

Lesson 3- Securing Your Azure OpenAI Credentials with API Keys, Key Vault, and .NET Secrets



Microsoft
Agent Framework

<https://github.com/microsoft/agent-framework/tree/main/dotnet/samples>



Microsoft
Agent Framework

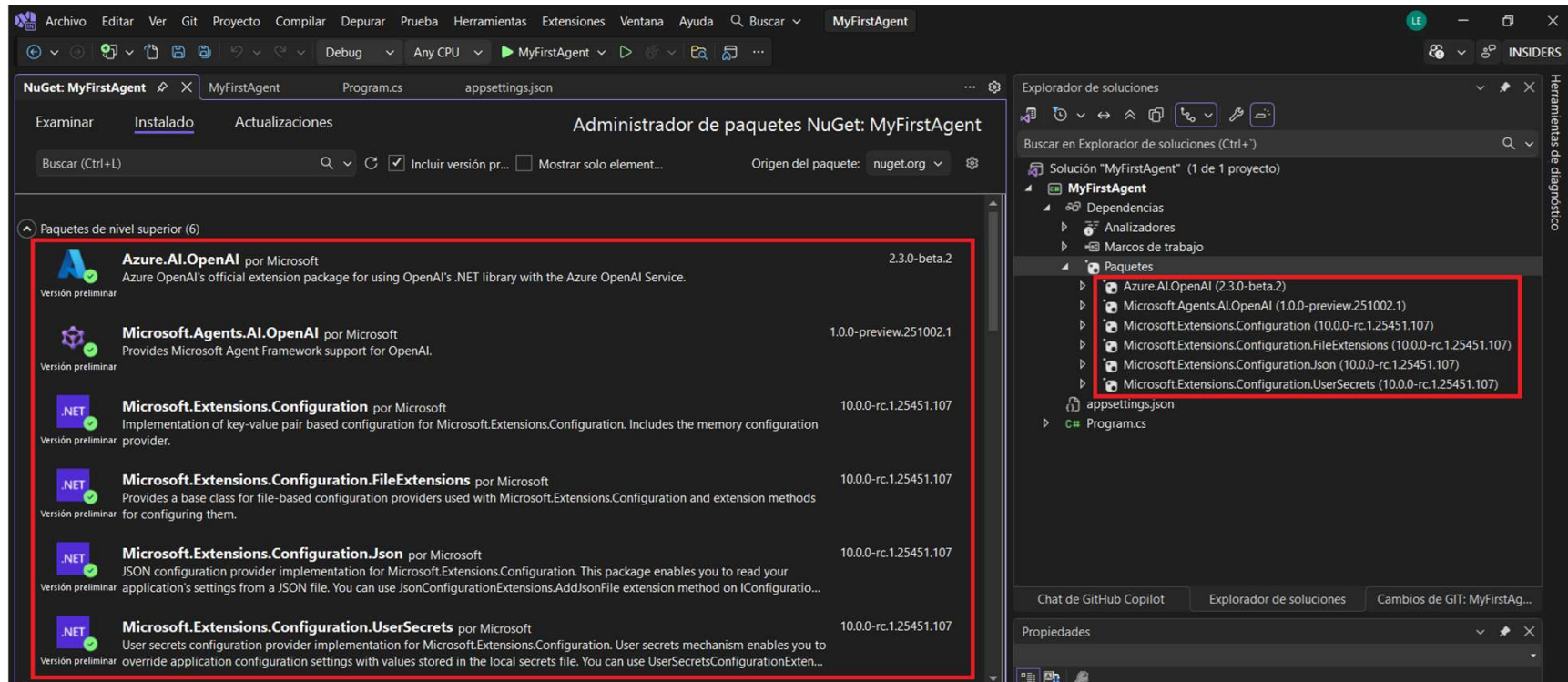
Create a C# Console Application

<https://github.com/microsoft/agent-framework/tree/main/dotnet/samples/GettingStarted/Agents>

Load NuGet Packages



Microsoft
Agent Framework



<https://github.com/microsoft/agent-framework/tree/main/dotnet/samples/GettingStarted/Agents>

