

LSEG Workspace | Automated Domain Management (ADM)

Installation and configuration guide

Contents

About this document	3
In this guide	3
Intended readership.....	3
Further information	3
About Automated Domain Management (ADM)	4
Roles and relationships	4
Customer organisation	4
LSEG (London Stock Exchange Group)	4
ADM workflow.....	5
Deploying the ADM application	6
Pre-requisites for deployment.....	6
Deploying ADM using a custom template.....	7
Creating an app registration	7
Registering the backend	7
Registering the frontend.....	10
Adding the Teams Administrator role to ADM.....	11
Obtaining an API key and a Container Registry password	11
Deploying the ARM template	12
Setting up a redirect URI for authentication.....	13
Post-deployment administration.....	14
Managing configuration	14
Creating a base policy	15
Creating a policy in the Teams Admin Center	15
Configuring domain feeds	17
Subscribing Admin users to notifications.....	18
Notifying Admin users of domain changes	18
Managing domains	19
Managing policies.....	20
Support	23
Appendix A: Required permissions	24
Example of permissions being used	24
Appendix B: Azure resources	25
Appendix C: Frequently asked questions.....	26
What is the Automated Domain Management (ADM) app?	26
What problem does the ADM app solve?	26
How does the app work?	26
What is Granular Federation Control?	26
How does the ADM app know which settings and domains to configure?	27
How does the ADM app know which users should be assigned the policy?.....	27
Do I have control over these policies?.....	27
How does the ADM app handle security and compliance?	27

What technology does this use and how is it deployed? 27

About this document

The Automated Domain Management (ADM) app helps Microsoft Teams administrators automatically manage external access policies using LSEG's Open Directory (OD) network. This automation minimises manual effort and delivers scalable federation across member organizations.

In this guide

This guide outlines the steps by which Automated Domain Management (ADM) can be installed, configured, and managed.

Intended readership

This guide is intended for LSEG Workspace customers who want to install and use the ADM application within their Azure environment.

Further information

To:

- Request product assistance, contact [Support](#).
- Access other LSEG Workspace technical content, see the [Workspace technical documentation site](#).
- Provide feedback on Workspace technical content, contact DocFeedback@lsegu.com.

About Automated Domain Management (ADM)

This section describes the different roles and relationships that are involved in the ADM deployment process.

Roles and relationships

To ensure clarity for all stakeholders, the following roles and relationships should be explicitly defined:

Customer organisation

Role	Relationship with LSEG	Relationship with Microsoft
Consumes the Automated Domain Management (ADM) app to manage external access policies for Microsoft Teams in connection with use of Open Directory (OD).	Acts as a participant in LSEG's Open Directory (OD) network, leveraging OD for federation across member organisations.	Uses Microsoft Teams as the collaboration platform.

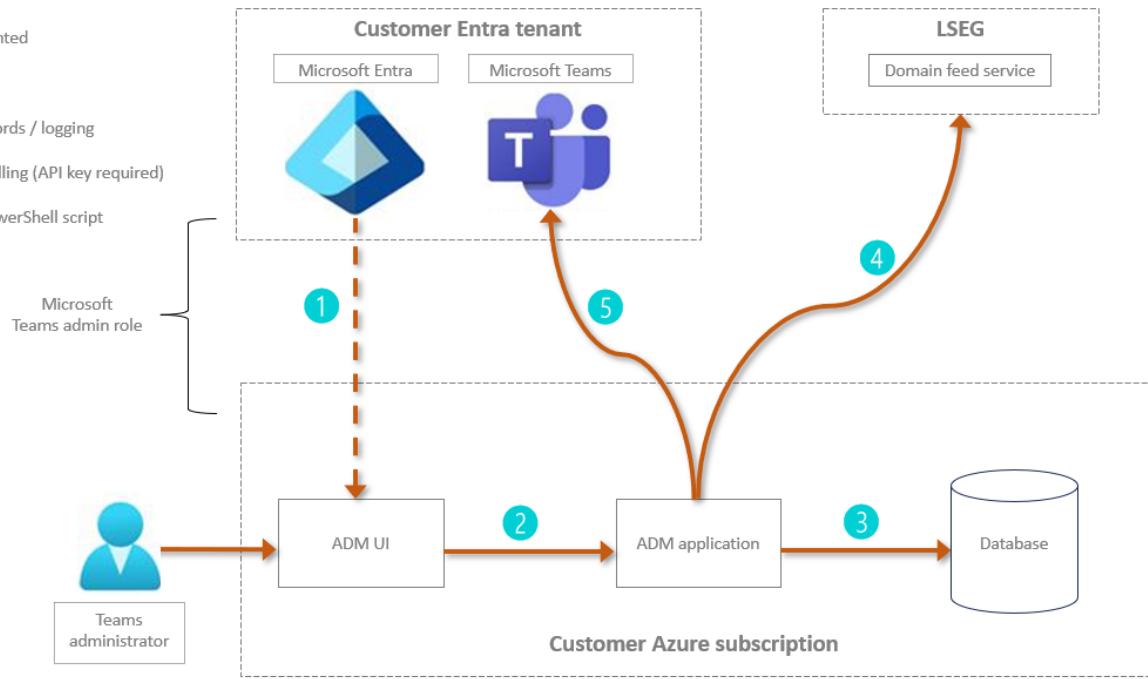
LSEG (London Stock Exchange Group)

Role	Relationship with customer organisation	Relationship with Microsoft
Provides and maintains the Open Directory (OD) network. Develops and supports the ADM app which supports federation.	Serves as the directory authority, ensuring OD membership integrity and policy enforcement. Provides technical support, documentation, and compliance guidance for ADM deployment.	Collaborates on integration standards to ensure OD and ADM works seamlessly with Microsoft Teams.

ADM workflow

The following diagram presents an architectural overview of ADM.

