APIs my organization uses My APIs

Apps in your directory that expose APIs are shown below

Name


Office 365 Exchange Online

15. Next, select Application permissions

16. Then, under Select permissions, expand Mailbox and check Mailbox.Migration, and Add permissions at the bottom on the screen.

Request API permissions ×

[< All APIs](#)

 Office 365 Exchange Online
https://ps.outlook.com

What type of permissions does your application require?

Delegated permissions
Your application needs to access the API as the signed-in user.

Application permissions
Your application runs as a background service or daemon without a signed-in user.

Select permissions [expand all](#)

🔍 Start typing a permission to filter these results

Permission	Admin consent required
> Other permissions	
> Calendars	
> Contacts	
> Exchange	
▼ Mailbox (1)	
<input checked="" type="checkbox"/> Mailbox.Migration ⓘ Move mailboxes between organizations	Yes
> MailboxSettings	
> Mail	

Add permissionsDiscard

17. Now select Certificates & secrets on the left nav for your application.

18. Under Client secrets, select New client secret.

Client secrets

A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as application password.

[+ New client secret](#)

Description	Expires	Value	Secret ID
No client secrets have been created for this application.			

19. In the Add a client secret window, enter a description, and configure your desired expiration settings.

[!NOTE] This is the password that will be used when creating your migration endpoint. It is extremely important that you copy this password to your clipboard and or copy this password to secure/secret password safe location. This is the only time you will be able to see this password! If you do somehow lose it or need to reset it, you can log back into our Azure portal, go to App registrations, find your migration app, select Secrets & certificates and create a new secret for your app.

1. Now that you have successfully created the migration application and secret, you will need to consent to the application. To consent to the application, go back to the Azure Active Directory landing page, click on Enterprise applications in the left navigation, find your migration app you just created, select it, and select Permissions on the left navigation.
2. Click on the Grant admin consent for [your tenant] button.
3. A new browser window will open and select Accept.
4. You can go back to your portal window and select Refresh to confirm your acceptance.
5. Formulate the URL to send to your trusted partner (source tenant admin) so they can also accept the application to enable mailbox migration. Here is an example of the URL to provide to them you will need the application ID of the app you just created:

[https://login.microsoftonline.com/sourcetenant.onmicrosoft.com/adminconsent?client_id=application_id_of_the_app_you_just_created]&redirect_uri=<https://office.com>

Prepare the target tenant by creating the Exchange Online migration endpoint and organization relationship

1. Create a Remote PowerShell connection to the target Exchange Online tenant.
2. Create a new migration endpoint for cross-tenant mailbox moves

1. [!NOTE] You will need the application ID of the mailbox migration app you just created and the password (the secret) you configured during this process.

powershell

```
$AppId = "[guid copied from the migrations app]"

$Credential = New-Object -TypeName
System.Management.Automation.PSCredential -ArgumentList $AppId,
(ConvertTo-SecureString -String "[this is your secret password you
saved in the previous steps]" -AsPlainText -Force)

New-MigrationEndpoint -RemoteServer outlook.office.com -RemoteTenant
"sourcetenant.onmicrosoft.com" -Credentials $Credential -
ExchangeRemoteMove:$true -Name "[the name of your migration endpoint]"
-ApplicationId $AppId
```

[!NOTE] Depending on the Microsoft 365 Cloud Instance you use your endpoint may be different. Please refer to the [Microsoft 365 endpoints](#) page and select the correct instance for your tenant and review the Exchange Online Optimize Required address and replace as appropriate.

1. Create new or edit your existing organization relationship object to your source tenant.

powershell

```
$sourceTenantId="[tenant id of your trusted partner, where the source
mailboxes are]"
$orgrels=Get-OrganizationRelationship
$existingOrgRel = $orgrels | ?{$_ .DomainNames -like $sourceTenantId}
If ($null -ne $existingOrgRel)
{
    Set-OrganizationRelationship $existingOrgRel.Name -Enabled:$true -
MailboxMoveEnabled:$true -MailboxMoveCapability Inbound
}
```

```

If ($null -eq $existingOrgRel)
{
    New-OrganizationRelationship "[name of the new organization
relationship]" -Enabled:$true -MailboxMoveEnabled:$true -
MailboxMoveCapability Inbound -DomainNames $sourceTenantId
}

```

Prepare the source (current mailbox location) tenant by accepting the migration application and configuring the organization relationship