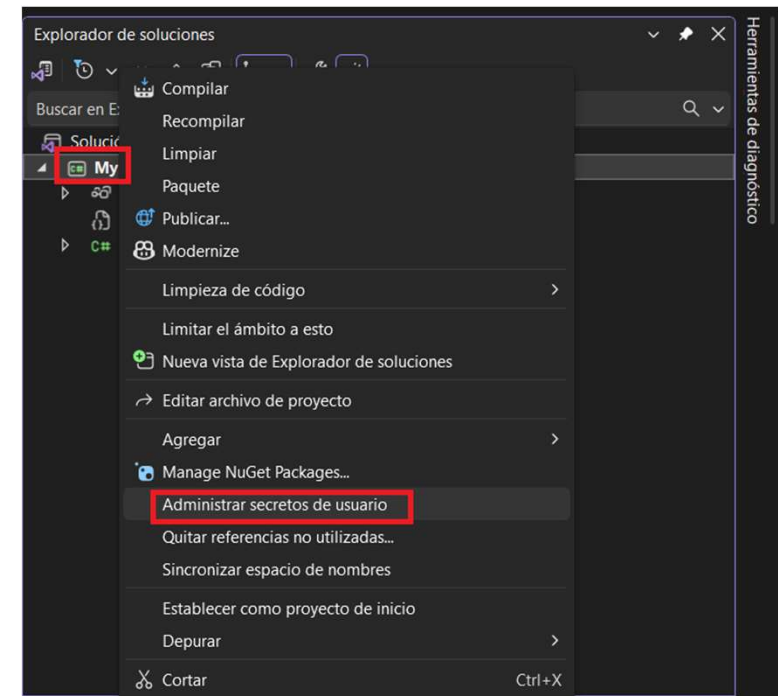
Securing with .NET Secrets



Microsoft
Agent Framework

secrets.json

```
secrets.json
Esquema: <No se seleccionó ningún esquema>
1  {
2    "AzureOpenAI": {
3      "ApiKey": "Neer"
4    }
5  }
```



<https://github.com/microsoft/agent-framework/tree/main/dotnet/samples/GettingStarted/Agents>



Microsoft
Agent Framework

Securing with .NET Secrets

appsettings.json

```
appsettings.json ✕  
Esquema: https://www.schemastore.org/appsettings.json  
1  {  
2  "AzureOpenAI": {  
3    "Endpoint": "https://myaiserviceluisccoco.openai.azure.com/",  
4    "DeploymentName": "gpt-4o-mini"  
5  }  
6  }
```

<https://github.com/microsoft/agent-framework/tree/main/dotnet/samples/GettingStarted/Agents>



Microsoft
Agent Framework

Securing with .NET Secrets

program.cs

```
using Azure; // AzureKeyCredential
using Azure.AI.OpenAI;
using Microsoft.Agents.AI;
using Microsoft.Extensions.Configuration;
using OpenAI;

var configuration = new ConfigurationBuilder()
    .SetBasePath(Directory.GetCurrentDirectory())
    .AddJsonFile("appsettings.json", optional: false, reloadOnChange: true) // Endpoint & DeploymentName
    .AddUserSecrets<Program>(optional: false) // ApiKey from user secrets
    .Build();

// Read non-secrets from appsettings.json
var endpoint = configuration["AzureOpenAI:Endpoint"]
    ?? throw new InvalidOperationException("Missing AzureOpenAI:Endpoint in appsettings.json.");

var deploymentName = configuration["AzureOpenAI:DeploymentName"] ?? "gpt-4o-mini";

// Read secret from .NET User Secrets
var apiKey = configuration["AzureOpenAI:ApiKey"]
    ?? throw new InvalidOperationException("Missing AzureOpenAI:ApiKey in user secrets.");
```

<https://github.com/microsoft/agent-framework/tree/main/dotnet/samples/GettingStarted/Agents>



Microsoft
Agent Framework

Securing with .NET Secrets

program.cs

```
// Create agent using API key auth (not AAD)
const string JokerName = "Joker";
const string JokerInstructions = "You are good at telling jokes.";

var AIAgent agent = new AzureOpenAIClient(
    new Uri(endpoint),
    new AzureKeyCredential(apiKey))
    .GetChatClient(deploymentName)
    .CreateAIAgent(JokerInstructions, JokerName);

// Use the agent
Console.WriteLine(await agent.RunAsync("Tell me a joke about a pirate.));

var await foreach (var chunk in agent.RunStreamingAsync("Another pirate joke, please.))
{
    Console.WriteLine(chunk);
}
```