- 1 Service principle granted
- 2 Configuration
- 3 Manage domain records / logging
- 4 Automatic hourly pulling (API key required)
- 5 Update policy via PowerShell script



Deploying the ADM application

The ADM app is a client-side application that must be deployed onto a tenant's Azure cloud environment:

- By an administrator with the appropriate permissions to deploy Azure services onto an Azure cloud environment, and
- Via an Azure Resource Management (ARM) template

➤ More information about the minimum required roles for ADM deployment is described in the [Azure resources](#) section. Additionally, you can view Teams administrator details [here](#).

Pre-requisites for deployment

Customers must have the following in place before deploying ADM:

Pre-requisite	Role / permission required	Reason
Azure subscription	For details on the required role / permissions, see Appendix B: Azure resources . Note that a customer's Azure policy must enable public network access for applications.	Required for creating Azure resources.
Azure ID	Teams administrator	Required to use private azure application
Entra ID	Entra application administrator	Required for creating App Registrations, consent to Graph API permissions and assign Directory roles.
Teams environment	Teams administrator	Required to enable the administrator to perform updates to Teams policies and domains using the ADM app.

➤ Ideally, all the pre-requisites would be part of the same subscription; however, deployment is still possible if this is not the case.

Deploying ADM using a custom template

Creating an app registration

This is a required step so ADM can manage domains and policies on the client tenant, including allowing specific domains.

➤ Note that this step is currently manual, but will be automated later.

To create an app registration, you must:

- [Register the backend](#)
- [Register the frontend](#), and
- [Add the Teams administrator role](#)

Registering the backend

To register the backend:

1. Go to the Azure Portal and login with your account.
2. Go to [App Registrations](#).
3. Click **New registration**.
4. In the **Name** field, type **adm-backend** ①.
5. Select the **Accounts in this organisational directory only (<tenant name> only - Single tenant)** radio button ②.
6. Click the **Register** button ③.

Home > App registrations > Register an application ...

* Name
The user-facing display name for this application (this can be changed later).
adm-backend ①

Supported account types
Who can use this application or access this API?
 Accounts in this organizational directory only (LSEG Workspace B only - Single tenant) ②
 Accounts in any organizational directory (Any Microsoft Entra ID tenant - Multitenant)
 Accounts in any organizational directory (Any Microsoft Entra ID tenant - 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.
Select a platform e.g. https://example.com/auth

Register an app you're working on here. Integrate gallery apps and other apps from outside your organization by adding from Enterprise applications.

By proceeding, you agree to the Microsoft Platform Policies [View](#)

Register ③

Configuring the backend

➤ **IMPORTANT:** If you do not make a note of the values which are required in this step, you will not be able [to deploy an ARM template](#).

To configure adm-backend:

1. Select App Registration > adm-backend.
2. In the **Certificates & secrets** tab, click **New client secret**.
 - a) Enter the **Description**: adm-backend-secret
 - b) In the **Expires** dropdown, select 730 days (24 months)
 - c) Click the **Add** button
3. Copy the **Value** that is generated. This is an important step, as you cannot go back to view these values.

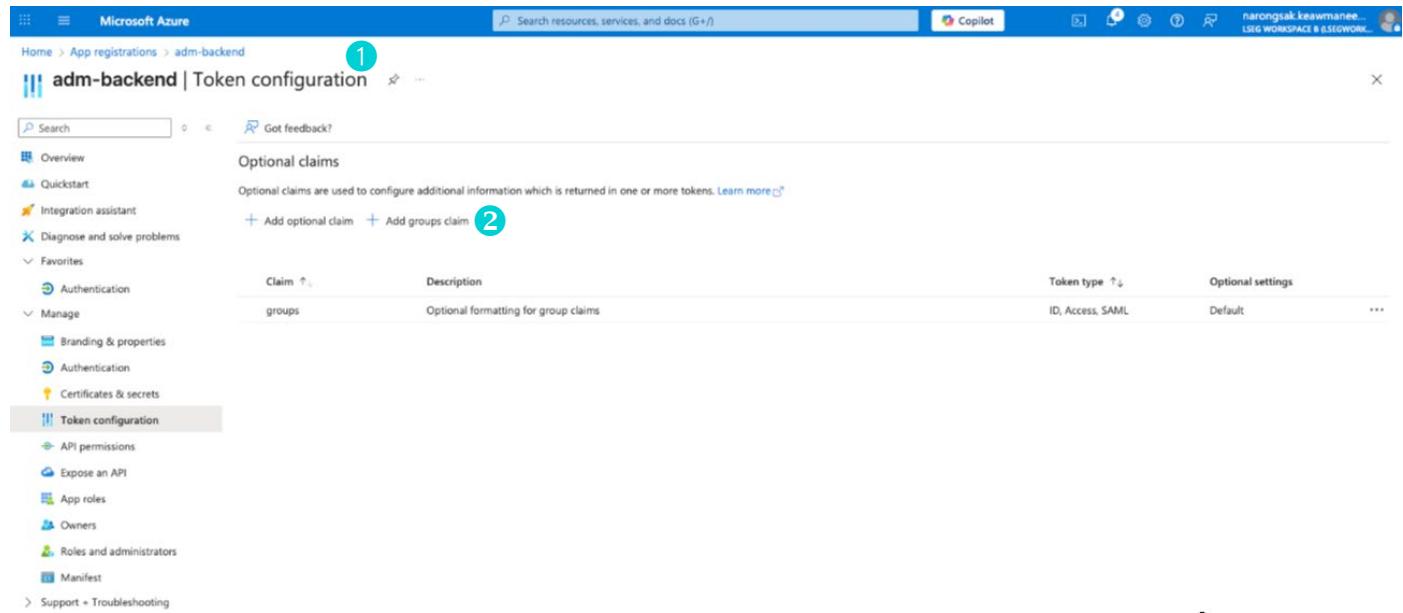
Configuring a token

You must add a groups claim to enable ADM to verify that the login user is a Teams Administrator.