1. From a browser go to the URL link provided by your trusted partner to consent to the mailbox migration application. The URL will look like this:
[\[https://login.microsoftonline.com/sourcetenant.onmicrosoft.com/adminconsent?client_id=application_id_of_the_app_you_just_created\]&redirect_uri=https://office.com](https://login.microsoftonline.com/sourcetenant.onmicrosoft.com/adminconsent?client_id=application_id_of_the_app_you_just_created&redirect_uri=https://office.com)
2. Accept the application when the pop up appears. You can also log into your Azure Active Directory portal and find the application under Enterprise applications.
3. Create new or edit your existing organization relationship object to your target (destination) tenant from an Exchange Online Remote PowerShell window.

powershell

```

$targetTenantId="[tenant id of your trusted partner, where the
mailboxes are being moved to]"
$appId="[application id of the mailbox migration app you consented
to]"
$scope="[name of the mail enabled security group that contains the
list of users who are allowed to migrate]"
$orgrels=Get-OrganizationRelationship
$existingOrgRel = $orgrels | ?{$_ .DomainNames -like $targetTenantId}
If ($null -ne $existingOrgRel)
{

```

```
Set-OrganizationRelationship $existingOrgRel.Name -Enabled:$true -
MailboxMoveEnabled:$true -MailboxMoveCapability RemoteOutbound -
OAuthApplicationId $appId -MailboxMovePublishedScopes $scope
}
If ($null -eq $existingOrgRel)
{
    New-OrganizationRelationship "[name of your organization
relationship]" -Enabled:$true -MailboxMoveEnabled:$true -
MailboxMoveCapability RemoteOutbound -DomainNames $targetTenantId -
OAuthApplicationId $appId -MailboxMovePublishedScopes $scope
}
```

How do I know this worked?

You can verify cross-tenant mailbox migration configuration by running `Test-MigrationServerAvailability` cmdlet against the cross-tenant migration endpoint that you created on your target tenant.

```
[!NOTE] Test-MigrationServerAvailability -Endpoint "[the name of your
cross-tenant migration endpoint]" -TestMailbox "[email address of a
source mailbox that is part of your migration scope]"
```

Move mailboxes back to the original source

If a mailbox move back to the original source tenant is required, the same set of steps and scripts will need to be run in both new source and new target tenants. The existing Organization Relationship object will be updated or appended, not recreated.

Prepare target user objects for migration

Users migrating must be present in the target tenant and Exchange Online system (as MailUsers) marked with specific attributes to enable the cross-tenant moves. The system will fail moves for users that are not properly set up in the target tenant. The following section details the MailUser object requirements for the target tenant.

Prerequisites for target user objects

You must ensure the following objects and attributes are set in the target organization.

1. For any mailbox moving from a source organization, you must provision a MailUser object in the Target organization:
 1. The Target MailUser must have these attributes from the source mailbox or assigned with the new User object:
 1. ExchangeGUID (direct flow from source to target) – The mailbox GUID must match. The move process will not proceed if this is not present on target object.
 2. ArchiveGUID (direct flow from source to target) – The archive GUID must match. The move process will not proceed if this is not present on the target object. (This is only required if the source mailbox is Archive enabled).
 3. LegacyExchangeDN (flow as proxyAddress, “x500:<LegacyExchangeDN>”) – The LegacyExchangeDN must be present on target MailUser as x500: proxyAddress. In addition, you also need to copy all x500 addresses from the source mailbox to the target mail user. The move processes will not proceed if these are not present on the target object.
 4. UserPrincipalName – UPN will align to the user’s NEW identity or target company (for example, user@northwindtraders.onmicrosoft.com).
 5. Primary SMTPAddress – Primary SMTP address will align to the user’s NEW company (for example, user@northwind.com).
 6. TargetAddress/ExternalEmailAddress – MailUser will reference the user’s current mailbox hosted in source tenant (for example user@contoso.onmicrosoft.com). When assigning this value, verify that you have/are also assigning PrimarySMTPAddress or this value will set the PrimarySMTPAddress which will cause move failures.

7. You cannot add legacy smtp proxy addresses from source mailbox to target MailUser. For example, you cannot maintain contoso.com on the MEU in fabrikam.onmicrosoft.com tenant objects). Domains are associated with one Azure AD or Exchange Online tenant only.