<https://github.com/microsoft/agent-framework/tree/main/dotnet/samples/GettingStarted/Agents>



Microsoft
Agent Framework

Securing with .NET Secrets

MyFirstAgent.csproj

```
<Project Sdk="Microsoft.NET.Sdk">

  <PropertyGroup>
    <OutputType>Exe</OutputType>
    <TargetFramework>net10.0</TargetFramework>
    <ImplicitUsings>enable</ImplicitUsings>
    <Nullable>enable</Nullable>
    <UserSecretsId>dbc6a46c-8812-48bc-8b2f-4d7f471c8552</UserSecretsId>
  </PropertyGroup>

  <!-- Ensure appsettings.json is copied to the build output -->
  <ItemGroup>
    <None Update="appsettings.json">
      <CopyToOutputDirectory>PreserveNewest</CopyToOutputDirectory>
    </None>
  </ItemGroup>

  <ItemGroup>
    <PackageReference Include="Azure.AI.OpenAI" Version="2.3.0-beta.2" />
    <PackageReference Include="Microsoft.Agents.AI.OpenAI" Version="1.0.0-preview.251002.1" />
    <PackageReference Include="Microsoft.Extensions.Configuration" Version="10.0.0-rc.1.25451.107" />
    <PackageReference Include="Microsoft.Extensions.Configuration.FileExtensions" Version="10.0.0-rc.1.25451.107" />
    <PackageReference Include="Microsoft.Extensions.Configuration.Json" Version="10.0.0-rc.1.25451.107" />
    <PackageReference Include="Microsoft.Extensions.Configuration.UserSecrets" Version="10.0.0-rc.1.25451.107" />
  </ItemGroup>

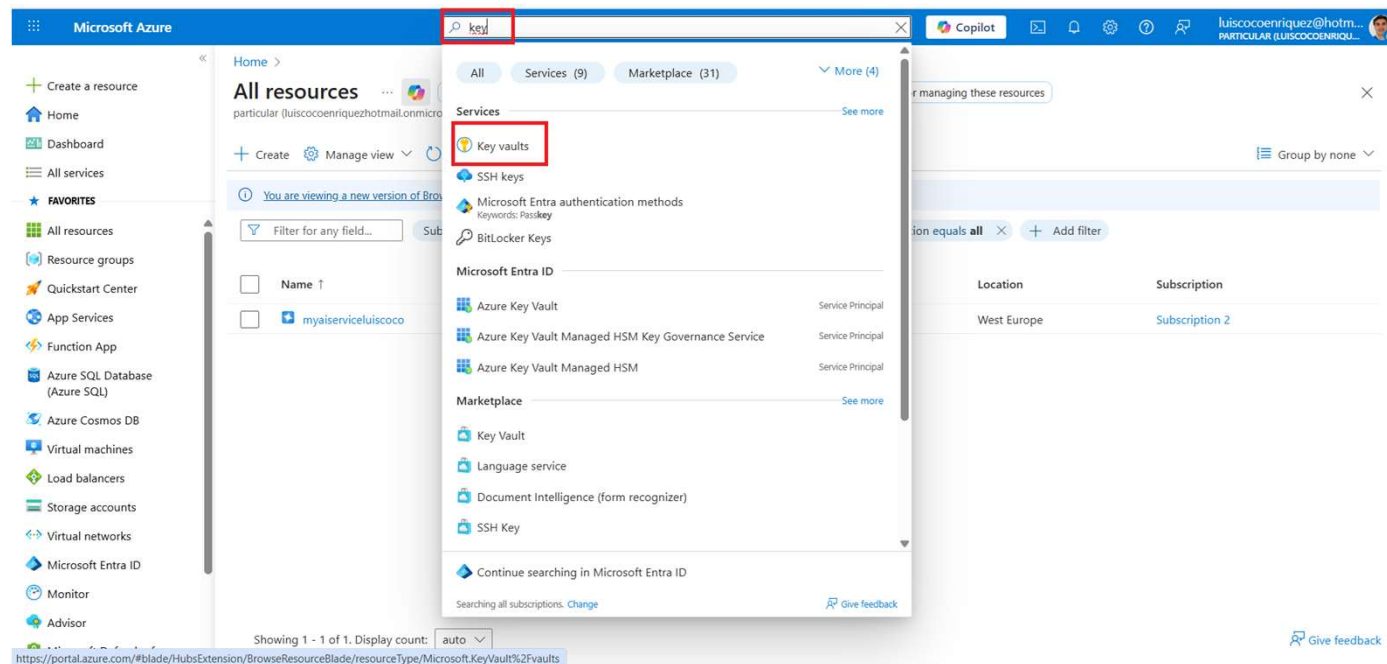
</Project>
```