To do this:

1. Select **Token configuration** ①.

2. Click **Add groups claim** ②.



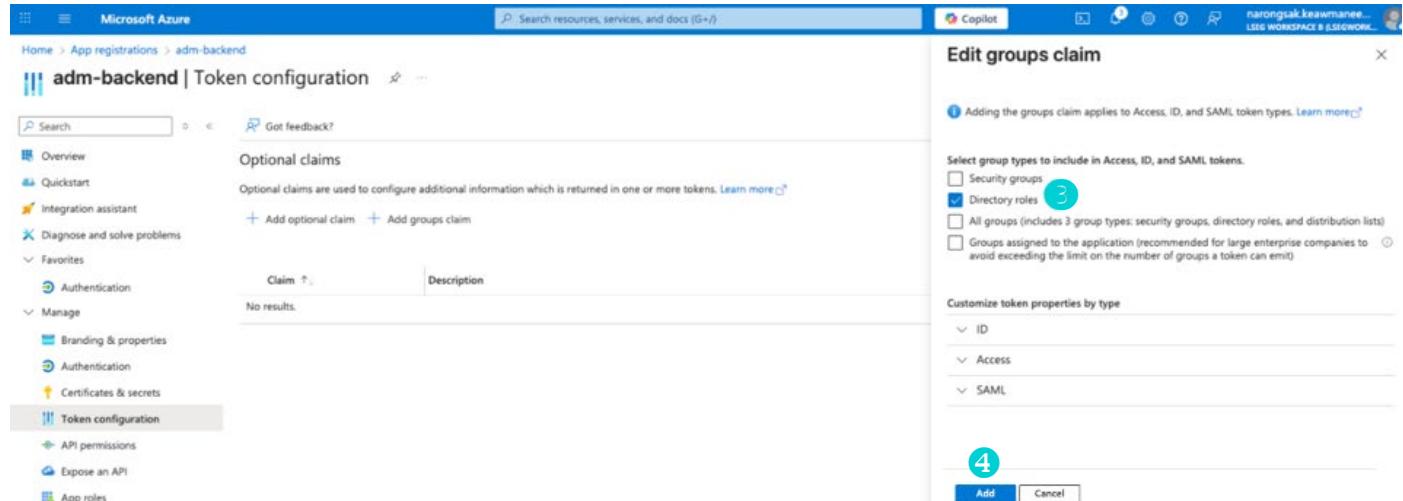
Optional claims

Optional claims are used to configure additional information which is returned in one or more tokens. [Learn more](#)

+ Add optional claim + Add groups claim ②

Claim	Description	Token type	Optional settings
groups	Optional formatting for group claims	ID, Access, SAML	Default

3. Check the **Directory roles** box ③.



Optional claims

Optional claims are used to configure additional information which is returned in one or more tokens. [Learn more](#)

+ Add optional claim + Add groups claim

Claim	Description
No results.	

Select group types to include in Access, ID, and SAML token types. [Learn more](#)

Security groups
 Directory roles ③
 All groups (includes 3 group types: security groups, directory roles, and distribution lists)
 Groups assigned to the application (recommended for large enterprise companies to avoid exceeding the limit on the number of groups a token can emit)

Customize token properties by type

✓ ID
✓ Access
✓ SAML

4

Add Cancel

4. Click the **Add** button ④.

Adding API permissions

To add an API permission:

1. Click **Add a permission**.
2. Select **Microsoft Graph** and then choose **Application Permissions**.
3. The permissions are as follows:
 - Application.Read.All – read all applications
 - Group.ReadWrite.All – read and write all groups
 - GroupMember.Read.All – read all group memberships
 - Mail.Send – send mail as any user
 - Organization.Read.All – read organisation information
 - User.ReadBasic.All – read all users' basic profiles
4. If it was created automatically, you should remove the User.Read permission. This permission is not required for the adm-backend.
5. Click **Grant admin consent** for all permissions.

↗ This is a required step for the app to work.

Exposing an API

To expose an API, you need to:

1. Create an application ID URI and click on it.
2. Add a scope as follows:
 - a) In the **Scope name** field, type 'access_as_user' **1**.
 - b) Select 'Admins and users' in the **Who can consent?** field **2**.
 - c) In the **Admin consent display name** field, type 'Access ADM backend API' **3**.
 - d) In the **Admin consent description**, type 'Access ADM backend API' **4**.
 - e) Ensure the **State** is 'Enabled' **5**.
 - f) Click the **Add scope** button **6**.

Obtaining the adm-backend client ID

To obtain the adm-backend client ID:

1. Return to **Overview** menu in App Registration > adm-backend
2. Copy the Application (client) ID. This will be used for the 'Backend Azure Client ID' in the [ARM Template](#).

Add a scope

Scope name * ⓘ
access_as_user **1**
api://c0b7f41b-4fca-4cd8-a419-facbd1bd32f/access_as_user

Who can consent? ⓘ **2**
Admins and users **Admins only**

Admin consent display name * ⓘ
Access ADM backend API **3**

Admin consent description * ⓘ
Access ADM backend API **4**

User consent display name ⓘ
e.g. Read your files

User consent description ⓘ
e.g. Allows the app to read your files.

State ⓘ **5**
Enabled **Disabled**

6 Add scope Cancel

Registering the frontend

To register the frontend:

1. Click **New registration** to create a new App Registration for the frontend app.
2. In the **Name** field, type **adm-frontend** ①.
3. Select the **Accounts in this organisational directory only (<tenant name> only - Single tenant)** radio button ②.
4. Click the **Register** button ③.

Home > App registrations > Register an application ...

* Name
The user-facing display name for this application (this can be changed later).
adm-frontend ①

Supported account types
Who can use this application or access this API?
 Accounts in this organizational directory only (LSEG Workspace B only - Single tenant) ②
 Accounts in any organizational directory (Any Microsoft Entra ID tenant - Multitenant)
 Accounts in any organizational directory (Any Microsoft Entra ID tenant - 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.
Select a platform e.g. https://example.com/auth

Register an app you're working on here. Integrate gallery apps and other apps from outside your organization by adding from [Enterprise applications](#).

By proceeding, you agree to the Microsoft Platform Policies [\[link\]](#)

Register ③

Configuring the frontend

To configure adm-frontend:

1. Select **App Registration** > adm-frontend.
2. Select **API permission** and click **Add a permission**.
3. If 'User.Read' was not automatically created, select **Microsoft Graph** and add it as a permission, ensuring the type of permission is 'Delegated'.