1. Example **target** MailUser object:

Attribute	Value
Alias	LaraN
RecipientType	MailUser
RecipientTypeDetails	MailUser
UserPrincipalName	LaraN@northwintraders.onmicrosoft.com
PrimarySmtpAddress	Lara.Newton@northwind.com
ExternalEmailAddress	SMTP:LaraN@contoso.onmicrosoft.com
ExchangeGuid	1ec059c7-8396-4d0b-af4e-d6bd4c12a8d8
LegacyExchangeDN	/o=First Organization/ou=Exchange Administrative Group (FYDIBOHF23SPDLT)/cn=Recipients/cn=74e5385fce4b46d19006876949855035Lara
EmailAddresses	x500:/o=First Organization/ou=Exchange Administrative Group (FYDIBOHF23SPDLT)/cn=Recipients/cn=d11ec1a2cacd4f81858c8190 7273f1f9-Lara smtp:LaraN@northwindtraders.onmicrosoft.com

	SMTP:Lara.Newton@northwind.com

2. Example **source** Mailbox object:

Attribute	Value
Alias	LaraN
RecipientType	UserMailbox
RecipientTypeDetails	UserMailbox
UserPrincipalName	LaraN@contoso.onmicrosoft.com
PrimarySmtpAddress	Lara.Newton@contoso.com
ExchangeGuid	1ec059c7-8396-4d0b-af4e-d6bd4c12a8d8
LegacyExchangeDN	/o=First Organization/ou=Exchange Administrative Group (FYDIBOHF23SPDLT)/cn=Recipients/cn=d11ec1a2cacd4f81858c81907273f1f9Lara
EmailAddresses	smtp:LaraN@contoso.onmicrosoft.com
	SMTP:Lara.Newton@contoso.com

2. Additional attributes may be included in Exchange hybrid write back already. If not, they should be included.
3. msExchBlockedSendersHash – Writes back online safe and blocked sender data from clients to on-premises Active Directory.
4. msExchSafeRecipientsHash – Writes back online safe and blocked sender data from clients to on-premises Active Directory.

5. msExchSafeSendersHash – Writes back online safe and blocked sender data from clients to on-premises Active Directory.
1. If the source mailbox is on LitigationHold and the source mailbox Recoverable Items size is greater than our database default (30 GB), moves will not proceed since the target quota is less than the source mailbox size. You can update the target MailUser object to transition the ELC mailbox flags from the source environment to the target, which triggers the target system to expand the quota of the MailUser to 100 GB, thus allowing the move to the target. These instructions will work only for hybrid identity running Azure AD Connect, as the commands to stamp the ELC flags are not exposed to tenant administrators.
2. [!NOTE] SAMPLE – AS IS, NO WARRANTY
3. This script assumes a connection to both source mailbox (to get source values) and the target on-premises Active Directory (to stamp the ADUser object). If source has litigation or single item recovery enabled, set this on the destination account. This will increase the dumpster size of destination account to 100 GB.

4. powershell

```
5. $ELCValue = 0
6. if ($source.LitigationHoldEnabled) {$ELCValue = $ELCValue + 8}
   if ($source.SingleItemRecoveryEnabled) {$ELCValue = $ELCValue +
   16} if ($ELCValue -gt 0) {Set-ADUser -Server $domainController -
   Identity $destination.SamAccountName -Replace
   @{msExchELCMailboxFlags=$ELCValue}}
```

2. Non-hybrid target tenants can modify the quota on the Recoverable Items folder for the MailUsers prior to migration by running the following command to enable Litigation Hold on the MailUser object and increasing the quota to 100 GB: Set-MailUser -EnableLitigationHoldForMigration. Note this will not work for tenants in hybrid.
3. Users in the target organization must be licensed with appropriate Exchange Online subscriptions applicable for the organization. You may apply a license

in advance of a mailbox move but ONLY once the target MailUser is properly set up with ExchangeGUID and proxy addresses. Applying a license before the ExchangeGUID is applied will result in a new mailbox provisioned in target organization.

7. [!NOTE] When you apply a license on a Mailbox or MailUser object, all SMTP type proxyAddresses are scrubbed to ensure only verified domains are included in the Exchange EmailAddresses array.
4. You must ensure that the target MailUser has no previous ExchangeGuid that does not match the Source ExchangeGuid. This might occur if the target MEU was previously licensed for Exchange Online and provisioned a mailbox. If the target MailUser was previously licensed for or had an ExchangeGuid that does not match the Source ExchangeGuid, you need to perform a cleanup of the cloud MEU. For these cloud MEUs, you can run `Set-User <identity> -PermanentlyClearPreviousMailboxInfo`.
8. [!CAUTION] This process is irreversible. If the object has a softDeleted mailbox, it cannot be restored after this point. Once cleared, however, you can sync the correct ExchangeGuid to the target object and MRS will connect the source mailbox to the newly created target mailbox. (Reference EHLO blog on the new parameter.)
9. Find objects that were previously mailboxes using this command.