<https://github.com/microsoft/agent-framework/tree/main/dotnet/samples/GettingStarted/Agents>

Securing with Azure KeyVault service



We search for the Azure KeyVault Service



<https://github.com/microsoft/agent-framework/tree/main/dotnet/samples/GettingStarted/Agents>

Securing with Azure KeyVault service



We create a new Azure KeyVault service

A screenshot of the Microsoft Azure portal interface. The left sidebar shows the navigation menu with "Key vaults" highlighted. The main content area shows the "Key vaults" page with a red box around the "Key vaults" title. Below the title, there are search filters and a "Create" button, which is also highlighted with a red box. The page displays "No key vaults to display" and a "Create" button. The "Create" button is a blue button with a white plus sign and the word "Create". Below it is a "Learn more" link.

<https://github.com/microsoft/agent-framework/tree/main/dotnet/samples/GettingStarted/Agents>

Securing with Azure KeyVault service



We input the KeyVault parameters

A screenshot of the Azure portal's "Create a key vault" wizard. The interface shows a left-hand navigation pane with various Azure services, a top search bar, and a main content area. The "Create a key vault" wizard is active, with tabs for "Basics", "Access configuration", "Networking", "Tags", and "Review + create". The "Basics" tab is selected, showing a description of Azure Key Vault and a "Project details" section. The "Project details" section includes a "Subscription" dropdown (set to "Subscription 2"), a "Resource group" dropdown (set to "myRG" with a "Create new" link), and an "Instance details" section. The "Instance details" section includes a "Key vault name" field (set to "myKeyVaultLuisCoco" with a green checkmark) and a "Region" dropdown (set to "West Europe"). At the bottom of the wizard, there are "Previous", "Next", and "Review + create" buttons.

Microsoft Azure

Search resources, services, and docs (G+ /)

Home > Key vaults >

Create a key vault

Basics Access configuration Networking Tags Review + create

Azure Key Vault is a cloud service used to manage keys, secrets, and certificates. Key Vault eliminates the need for developers to store security information in their code. It allows you to centralize the storage of your application secrets which greatly reduces the chances that secrets may be leaked. Key Vault also allows you to securely store secrets and keys backed by Hardware Security Modules or HSMs. The HSMs used are Federal Information Processing Standards (FIPS) 140-2 Level 2 validated. In addition, key vault provides logs of all access and usage attempts of your secrets so you have a complete audit trail for compliance.

Project details

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

Subscription * Subscription 2

Resource group * myRG [Create new](#)

Instance details

Key vault name * myKeyVaultLuisCoco ✓

Region * West Europe

Previous Next Review + create

<https://github.com/microsoft/agent-framework/tree/main/dotnet/samples/GettingStarted/Agents>

Securing with Azure KeyVault service



We copy the **Azure KeyVault URI** to input in the **appsettings.json** file in your C# console application

A screenshot of the Azure Portal interface. The left sidebar shows the navigation menu with 'Overview' selected. The main content area displays the 'Overview' page for a Key Vault named 'mykeyvaultcocoenriquez'. The 'Essentials' section shows the Vault URI as 'https://mykeyvaultcocoenriquez.vault.azure.net/'. The 'Tags' section shows 'Add tags'. The 'Get started' tab is active, displaying instructions on how to use a vault per application per environment.