Microsoft Graph (1)

User.Read	Delegated	Sign in and read user profile	No	...
-----------	-----------	-------------------------------	----	-----

4. Click **Add a permission** > select **APIs my organization uses** > adm-backend.
5. Select **Delegated permissions**. The resulting screen displays as follows:

[access_as_user](#) Delegated Access ADM backend API

6. Select the permission **access_as_user** and click the **Add permissions** button.
7. Grant admin consent to all permissions. This is a required step for the app to work.

Adding the Teams Administrator role to ADM

To add the Teams Administrator role to ADM:

1. Go to **Microsoft Entra roles and administrators** in Azure.
2. Search for 'Teams Administrator' and click it.
3. To add the required assignments to the Teams administrator role:
 - a) Click on **Add assignments**.
 - b) Select the member(s) for whom you want to add assignments.
4. Search for **adm-backend** and select it.
5. Click the **Next** button.
6. Select **Active**.
7. Select **Permanently assigned**.
8. Click the **Assign** button.

The Teams Administrator role is now created and assigned.

Name	Principal name	Type	Scope	Membership	State
Teams Administrator					
adm-backend	e4cd6ce0-2a55-4883-93a...	Service principal	Directory	Direct	Assigned

Obtaining the adm-frontend client ID

To obtain the adm-frontend client ID:

1. Return to the **Overview** menu in App Registration > adm-frontend
2. Copy the **Application (client) ID**. This will be used as the 'Frontend Client ID' in the [ARM template](#).
3. Ensure the following are all saved for use in ADM template deployment:
 - Backend Azure Client Id
 - Backend Azure Client Secret
 - Frontend Client Id

Obtaining an API key and a Container Registry password

The LSEG API Key and LSEG Container Registry Password will be provided to customers by LSEG as part of the onboarding process.

- The API Key is unique for each client.
- The LSEG Container Registry Password is required for accessing container resources needed for ADM backend deployment.
- Contact WSTEAMsonboarding@lsegu.com if you experience any issues with your API key or password.

Deploying the ARM template

To deploy the ARM template:

1. Ensure you have the following required information:
 - Backend Azure Client Id
 - Backend Azure Client Secret
 - Frontend Client Id
 - LSEG API Key
 - LSEG Container Registry Password
2. Open the Azure Portal and load the LSEG ARM template for ADM deployment.
3. In the **Project details** section of the screen:
 - a) Select your **Subscription** 1.
 - b) Select existing **Resource group** 2 or create a new one (recommended).
4. In the **Instance details** section of the screen:
 - a) Select the **Region** 3 where the ADM should be deployed.
 5. Enter the **Backend Azure Client Id** 4 (Application Client ID).
 6. Enter **Backend Azure Client Secret** 5.
 7. Enter **Frontend Client Id** 6 (Application Client ID).
 8. Enter the **LSEG API Key** 7.
 9. Enter the LSEG Container Registry Password, and then click **Review + create** 8.
10. Review the terms and click **Create** to start the deployment.

Home > **Custom deployment** ... Difference between ARM Template, Terraform & Bicep? Can I deploy multi-region?

Template

⚠ Ensure that the source of the template is trustworthy. Deploying a template from untrustworthy sources can result in your cloud environment being compromised.

Customized template 19 resources Edit template Edit parameters Visualize

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * 1 Resource group * 2 Create new

Instance details

Region * 3 Backend Azure Client Id * 4 Backend Azure Client Secret * 5 Frontend Client Id * 6 Lseg Api Key * 7 Lseg Container Registry Password * 8

Review + create

After the deployment has been completed, the ADM application URL will be displayed in **Outputs** 9.

Home > Microsoft.Template-20251024131811

Microsoft.Template-20251024131811 | Outputs ... X

Outputs

adm portal URL 9 https://purple-meadow-0aa879d0f3.azurestaticapps.net

deployment status Successfully deployed.

Give Feedback

Tell us about your experience with the Deployment Outputs page

Setting up a redirect URI for authentication

This step is required to bind the Entra login and make it redirect to the ADM app after a successful login.

1. Copy the URL created when [Deploying the ARM template](#) (see previous page). This is required for adding the URI in the adm-frontend.
2. Go to **App Registration**.
3. Search for, and select, **adm-frontend**.
4. Go to **Manage > Authentication > Platform configuration**.
5. Click **Add a platform**.
6. In the displayed panel, select **Single-page application**.
7. Enter **Redirect URIs** using the adm-frontend URL you have copied (see Step 1).
8. Click **Configure**.
9. Select the **Accounts in this organizational directory only (<tenant name> only - Single tenant)** radio button.
10. Click the **Save** button.

The redirect URI has now been set up for authentication.

Post-deployment administration

After ADM has been installed / deployed, customers can manage a range of administrative tasks, including:

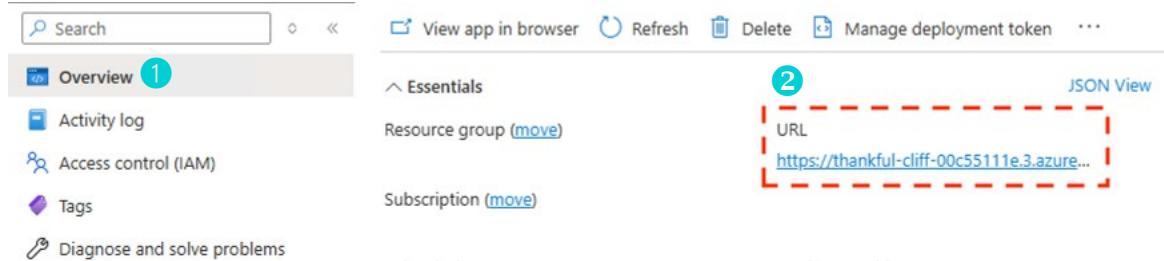
- [Managing configuration](#)
- [Creating a base policy](#)
- [Managing domains](#)
- [Managing policies](#)

Managing configuration

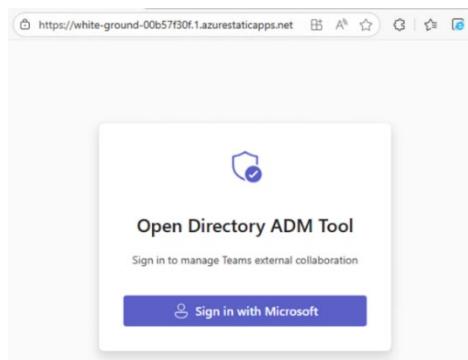
Customers are required to configure ADM before using the unique API Key provided to them by LSEG.