10. powershell

```
11. Get-User <identity> | select Name, *recipient* | ft -AutoSize
```

12. Here is an example.

13. powershell

```
14. PS demo> get-user John@northwindtraders.com |select name,  
*recipient*| ft -AutoSize
```

15.

```
16. Name          PreviousRecipientTypeDetails      RecipientType  
RecipientTypeDetails
```



```
17.  ----  -----  
-----  
18.  John      UserMailbox              MailUser      MailUser
```

19. Clear the soft-deleted mailbox using this command.

20. `powershell`

```
21.  Set-User <identity> -PermanentlyClearPreviousMailboxInfo
```

22. Here is an example.

23. `powershell`

```
24.  PS demo> Set-User John@northwindtraders.com -  
PermanentlyClearPreviousMailboxInfo Confirm  
25.  Are you sure you want to perform this action?  
26.  Delete all existing information about user  
    "John@northwindtraders.com"?. This operation will clear existing  
    values from Previous home MDB and Previous Mailbox GUID of the  
    user. After deletion, reconnecting to the previous mailbox that  
    existed in the cloud will not be possible and any content it had  
    will be unrecoverable PERMANENTLY.  
27.  Do you want to continue?  
28.  [Y] Yes [A] Yes to All [N] No [L] No to All [?] Help  
    (default is "Y"): Y
```

Perform mailbox migrations

Cross-tenant Exchange mailbox migrations are submitted as migration batches initiated from the target tenant. This is similar to the way that on-boarding migration batches work when migrating from Exchange on-premises to Microsoft 365.

Create Migration batches

Here is an example migration batch cmdlet for kicking off moves.

`powershell`

```

New-MigrationBatch -Name T2Tbatch-testforignitedemo -SourceEndpoint
target_source_7977 -CSVData
([System.IO.File]::ReadAllBytes('users.csv')) -Autostart -
TargetDeliveryDomain targetformoves.onmicrosoft.com

Identity                Status  Type                TotalCount
-----                -
T2Tbatch-testforignitedemo Syncing ExchangeRemoteMove 1

```

[!NOTE] The email address in the CSV file must be the one specified in the target tenant, not the source tenant. [For more information on the cmdlet click here](#) [For and example CSV file click here](#)

Migration batch submission is also supported from the new Exchange Admin Center when selecting the cross-tenant option.

Update on-premises MailUsers

Once the mailbox moves from source to target, you should ensure that the on-premises mail users, in both the source and target, are updated with the new targetAddress. In the examples, the targetDeliveryDomain used in the move is **contoso.onmicrosoft.com**. Update the mail users with this targetAddress.

Frequently asked questions

Do we need to update RemoteMailboxes in source on-premises after the move?

Yes, you should update the targetAddress (RemoteRoutingAddress/ExternalEmailAddress) of the source on-premises users when the source tenant mailbox moves to target tenant. While mail routing can follow the referrals across multiple mail users with different targetAddresses, Free/Busy lookups for mail users MUST target the location of the mailbox user. Free/Busy lookups will not chase multiple redirects.

Do Teams meetings migrate cross-tenant?

The meetings will move however the Teams meeting URL does not update when items migrate cross-tenant. Since the URL will be invalid in the target tenant you will need to remove and recreate the Teams meetings.

Does the Teams chat folder content migrate cross-tenant?

No, the Teams chat folder content does not migrate cross-tenant.

How can I see just moves that are cross-tenant moves, not my onboarding and off-boarding moves?

Use the `-flags` parameter. Here is an example.

powershell

```
Get-MoveRequest -Flags "CrossTenant"
```

Can you provide example scripts for copying attributes used in testing?

[!NOTE] SAMPLE – AS IS, NO WARRANTY

This script assumes a connection to both source mailbox (to get source values) and the target on-premises Active Directory Domain Services (to stamp the ADUser object). If source has litigation or single item recovery enabled, set this on the destination account. This will increase the dumpster size of destination account to 100 GB.