Microsoft Azure

Search resources, services, and docs (G+)

Copilot

luiscoenriquez@hotmail.com

mykeyvaultcocoenriquez

Key vault

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Access policies

Resource visualizer

Events

Objects

Keys

Secrets

Certificates

Essentials

Resource group (move) : myRG

Location : West Europe

Subscription (move) : Subscription 2

Subscription ID : e5bd93f3-dcd9-4833-a589-82e16245997c

Vault URI : https://mykeyvaultcocoenriquez.vault.azure.net/

Sku (Pricing tier) : Standard

Directory ID : e099cebd-5eea-41a3-88db-bcb9a9c8a83e

Directory Name : particular

Soft-delete : Enabled

Purge protection : Disabled

Tags (edit) : Add tags

Get started Properties Monitoring Tools + SDKs Tutorials

Manage keys and secrets used by apps and services

Our recommendation is to use a vault per application per environment (Development, Pre-Production and Production). This helps you to not share secrets across environments and also reduces the threat in case of a breach.

<https://github.com/microsoft/agent-framework/tree/main/dotnet/samples/GettingStarted/Agents>



Microsoft
Agent Framework

Securing with Azure KeyVault service

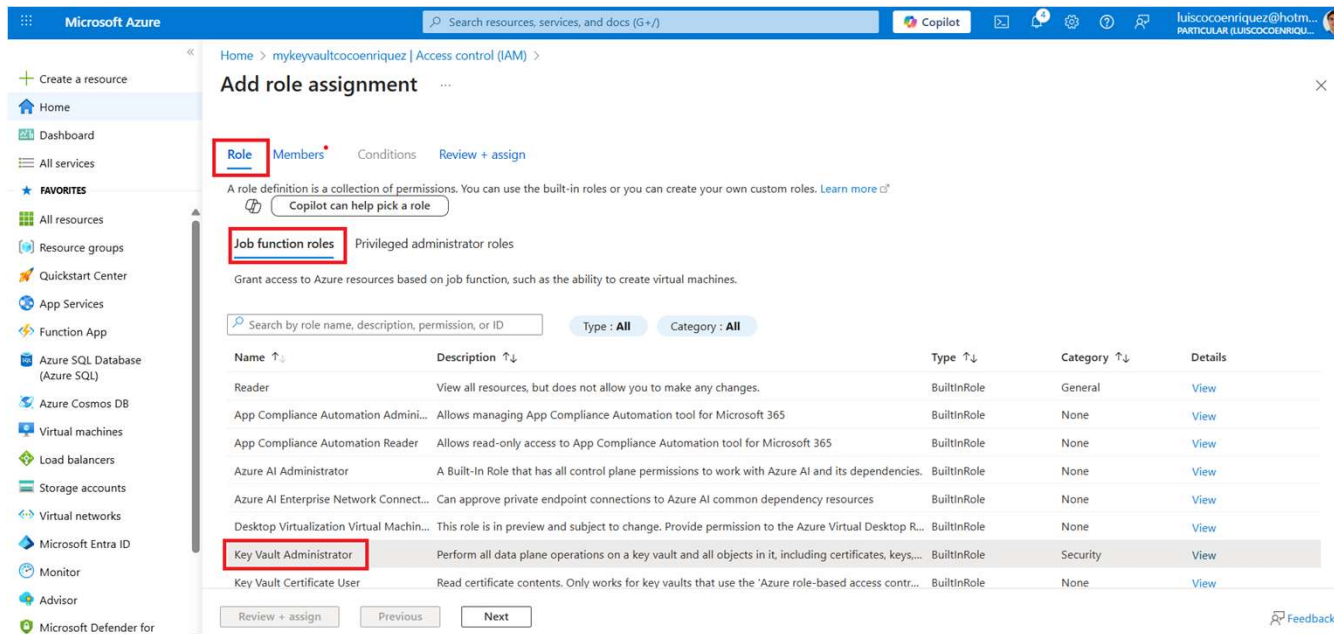
We grant permission to the KeyVault for creating a new Secret

The screenshot displays the Azure portal interface for managing access to a Key Vault resource. The left sidebar shows the navigation menu with 'Access control (IAM)' highlighted. The main pane shows the 'Access control (IAM)' page for the resource 'mykeyvaultcocoenriquez'. The 'Key vault' label is highlighted in the breadcrumb. The 'Access control (IAM)' tab is selected. The 'My access' section shows the user's level of access. The 'Check access' section provides a way to review access levels. The 'Grant access to this resource' section includes a red box around the 'Add role assignment' button. The 'View access to this resource' and 'View deny assignments' sections each have a 'View' button.