To configure ADM:

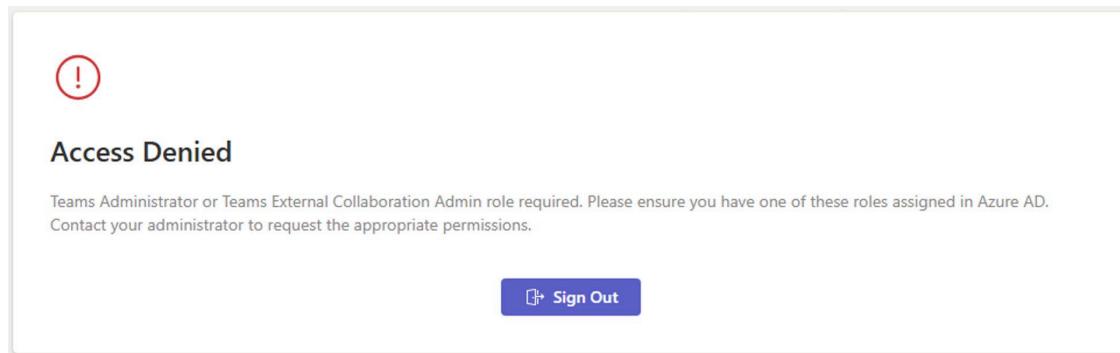
1. Open the ADM tool by using adm-frontend URL from [Deploying the ARM template](#) or by finding it in Static Web Apps > adm-frontend > Overview 1 > URL 2.



2. In the resulting popup window, click **Sign in with Microsoft**.



☞ If you have not been assigned the Microsoft Teams Administrator role, you will be blocked from accessing the app and the following window will be displayed.



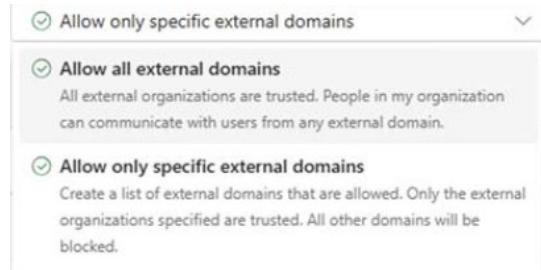
Creating a base policy

The base policy is managed in the Teams Admin Center, outside the ADM app. ADM will not interfere with the existing policy because it will create an inherited version of the base policy instead.

Creating a policy in the Teams Admin Center

To create a base policy in the Teams Admin Center:

1. In the **Organization Settings** tab, select either **Allow all external domains** or **Allow only specific external domains** from the dropdown list.



2. Go to the **Policies** tab ①.

The image shows the Microsoft Teams Admin Center with the 'External access' section selected. The 'Policies' tab is active (marked with a circled 1). A summary box displays '4 Default policies' and '26 Custom policies'. Below the summary, there are buttons for '+ Add', 'Edit', 'Duplicate', 'Delete', 'Reset Global policy', 'Assign users', and a total count of '30 items'.

3. Click **Add** ②.

4. Enter the **Name** of the new policy **3**.
5. Enter a **Description** of the policy (Optional) **4**.
6. Turn the **Teams and Skype for Business users in external organizations** switch to **On**. **5** This option is a minimum requirement for chat with external organizations.

External access policies \ Add

Add policy details

Name
Add a name for your external access policy **3**

Description
Add a description so you know why it was created **4**

External access policies might be overridden by your organization settings. [\(External access > Organization settings\)](#)

Teams and Skype for Business users in external organizations On **5**

People in my organization can communicate with unmanaged Teams accounts Off

People in my organization can communicate with users who are using custom applications built with Azure Communication Services Off

Communication with Teams and Skype for Business users from trusted organizations in group chats is limited to two orgs max Off

Save **Cancel**

7. Click the **Save** **6** button.

Configuring domain feeds

The first time you connect to a domain feed, you will use an API key provided to you by LSEG.

To do this:

1. In the **Provider key** field, enter the API key.
2. Click the **Continue** button.

API Feed: Disconnected Teams Status: Disconnected

Connect to a data provider

Provider key *

Secret key

Back Continue

3. Click **Go to Policies** to be directed to the Policies screen.

Thereafter, you may need to configure these domain feeds:

1. In ADM, select **Settings** 1.

Policies 17
6 managed • 11 unmanaged

Domains 23
5 approved • 18 pending

Logs

Settings 1

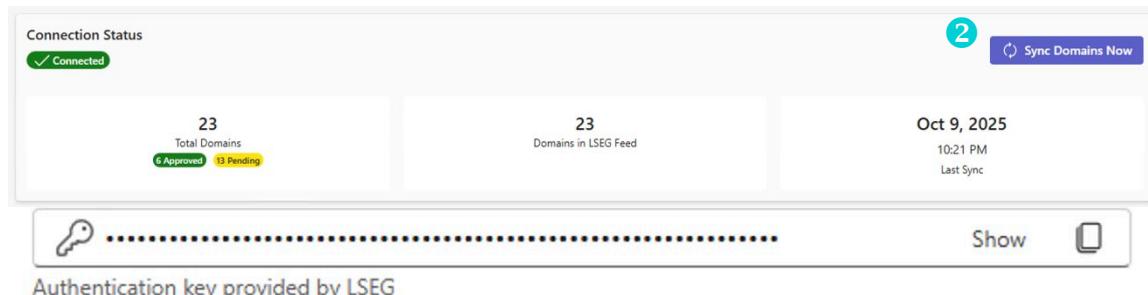
User Guide

2. In the resulting window, select **Domain Feed Configuration**.

Cloud icon Domain Feed Configuration

Configure Banking Domain Feed API endpoint, authentication, and synchronization settings

3. In the **API Configuration** section, add the API key provided by LSEG. If you have any issues, contact [LSEG Support](#).
4. Click **Save Configuration**.
5. Validate connectivity by clicking the **Sync Domains Now**  button.