powershell

```
#Dumps out the test mailboxes from SourceTenant
#Note, the filter applied on Get-Mailbox is for an attribute set on
CustomAttribute1 = "ProjectKermit"
#These are the 'target' users to be moved to the Northwind org tenant
#####
$outFileUsers = "$home\desktop\userstomigrate.txt"
$outFileUsersXML = "$home\desktop\userstomigrate.xml"
#output the test objects
Get-Mailbox -Filter "CustomAttribute1 -like 'ProjectKermit'" -
ResultSize Unlimited | Select-Object -ExpandProperty Alias | Out-File
$outFileUsers
```

```

$mailboxes = Get-Content $outFileUsers
$mailboxes | ForEach-Object {Get-Mailbox $_} | Select-Object
PrimarySMTPAddress, Alias, SamAccountName, FirstName, LastName, DisplayName
, Name, ExchangeGuid, ArchiveGuid, LegacyExchangeDn, EmailAddresses |
Export-Clixml $outFileUsersXML

#####
#Copy the file $outfile to the desktop of the target on-premises
#then run the below to create MEU in Target
#####
$mailboxes = Import-Clixml $home\desktop\userstomigrate.xml

foreach ($m in $mailboxes) {
    $organization = "@contoso.onmicrosoft.com"
    $mosi = $m.Alias+$organization
    $Password =
[System.Web.Security.Membership]::GeneratePassword(16,4) | ConvertTo-
SecureString -AsPlainText -Force
    $x500 = "x500:" +$m.LegacyExchangeDn
    $tmpUser = New-MailUser -MicrosoftOnlineServicesID $mosi -
PrimarySmtptAddress $mosi -ExternalEmailAddress $m.PrimarySmtptAddress -
FirstName $m.FirstName -LastName $m.LastName -Name $m.Name -
DisplayName $m.DisplayName -Alias $m.Alias -Password $Password
    $tmpUser | Set-MailUser -EmailAddresses @{add=$x500} -ExchangeGuid
$m.ExchangeGuid -ArchiveGuid $m.ArchiveGuid -CustomAttribute1
"ProjectKermit"
    $tmpx500 = $m.EmailAddresses | ?{$_ -match "x500"}
    $tmpx500 | %{Set-MailUser $m.Alias -EmailAddresses @{add="$_"}}
}

#####
# On AADSync machine, run AADSync
#####
Start-ADSyncSyncCycle

```

#AADSync and FWDSync will create the target MEUs in the Target tenant

How do we access Outlook on Day 1 after the use mailbox is moved?

Since only one tenant can own a domain, the former primary SMTPAddress will not be associated to the user in the target tenant when the mailbox move completes; only those domains associated with the new tenant. Outlook uses the users new UPN to authenticate to the service and the Outlook profile expects to find the legacy primary SMTPAddress to match the mailbox in the target system. Since the legacy address is not in the target System the outlook profile will not connect to find the newly moved mailbox.

For this initial deployment, users will need to rebuild their profile with their new UPN, primary SMTP address and re-sync OST content.

[!NOTE] Plan accordingly as you batch your users for completion. You need to account for network utilization and capacity when Outlook client profiles are created and subsequent OST and OAB files are downloaded to clients.

What Exchange RBAC roles do I need to be member of to set up or complete a cross-tenant move?

There a matrix of roles based on assumption of delegated duties when executing a mailbox move. Currently, two roles are required:

- The first role is for a one-time setup task that establishes the authorization of moving content into or out of your tenant/organizational boundary. As moving data out of your organizational control is a critical concern for all companies, we opted with the highest assigned role of Organization Administrator (OrgAdmin). This role must alter or setup a new OrganizationRelationship that defines the -MailboxMoveCapability with the remote organization. Only the OrgAdmin can alter the MailboxMoveCapability setting, while other attributes on the OrganizationRelationship can be managed by the Federated Sharing administrator.

- The role of executing the actual move commands can be delegated to a lower-level function. The role of Move Mailboxes is assigned the capability of moving mailboxes in or out of the organization.

How do we target which SMTP address is selected for targetAddress (TargetDeliveryDomain) on the converted mailbox (to MailUser conversion)?

Exchange mailbox moves using MRS craft the targetAddress on the original source mailbox when converting to a MailUser by matching an email address (proxyAddress) on the target object. The process takes the -TargetDeliveryDomain value passed into the move command, then checks for a matching proxy for that domain on the target side. When we find a match, the matching proxyAddress is used to set the ExternalEmailAddress (targetAddress) on the converted mailbox (now MailUser) object.

How do mailbox permissions transition?