<https://github.com/microsoft/agent-framework/tree/main/dotnet/samples/GettingStarted/Agents>

Securing with Azure KeyVault service

We select the KeyVault Administrator



Microsoft Azure

Search resources, services, and docs (G+/I)

Copilot

luiscoenriquez@hotmail...
PARTICULAR (LUISCOCENRIQU...)

Home > mykeyvaultcoenriquez | Access control (IAM)

Add role assignment

Role Members Conditions Review + assign

A role definition is a collection of permissions. You can use the built-in roles or you can create your own custom roles. [Learn more](#)

[Copilot can help pick a role](#)

Job function roles Privileged administrator roles

Grant access to Azure resources based on job function, such as the ability to create virtual machines.

Search by role name, description, permission, or ID

Type: All Category: All

Name	Description	Type	Category	Details
Reader	View all resources, but does not allow you to make any changes.	BuiltInRole	General	View
App Compliance Automation Admini...	Allows managing App Compliance Automation tool for Microsoft 365	BuiltInRole	None	View
App Compliance Automation Reader	Allows read-only access to App Compliance Automation tool for Microsoft 365	BuiltInRole	None	View
Azure AI Administrator	A Built-In Role that has all control plane permissions to work with Azure AI and its dependencies.	BuiltInRole	None	View
Azure AI Enterprise Network Connect...	Can approve private endpoint connections to Azure AI common dependency resources	BuiltInRole	None	View
Desktop Virtualization Virtual Machin...	This role is in preview and subject to change. Provide permission to the Azure Virtual Desktop R...	BuiltInRole	None	View
Key Vault Administrator	Perform all data plane operations on a key vault and all objects in it, including certificates, keys...	BuiltInRole	Security	View
Key Vault Certificate User	Read certificate contents. Only works for key vaults that use the 'Azure role-based access contr...	BuiltInRole	None	View

Review + assign Previous Next

Feedback

<https://github.com/microsoft/agent-framework/tree/main/dotnet/samples/GettingStarted/Agents>

Securing with Azure KeyVault service



We select the KeyVault Administrator

The screenshot shows the Microsoft Azure portal interface. The main pane is titled "Add role assignment" and has tabs for "Role", "Members", "Conditions", and "Review + assign". The "Members" tab is selected. Under "Selected role", "Key Vault Administrator" is chosen. Under "Assign access to", "User, group, or service principal" is selected. The "Members" section has a "+ Select members" button highlighted with a red box. Below it is a table with columns "Name", "Object ID", and "Type", currently showing "No members selected". The "Description" field is optional. On the right, a "Select members" dialog is open, showing a search bar and a list of members. One member, "luis coco enriquez(Guest)", is highlighted with a red box. Below the list, the "Selected members:" section shows the same user. At the bottom of the dialog, the "Select" button is highlighted with a red box.

<https://github.com/microsoft/agent-framework/tree/main/dotnet/samples/GettingStarted/Agents>

Securing with Azure KeyVault service



We review and assign the role to the Azure KeyVault service

The screenshot shows the 'Add role assignment' page in the Azure portal. The breadcrumb path is 'Home > mykeyvaultcocoenriquez | Access control (IAM)'. The page has tabs for 'Role', 'Members', 'Conditions', and 'Review + assign'. The 'Members' tab is active. Under 'Selected role', 'Key Vault Administrator' is chosen. Under 'Assign access to', 'User, group, or service principal' is selected. The 'Members' section shows a table with one member: 'luis coco enriquez(Guest)' with Object ID '40520058-0714-4e1f-a880-5f33776e5efe' and Type 'User'. The 'Description' field is optional. At the bottom, the 'Review + assign' button is highlighted.

Name	Object ID	Type
luis coco enriquez(Guest)	40520058-0714-4e1f-a880-5f33776e5efe	User

<https://github.com/microsoft/agent-framework/tree/main/dotnet/samples/GettingStarted/Agents>



Microsoft
Agent Framework

Securing with Azure KeyVault service

We now can create the **Secret** to store the Azure OpenAI API Key inside the Azure KeyVault

Microsoft Azure

Search resources, services, and docs (G+)

Copilot

luiscoenriquez@hotmail...
PARTICULAR (LUISCOENRIQU...)