The screenshot shows the 'Connection Status' section with a green 'Connected' button. Below it, 'Total Domains' is 23, with 6 Approved and 13 Pending. The 'Domains in LSEG Feed' is also 23. On the right, the 'Last Sync' was on 'Oct 9, 2025' at '10:21 PM'. At the bottom, there's a key icon and a 'Show' button, with the text 'Authentication key provided by LSEG'.

Subscribing Admin users to notifications

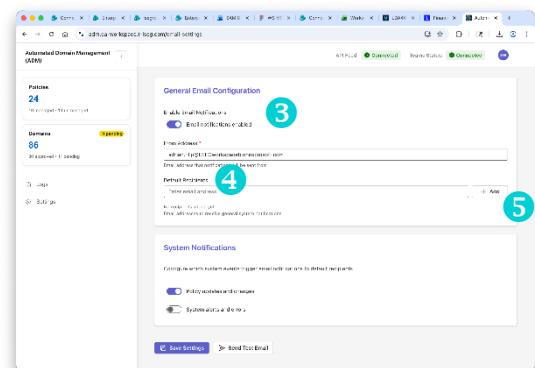
To subscribe Admin users to general system notifications:

1. In ADM, select **Settings**.
2. Select **Email Notification Settings** and ensure **Email notifications enabled**  is switched on.
3. Add the relevant email addresses in the **Default Recipients**  field and click the **Add** button .

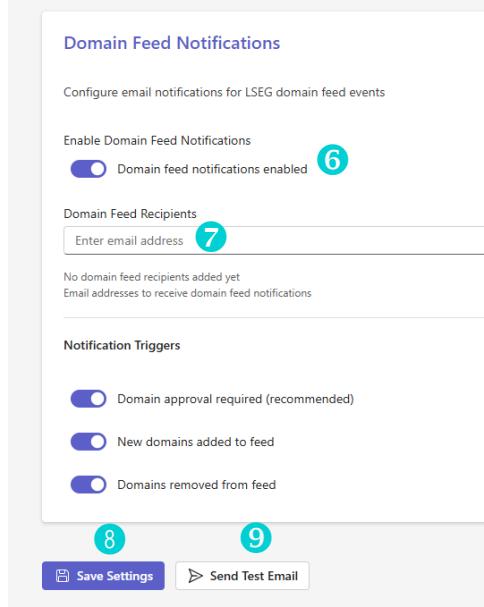
Notifying Admin users of domain changes

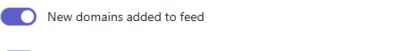
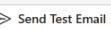
To notify Admin users of changes to the domains list:

1. In ADM, select **Settings**.
2. Select **Domain Feed Notifications** and ensure **Domain feed notifications enabled**  is switched on.
3. Add the relevant email addresses in the **Default Feed Recipients**  field.
4. Click the **Save Settings**  button and then click **Send Test Email**  to verify the setup.



The screenshot shows the 'General Email Configuration' section. It includes fields for 'Email address' (with placeholder 'Email address for domain feed notifications') and 'Default recipients' (with placeholder 'Email addresses to receive domain feed notifications'). The 'Email notifications enabled' switch is on, indicated by a green circle with a white checkmark  . The 'Save Settings' button is at the bottom .



The screenshot shows the 'Domain Feed Notifications' section. It includes a 'Domain feed notifications enabled' switch          

Managing domains

To either approve or suspend domains:

1. Select Domains ①.

The screenshot shows the ADM interface. The top navigation bar says "Automated Domain Management (ADM)". Below it, the "Policies" section shows "28" domains with "11 managed + 17 unmanaged". The "Domains" section is highlighted with a blue border, showing "61" domains with "7 approved + 50 pending".

2. Select domains in the Pending ② list.
3. Approve ③ or ④ suspend the selected domains.

The screenshot shows the "Domains" management screen. The top navigation bar has a search bar and a "Sync Feed" button. Below it, there are three tabs: "Pending" (2), "Approved" (10), and "Suspended" (4). The "Pending" tab is selected. Below the tabs are two buttons: "Approve (1)" and "Suspend (1)". The main table lists one domain: "lseg.com" (LSEG), added on "10/08/2025", with a status of "Pending 2 days".

As a result:

- The approved domain will be added to the Approved list, and these domains will be available in the domain selection in Policies Management.
- The suspended domain will be added to the Suspended list, and these domains will be removed from all ADM managed policies.

Managing policies

There are two types of policies that are relevant to ADM:

Policies managed by Teams Admin Center

- These policies are not managed in ADM. They are created by the administrator in the Teams Admin Center, and can be used as base policies for ADM managed policies. See [Creating a base policy](#) for more information.
- Each ADM managed policy needs to be mapped 1:1 to each org policy (base policy for ADM policy).
- ADM does not change the base org policy.

Policies managed by ADM

- These policies are created in ADM and can be modified within ADM.
- ADM is assigning domains, users and groups to these policies without touching anything on the org policy.

To navigate to the screen where you can manage policies for your ADM:

1. Select **Policies**.

Automated Domain Management (ADM)

Policies
28
11 managed • 17 unmanaged

Domains
61
7 approved • 50 pending

Logs

Settings

2. Click **Sync Teams** to sync the policies available for Microsoft Teams.

Policies Last sync: Just now

Managed by ADM

These Teams External Access policies are managed by ADM. To make changes, edit them here rather than in Teams Admin Center.

NAME	ASSIGNMENTS	STATUS
ADM Test Policy 2_OD External Access Policy	1 group 1 assignment total	Onboarded
ADM Test Policy 3_OD External Access Policy	1 group 1 assignment total	Onboarded

Creating a new policy

To create a new policy:

1. Click **Add New Policy** (next to the Sync Teams button).
2. The Create Policy screen appears.
3. From the **Select Base Policy (Ready to Onboard)** field, select the required base policy.
3. In the **Assign policy to groups** field, select the relevant **Security Group** ①.