Mailbox permissions include Send on Behalf of and Mailbox Access:

- Send On Behalf Of (AD:publicDelegates) stores the DN of recipients with access to a user's mailbox as a delegate. This value is stored in Active Directory and currently does not move as part of the mailbox transition. If the source mailbox has publicDelegates set, you will need to restamp the publicDelegates on the target Mailbox once the MEU to Mailbox conversion completes in the target environment by running `Set-Mailbox <principle> -GrantSendOnBehalfTo <delegate>`.
- Mailbox Permissions that are stored in the mailbox will move with the mailbox when both the principal and the delegate are moved to the target system. For example, the user TestUser_7 is granted FullAccess to the mailbox TestUser_8 in the tenant SourceCompany.onmicrosoft.com. After the mailbox move completes to TargetCompany.onmicrosoft.com, the same permissions are set up in the target directory. Examples using *Get-MailboxPermission* for TestUser_7 in both source and target tenants are shown below. Exchange cmdlets are prefixed with source and target accordingly.

Here's an example of the output of the mailbox permission before a move.

powershell

```
PS C:\PowerShell\> Get-SourceMailboxPermission testuser_7 | ft -
AutoSize User, AccessRights, IsInherited, Deny
User                AccessRights
IsInherited Deny
----                -
-----
NT AUTHORITY\SELF   {FullAccess,
ReadPermission}      False
False
TestUser_8@SourceCompany.onmicrosoft.com {FullAccess}
False      False....
```

Here's an example of the output of the mailbox permission after the move.

powershell

```
PS C:\PowerShell\> Get-TargetMailboxPermission testuser_7 | ft -
AutoSize User, AccessRights, IsInherited, Deny
User                AccessRights
IsInherited Deny
----                -
-----
NT AUTHORITY\SELF   {FullAccess,
ReadPermission}      False
FalseTestUser_8@TargetCompany.onmicrosoft.com {FullAccess}
False      False
```

[!NOTE] Cross-tenant mailbox and calendar permissions are NOT supported. You must organize principals and delegates into consolidated move batches so that these connected mailboxes are transitioned at the same time from the source tenant.

What X500 proxy should be added to the target MailUser proxy addresses to enable migration?

The cross-tenant mailbox migration requires that the LegacyExchangeDN value of the source mailbox object to be stamped as an x500 email address on the target MailUser object.

Example:

powershell

```
LegacyExchangeDN value on source mailbox is:  
/o=First Organization/ou=Exchange Administrative  
Group(FYDIBOHF23SPDLT)/cn=Recipients/cn=d11ec1a2cacd4f81858c81907273f1  
f9Lara  
  
so the x500 email address to be added to target MailUser object would  
be:  
x500:/o=First Organization/ou=Exchange Administrative Group  
(FYDIBOHF23SPDLT)/cn=Recipients/cn=d11ec1a2cacd4f81858c81907273f1f9-  
Lara
```

[!NOTE] In addition to this X500 proxy, you will need to copy all X500 proxies from the mailbox in the source to the mailbox in the target.

Can the source and target tenant utilize the same domain name?

No. The source and target tenant domain names must be unique. For example, a source domain of contoso.com and the target domain of fourthcoffee.com.

Will shared mailboxes move and still work?

Yes, however we only keep the store permissions as described in these articles:

- [Microsoft Docs | Manage permissions for recipients in Exchange Online](#)
- [Microsoft Support | How to grant Exchange and Outlook mailbox permissions in Office 365 dedicated](#)

Do you have any recommendations for batches?

Do not exceed 2000 mailboxes per batch. We strongly recommend submitting batches two weeks prior to the cut-over date as there is no impact to the end users during sync. If you need guidance for mailboxes quantities over 50,000 you can

reach out to the Engineering Feedback Distribution List at crosstenantmigrationpreview@service.microsoft.com.

What if I use Service encryption with Customer Key?

The mailbox will be decrypted prior to moving. Ensure Customer Key is configured in the target tenant if it is still required. See [here](#) for more information.

What is the estimated migration time?

To help you plan your migration, the table present [here](#) shows the guidelines about when to expect bulk mailbox migrations or individual migrations to complete. These estimates are based on a data analysis of previous customer migrations. Because every environment is unique, your exact migration velocity may vary.

Do remember that this feature is currently in preview and the SLA and any applicable Service Levels do not apply to any performance or availability issues during the preview status of this feature.

Making documents protected in the source tenant consumable by users in the destination tenant.

Cross-tenant migration only migrates mailbox data and nothing else. There are multiple other options which are documented in the following blog post that may help: <https://techcommunity.microsoft.com/t5/security-compliance-and-identity/mergers-and-spinoffs/ba-p/910455>

Can I have the same labels in the destination tenant as you had in the source tenant, either as the only set of labels or an additional set of labels for the migrated users depending on alignment between the organizations.