Home > mykeyvaultcocoenriquez

mykeyvaultcocoenriquez | Secrets

Key vault

Search

+ Generate/Import Refresh Restore Backup Manage deleted secrets View sample code

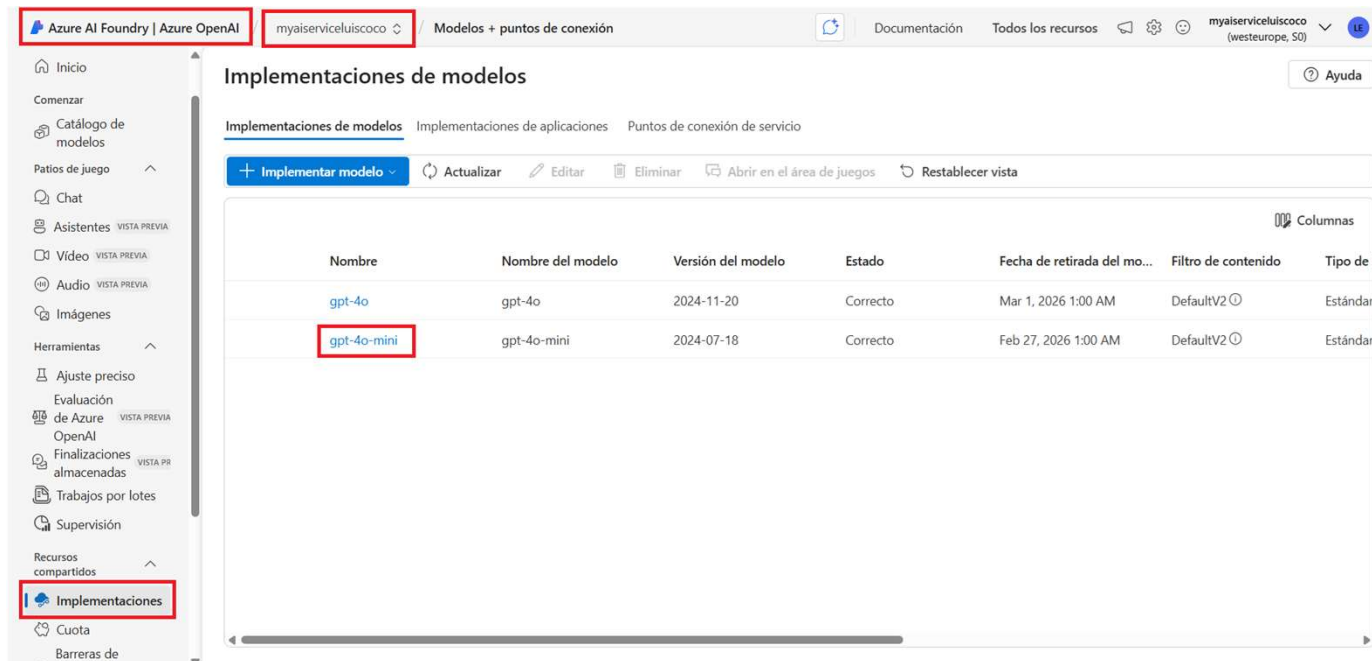
Name	Type	Status	Expiration date
There are no secrets available.			

Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems
Access policies
Resource visualizer
Events
Objects
Keys
Secrets
Certificates

<https://github.com/microsoft/agent-framework/tree/main/dotnet/samples/GettingStarted/Agents>