Automated Domain Management (ADM)

Policies: 22 (7 managed, 15 unmanaged)

Domains: 107 (29 approved, 33 pending)

Logs

Settings

Add Policy

Create a new Teams External Access policy

Select Base Policy (Ready to Onboard): *

Important: Policy creation requires at least one group assignment for policy effectiveness. Policies without assignments are created but have no effect.

Assign policy to groups: *

1. Search for security groups...

Enable bilateral chat (limit new external group chats to two organisations)

On

2. Add Policy

Select Approved Domains

Deselect All

20 selected

lseg0.onmicrosoft.com (lseg0)

lsegworkspace0.onmicrosoft.com (lseg Workspace 0)

lsegworkspacea.onmicrosoft.com (lseg Workspace A)

wellsfargo.com (Wells Fargo)

4. Click the **Add policy** button ②.

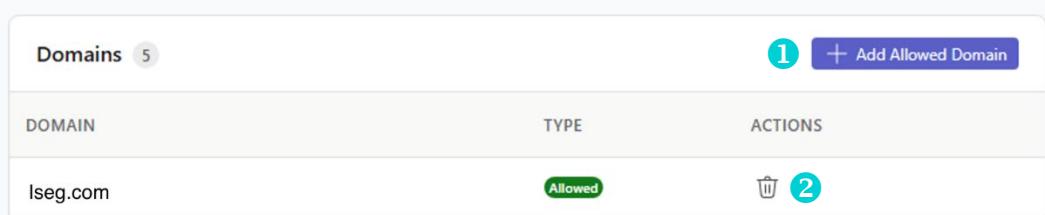
Adding or deleting a domain

To add a domain to a policy:

- Select an existing ADM managed policy and click the **Add Allowed Domain** button ①.

To delete a domain:

- Select an existing domain and click the  button ②.



Domains 5		
DOMAIN	TYPE	ACTIONS
lseg.com	Allowed	 ②

Managing a ranking

A rank refers to the priority of Teams policies applied individually to security groups or users. The lower the number of the policy ranking, the higher the priority.

To manage a ranking:

1. Click the **Manage Rank** button.

Policies



Managed by ADM

These Teams External Access policies are managed by ADM. To make changes, edit them here rather than in Teams Admin Center.

2. Adjust the rank as required.
3. Click the **Save Changes** button.

Support

If you need support during any stage of the installation and deployment process, during the preview phase you can contact us here: WSTEAMSonboarding@lsegu.com.

Appendix A: Required permissions

The following table describes the permissions that should be granted to enable seamless collaboration and personalized user experiences within the LSEG Workspace app:

Resource Type	What it allows	Why it is needed	Purpose
Chat Create	Create new 1:1 or group chats.	This is the base permission to start a new chat thread.	Enables proactive communication—users do not need to manually start a chat before sharing content.
Chat.Read.Basic	View basic information about chats (such as chat IDs and participants).	Helps the app identify existing chats or confirm chat creation.	Retrieves a list of recent chats the user has participated in to generate type-ahead suggestions for recipients.
ChatMessage.Send	Send messages into a chat.	Needed to post content into the chat after it has been created.	Deliver the message with the shared content.
openid	Enables silent single sign on (SSO). Silent SSO allows users to access Microsoft Teams without re-entering credentials by using a session cookie and Microsoft Entra ID.	Required to verify the end user's Workspace license.	Allows Workspace end users to seamlessly access Workspace Teams.
TeamsAppInstallation.ReadWriteAndConsentForChat	Allows the app and app bot to install itself into a chat before sending a message.	Ensures the app is properly set up to deliver the shared content.	Ensures the app is present in the chat, to support features such as adaptive cards or bots.
User.ReadBasic.All	Accesses basic profile information such as name and photo.	Useful for showing user details in the chat UI or suggesting contacts.	Displays user details such as name and profile photo to enhance the experience and give users confidence that they are messaging the correct person.

Example of permissions being used

👉 The scenario below illustrates how some of these permissions are utilised in a typical workflow.

Scenario: A financial analyst is researching a company in LSEG Workspace and wants to quickly share insights with a colleague via Microsoft Teams. The following workflow is initiated:

1. User clicks ‘Send via Teams’ in LSEG Workspace.
2. The app uses the **chat.create** permission to check if a 1:1 or group chat already exists.
3. If not, the app will create a new chat thread between the analyst and the recipient.
4. The app uses **chatmessage.send** to post a message with a link to the company insights.
5. If the app is not already installed in the chat, it uses **teamsappinstallation.readwriteandconsentforchat** to install itself.
6. The app may also use **user.readbasic.all** to display the recipient's name and profile picture in the UI.

Appendix B: Azure resources

The following resources will be deployed on the customer's Azure subscription during deployment of ADM:

Resource Type	Default Specification	Minimum Role Required
Action Group	Default value	Contributor or Monitoring Contributor
Application Insights	Default value	Contributor or custom role with Microsoft.Insights/components/write
Azure Database for PostgreSQL flexible server	Name:Standard_B1ms Tier:Burstable PostgreSQL version: 14.19	Contributor or DBAas Contributor
Container App	Default value	Contributor or Azure Kubernetes Service RBAC Writer
Container Apps Environment	workloadProfileType: Consumption	Contributor or custom role with Microsoft.App/managedEnvironments/write
Key vault	Family:A Name:Standard	Contributor or Key Vault Contributor or custom role with Microsoft.KeyVault/vaults/* permissions
Log Analytics workspace	Name:PerGB2018	Contributor or Log Analytics Contributor
Static Web App	Name:Free Tier:Free	Contributor or Website Contributor
Storage account	Name:Standard_LRS Tier:Standard	Contributor or Storage Account Contributor
App Registration - frontend	Graph API delegated permissions (admin consent required): <ul style="list-style-type: none"> User.Read – Required for reading user info of the current admin user who is using the app 	Contributor
App Registration - backend	Graph API Application permissions (admin consent required): <ul style="list-style-type: none"> Application.Read.All - Required for checking app consents are configured correctly Group.ReadWrite.All - Required for creating security groups to assign newly created policies GroupMember.Read.All - Required for read security group members to analyze assignments Mail.Send - required for sending email Organization.Read.All - Required for teams powershell authentication User.ReadBasic.All - Required for reading user info for individual user search 	Contributor

Appendix C: Frequently asked questions

What is the Automated Domain Management (ADM) app?