Since Cross-tenant migrations do not export labels and there is no way to share labels between tenants you can only achieve this by recreating the labels in the destination tenant. While this can be time consuming it is typically done manually. There is a tool at <https://microsoft365dsc.com/>, that can be used to export a copy of the label configuration from the source tenant and then import it into the destination tenant. This is a copy of the label and not a migration so content that is labeled in the source would potentially not be labeled in the destination.

[!Caution] SAMPLE – AS IS, NO WARRANTY

All issues with Microsoft365DSC should be raised via there GitHub site at <https://github.com/microsoft/Microsoft365DSC/issues>

Do you support moving Microsoft 365 Groups?

At this time the Cross-Tenant mailbox migrations feature does not support the migration of Microsoft 365 Groups.

Known issues

- **Issue: Post migration Teams functionality in the source forest will be limited** After the mailbox is migrated to the target tenant, Teams in the source tenant will no longer have access to the users mailbox. So if a user logs into Teams with the source tenant credential then there will be a loss of functionality such as the inability to update your profile picture, no calendar application and a inability to search and join public teams.
- **Issue: Auto Expanded archives cannot be migrated.** The cross-tenant migration feature support migrations of the primary mailbox and archive mailbox for a specific user. If the user in the source however has an auto expanded archive – meaning more than one archive mailbox, the feature is unable to migrate the additional archives and should fail.
- **Issue: Cloud MailUsers with non-owned smtp proxyAddress block MRS moves background.** When creating target tenant MailUser objects, you must ensure that all SMTP proxy addresses belong to the target tenant organization. If an SMTP proxyAddress exists on the target mail user that does not belong to the local tenant, the conversion of the MailUser to Mailbox is prevented. This is due to our assurance that mailbox objects can only send mail from domains for which the tenant is authoritative (domains claimed by the tenant):
 - When you sync users from on-premises using Azure AD Connect, you provision on-premises MailUser objects with ExternalEmailAddress pointing to the source tenant where the mailbox exists (laran@contoso.onmicrosoft.com) and you stamp the PrimarySMTPAddress as a domain that resides in the target tenant (Lara.Newton@northwind.com). These values sync down to the tenant

and an appropriate mail user is provisioned and ready for migration. An example object is shown here.

1. powershell

```
2. target/AADSynced user] PS C> Get-MailUser laran | select
   ExternalEmailAddress, EmailAddresses
3. ExternalEmailAddress          EmailAddresses
4. -----
5. SMTP:laran@contoso.onmicrosoft.com
   {SMTP:lara.newton@northwind.com}
```

29. [!NOTE] The *contoso.onmicrosoft.com* address is *not* present in the `EmailAddresses / proxyAddresses` array.

- **Issue: MailUser objects with “external” primary SMTP addresses are modified / reset to “internal” company claimed domains**

30. MailUser objects are pointers to non-local mailboxes. In the case for cross-tenant mailbox migrations, we use MailUser objects to represent either the source mailbox (from the target organization’s perspective) or target mailbox (from the source organization’s perspective). The MailUsers will have an ExternalEmailAddress (targetAddress) that points to the smtp address of the actual mailbox (ProxyTest@fabrikam.onmicrosoft.com) and primarySMTP address that represents the displayed SMTP address of the mailbox user in the directory. Some organizations choose to display the primary SMTP address as an external SMTP address, not as an address owned/verified by the local tenant (such as fabrikam.com rather than as contoso.com). However, once an Exchange service plan object is applied to the MailUser via licensing operations, the primary SMTP address is modified to show as a domain verified by the local organization (contoso.com). There are two potential reasons:

- When any Exchange service plan is applied to a MailUser, the Azure AD process starts to enforce proxy scrubbing to ensure that the local organization is not able to send mail out, spoof, or mail from another tenant. Any SMTP address on a recipient object with these service

plans will be removed if the address is not verified by the local organization. As is the case in the example, the Fabikam.com domain is NOT verified by the contoso.onmicrosoft.com tenant, so the scrubbing removes that fabrikam.com domain. If you wish to persist these external domain on MailUser, either before the migration or after migration, you need to alter your migration processes to strip licenses after the move completes or before the move to ensure that the users have the expected external branding applied. You will need to ensure that the mailbox object is properly licensed to not affect mail service.

- An example script to remove the service plans on a MailUser in the Contoso.onmicrosoft.com tenant is shown here.

