

Create a new Enterprise Application:

We need to create an app registration. This is an app within Azure that can be authenticated via a secret, which we can grant permissions to certain things in Smartspace. Giving you an authentication method for non-users that you can hide in a serverless function.

Step 1) In Microsoft Entra ID, create the App Registration

The screenshot shows the Microsoft Entra ID Overview page for the application "smartspace.ai". The left sidebar has a "Overview" tab selected. The main content area is titled "Basic information". A navigation menu on the right includes options like "User", "Group", "Enterprise application", and "App registration", with "App registration" currently highlighted. The URL in the browser is https://entra.microsoft.com/~/myapps/overview?p=smartspace.ai

Step 2) Give it a name, and “Register” with the default options.

Step 3) Write down the Application Client ID and the Application Object ID:

Display name	:	Smartspace External Integration App	Client credentials	:	Add a certificate or secret
Application (client) ID	:	[REDACTED]	Redirect URIs	:	Add a Redirect Add a certificate or secret
Object ID	:	[REDACTED]	Application ID URI	:	Add an Application ID URI
Directory (tenant) ID	:	[REDACTED]	Managed application in L...	:	Smartspace External Integration App
Supported account types : My organization only					

Step 4) Click on “Add a certificate or secret” and create a new client secret and write down the value for the new secret

The screenshot shows the 'Certificates & secrets' section of the Azure portal. On the left, there's a sidebar with links like Overview, Quickstart, Integration assistant, Diagnose and solve problems, Manage (with sub-links for Branding & properties, Authentication (Preview), Certificates & secrets, Token configuration, API permissions, Expose an API, App roles, and Owners). The 'Certificates & secrets' link is highlighted. The main area has a heading 'Client secrets (1)'. Below it, a note says 'A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as application key or password.' There's a button '+ New client secret'. A table lists one client secret: 'New Secret' with an expiration date of '6/15/2026'. The 'Value' column contains a redacted string.

Home > smartspace.ai | Enterprise applications > Enterprise applications

The screenshot shows the 'All applications' section of the Azure portal. The sidebar includes links for Overview, Manage (with sub-links for All applications, Private Network connectors, User settings, App launchers, Custom authentication extensions, Security, Activity, and Troubleshooting + Support), and a search bar. The 'All applications' link is highlighted. A note says 'Agent ID (Preview) has been moved to the Agent ID experience. View agent identities.' The main area shows a search bar with 'Stefan' and a filter 'Application type == Enterprise Applications'. It displays a table with one result: 'StefansTestReg' with columns for Name, Object ID, Application ID, and Homepage URL (both redacted).

Step 5) Use the Azure CLI to find the role id of the “Integrations” role in the Smartspace app, replace **SmartspaceAppName** with the name of the smartspace app that was chosen at install time. If you have lost the name, you can find the Client ID for this application in your admin portal under the Admin API documentation -> CLI Setup. You can use this client ID to find the application under enterprise applications. Save the IntegrationsAppRoleId for later.

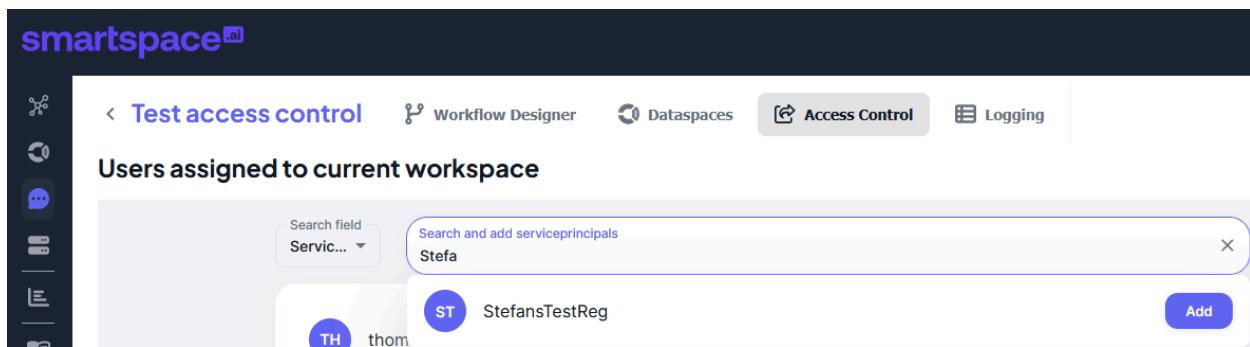
```
az ad sp list --filter "displayName eq 'SmartspaceAppName'" --query  
"[].{SmartSpaceEnterpriseAppObjectId:id,  
IntegrationsAppRoleId:appRoles[?displayName == 'Integrations'].id | [0]}"
```

```
PS /home/stefan> az ad sp list --filter "displayName eq 'app-w...  
[  
 {  
   "IntegrationsAppRoleId": [REDACTED]  
   "SmartSpaceEnterpriseAppObjectId": [REDACTED]  
 }  
 ]
```

Step 6) Use the Azure CLI to assign that role, to the your new app:

```
Connect-AzureAD  
New-AzureADServiceAppRoleAssignment `  
    -ObjectId      'your app object id' `  
    -PrincipalId  'your app object id' `  
    -ResourceId   'smartspace app object id' `  
    -Id           'IntegrationsAppRoleId'
```

Step 7) Assign workspace permissions to this app in the Smartspace Admin Portal by going to the workspace > Access control > Search for service Principal and add:



The screenshot shows the Smartspace Admin Portal interface. At the top, there's a navigation bar with icons for Test access control, Workflow Designer, Dataspaces, Access Control (which is currently selected), and Logging. Below the navigation bar, the page title is "smartspace". Under the "Access Control" tab, there's a sub-section titled "Users assigned to current workspace". A search bar is present with the placeholder "Search and add serviceprincipals" and a dropdown menu showing the entry "Stefa". Below the search bar, a list of users is shown, including "thom" and "StefansTestReg". An "Add" button is located in the bottom right corner of the user list area.

Now you have an app that you can use to authenticate with Smartspace and interact with this workspace. I've attached a python script that shows how you can authenticate and send a basic message to the Smartspace workspace.