The ADM app is a management tool for Microsoft Teams administrators designed to keep external collaboration policies aligned with LSEG's Open Directory network. It automates the process of:

- Subscribing to a domain feed
- Updating federation policies
- Managing collaboration rules at scale

What problem does the ADM app solve?

To communicate externally in Microsoft Teams, organisations must federate with numerous entities in a point-to-point way. Existing workflows require manual processing, which is time-consuming and prone to error. The ADM app automates this process, reduces administrative overhead, and allows policies to remain up to date. ADM features a user-friendly front-end, robust backend services, and is deployed inside the customer's own environment to ensure that sensitive data does not leave the customer data boundary.

How does the app work?

The ADM app creates external access policies within Microsoft Teams, facilitating communication between users and other members of the Open Directory network. These policies, also referred to as external collaboration or federation policies, ensure that only specific individuals in the organisation (in other words, Open Directory users) can communicate only with other Open Directory customers, and to those within your existing federation policies. This capability is enabled by the new Granular Federation Controls feature in Microsoft Teams.

The ADM app will:

- Create new external access policies in Teams
- Synchronise created policies with:
 - Approved domains received from LSEG
 - Other, organisation-managed, policies in Teams
- Assign policies to appropriate users / groups
- Provide workflow for admins to approve / reject domains received from LSEG

The ADM app does not:

- Send tenant configuration or messaging data to LSEG
- Store tenant configuration other than for policies it manages (which it does locally in your environment)
- Edit existing organisation configuration or policies, except in very limited circumstances and with admin consent (see [What is Granular Federation Control?](#), below).

What is Granular Federation Control?

Granular Federation Control is a new feature in Microsoft Teams which enables administrators to configure different federation policies for different groups of users in their organisation.

For granular federation controls to work, the property `AllFederatedUsers` must be set to true. This is a tenant-wide setting. The ADM app will check this and inform the administrator that it must be set correctly before continuing. The administrator can do this themselves, or the ADM app can do it on their behalf. Changing this value from false to true will enable federation at the tenant-wide level. If this was set to prevent any federation within the tenant, the admin should set the `AllowedDomains` property to null.

For more information, see [Set Tenant Federation Configuration](#) and [Set External Access Policy](#) on Microsoft Learn.

How does the ADM app know which settings and domains to configure?

New policies generated by the ADM app are based upon a pre-existing policy that is managed by the organisation's administrator within the Teams Admin Centre (TAC). The ADM app continuously synchronises these new policies with the corresponding base policy. Administrators continue to update their org policies as usual, and the ADM app keeps the policies it manages aligned with any updates made by administrators to the base policy in the TAC.

Federated domains are configured on the ADM-managed policy by referencing both the original baseline policy and the list of approved domains provided by LSEG. This approach allows approved domains from LSEG to be configured, while retaining the organisation's ability to customise policies enabling communication beyond Open Directory.

ADM does not edit any organisation-managed policies, meaning that administrators can continue to manage their existing policies as usual and the ADM app will resolve any policy conflicts as per the settings chosen by the administrator.

How does the ADM app know which users should be assigned the policy?

The app knows which users should be assigned the policy once administrators have created the relevant security groups and specified users to them.

Do I have control over these policies?

Yes. Administrators can specify whether new domains added to the network by LSEG should be automatically approved and applied to their organisation's policies, or if approval is required first.

Additionally, administrators can specify 'always-block' lists that take precedence over the domain feed, ensuring critical domains remain blocked regardless of feed updates.

As policies are synced with base organisation policies, administrators continue to manage their organisation policies as usual, and changes will be replicated to the corresponding ADM-managed policy. This allows administrators to add additional domains which are not members of Open Directory, ensuring users are still able to communicate with these organisations.

How does the ADM app handle security and compliance?

The ADM app leverages Microsoft Azure's platform-managed services to allow high availability, security, and compliance. It supports secure authentication with Entra ID and maintains comprehensive audit logs. The ADM app requires a service principal in your Entra tenant so that it can connect to the Teams PowerShell service. This service principal requires the Teams Administrator role.

The ADM app also requires administrators to create an app registration in Entra to enable SSO to the management portal.

What technology does this use and how is it deployed?

The solution uses Azure platform-managed services, which provide native high availability without requiring custom application-level high availability logic.

The ADM app will have dependencies on the following Azure platform services:

- Azure Container Apps
- Azure Static Web Apps
- Key Vault
- Application Insights
- PostgreSQL

Services such as Azure App Service, Function Apps, Key Vault, and Entra ID are inherently resilient and distributed, ensuring uptime through zone and regional redundancy.

For disaster recovery, the solution is designed to be re-deployable by administrators in alignment with their specific disaster recovery requirements. Infrastructure-as-Code templates support rapid provisioning in alternate regions, while data services use

geo-redundant configurations to protect against regional failures. Key Vault secrets and configuration settings can be replicated across vaults, and monitoring via Application Insights allows visibility and supports proactive recovery actions.

Deploying the ADM app requires provisioning infrastructure in Azure, which can be automated using Azure Resource Manager (ARM) and PowerShell deployment scripts.

Customers are free to tailor the solution per their requirements, for example selecting alternative high availability / disaster recovery options, scaling, network connectivity, load balancing, and so on.

© 2026 LSEG. Republication or redistribution of LSEG content, including by framing or similar means, is prohibited without the prior written consent of LSEG. LSEG is not liable for any errors or delays in LSEG content, or for any actions taken in reliance on such content. LSEG Data & Analytics logo is a trademark of LSEG and its affiliated companies.

lseg.com

LSEG DATA &
ANALYTICS

Document version: 100.04