1. powershell

```
2. $LO = New-MsolLicenseOptions -AccountSkuId
   "contoso:ENTERPRISEPREMIUM" DisabledPlans
3. "LOCKBOX_ENTERPRISE", "EXCHANGE_S_ENTERPRISE", "INFORMATION_
   BARRIERS", "MIP_S_CLP2", "
4. MIP_S_CLP1", "MYANALYTICS_P2", "EXCHANGE_ANALYTICS", "EQUIVIO
   _ANALYTICS", "THREAT_INTE
5. LLIGENCE", "PAM_ENTERPRISE", "PREMIUM_ENCRYPTION"
6. Set-MsolUserLicense -UserPrincipalName
   proxytest@contoso.com LicenseOptions $lo
```

7. Results in the set of ServicePlans assigned are shown here.

8. powershell

```
9. (Get-MsolUser -UserPrincipalName
   proxytest@contoso.com).licenses |select
10. -ExpandProperty servicestatus |sort ProvisioningStatus -
   Descending
11. ServicePlan          ProvisioningStatus
12. -----
13. ATP_ENTERPRISE      PendingProvisioning
14. MICROSOFT_SEARCH    PendingProvisioning
```

15.	INTUNE_0365	PendingActivation
16.	PAM_ENTERPRISE	Disabled
17.	EXCHANGE_ANALYTICS	Disabled
18.	EQUIVIO_ANALYTICS	Disabled
19.	THREAT_INTELLIGENCE	Disabled
20.	LOCKBOX_ENTERPRISE	Disabled
21.	PREMIUM_ENCRYPTION	Disabled
22.	EXCHANGE_S_ENTERPRISE	Disabled
23.	INFORMATION_BARRIERS	Disabled
24.	MYANALYTICS_P2	Disabled
25.	MIP_S_CLP1	Disabled
26.	MIP_S_CLP2	Disabled
27.	ADALLOM_S_0365	PendingInput
28.	RMS_S_ENTERPRISE	Success
29.	YAMMER_ENTERPRISE	Success
30.	PROJECTWORKMANAGEMENT	Success
31.	BI_AZURE_P2	Success
32.	WHITEBOARD_PLAN3	Success
33.	SHAREPOINTENTERPRISE	Success
34.	SHAREPOINTWAC	Success
35.	KAIZALA_STANDALONE	Success
36.	OFFICESUBSCRIPTION	Success
37.	MCSTANDARD	Success
38.	Deskless	Success
39.	STREAM_0365_E5	Success
40.	FLOW_0365_P3	Success
41.	POWERAPPS_0365_P3	Success
42.	TEAMS1	Success
43.	MCOEV	Success
44.	MCOMEETADV	Success
45.	BPOS_S_TODO_3	Success
46.	FORMS_PLAN_E5	Success
47.	SWAY	Success

48. The user's PrimarySMTPAddress is no longer scrubbed. The fabrikam.com domain is not owned by the contoso.onmicrosoft.com tenant and will persist as the primary SMTP address shown in the directory.

49. Here is an example.

50. powershell

```
51. get-recipient proxytest | ft -a userprin*, primary*,
    external*
52. PrimarySmtpAddress      ExternalDirectoryObjectId
    ExternalEmailAddress
53. -----
    -----
54. proxytest@fabrikam.com   e2513482-1d5b-4066-936a-
    cbc7f8f6f817      SMTP:proxytest@fabrikam.com
```

- When msExchRemoteRecipientType is set to 8 (DeprovisionMailbox), for on-premises MailUsers that are migrated to the target tenant, the proxy scrubbing logic in Azure will remove nonowned domains and reset the primarySMTP to an owned domain. By clearing msExchRemoteRecipientType in the on-premises MailUser, the proxy scrub logic no longer applies.

1. Below is the full set of possible Service Plans that include Exchange Online.

Name
Advanced eDiscovery Storage (500GB)
Customer Lockbox
Data Loss Prevention
Exchange Enterprise CAL Services (EOP, DLP)

Exchange Essentials
Exchange Foundation
Exchange Online (P1)
Exchange Online (Plan 1)
Exchange Online (Plan 2)
Exchange Online Archiving for Exchange Online
Exchange Online Archiving for Exchange Server
Exchange Online Inactive User Add-on
Exchange Online Kiosk
Exchange Online Multi-Geo
Exchange Online Plan 1
Exchange Online POP
Exchange Online Protection
Information Barriers
Information Protection for Office 365 - Premium
Information Protection for Office 365 - Standard
Insights by MyAnalytics
Microsoft 365 Advanced Auditing
Microsoft Bookings
Microsoft Business Center
Microsoft MyAnalytics (Full)
Office 365 Advanced eDiscovery

Microsoft Defender for Office 365 (Plan 1)
Microsoft Defender for Office 365 (Plan 2)
Office 365 Privileged Access Management
Premium Encryption in Office 365