Securing with Azure KeyVault service

We search for the **Azure OpenAI APIKey** in **Azure Foundry**



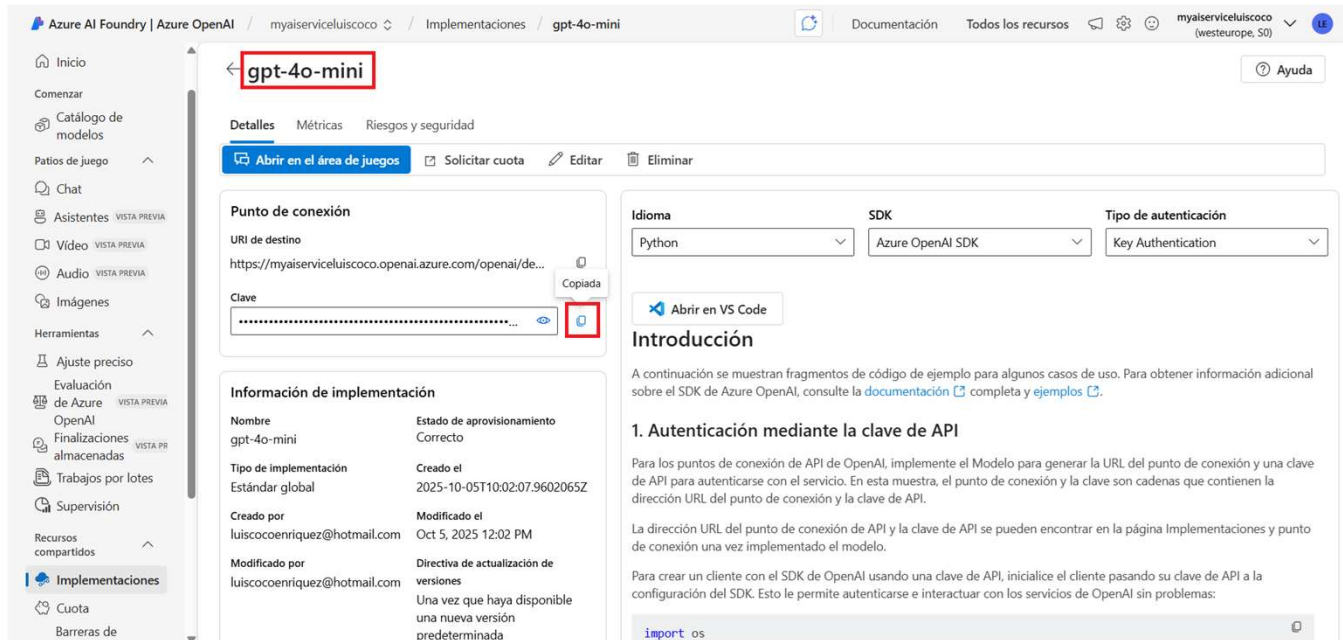
The screenshot shows the Azure Foundry console interface. The top navigation bar includes 'Azure AI Foundry | Azure OpenAI', a user profile 'myaiserviceluisccoco', and a dropdown menu 'Modelos + puntos de conexión'. The left sidebar contains various navigation options, with 'Implementaciones' highlighted. The main content area is titled 'Implementaciones de modelos' and displays a table of model implementations.

Nombre	Nombre del modelo	Versión del modelo	Estado	Fecha de retirada del mo...	Filtro de contenido	Tipo de i
gpt-4o	gpt-4o	2024-11-20	Correcto	Mar 1, 2026 1:00 AM	DefaultV2 ⓘ	Estándar
gpt-4o-mini	gpt-4o-mini	2024-07-18	Correcto	Feb 27, 2026 1:00 AM	DefaultV2 ⓘ	Estándar

<https://github.com/microsoft/agent-framework/tree/main/dotnet/samples/GettingStarted/Agents>

Securing with Azure KeyVault service

We copy the Azure OpenAI **APIKey** Value



The screenshot shows the Azure AI Foundry console for the 'gpt-4o-mini' deployment. The 'Punto de conexión' (Connection Point) section is visible, showing the 'Clave' (Key) field. The 'Clave' field is highlighted with a red box, and the 'Copiar' (Copy) button is also highlighted with a red box. The 'Información de implementación' (Deployment Information) section shows the deployment name 'gpt-4o-mini' and its status 'Correcto'.

Información de implementación	
Nombre	gpt-4o-mini
Estado de aprovisionamiento	Correcto
Tipo de implementación	Estándar global
Creado el	2025-10-05T10:02:07.9602065Z
Creado por	luiscoenriquez@hotmail.com
Modificado el	Oct 5, 2025 12:02 PM
Modificado por	luiscoenriquez@hotmail.com
Directiva de actualización de versiones	Una vez que haya disponible una nueva versión predeterminada

<https://github.com/microsoft/agent-framework/tree/main/dotnet/samples/GettingStarted/Agents>

Securing with Azure KeyVault service



Microsoft
Agent Framework

We paste the **Azure OpenAI APIKey** value into the **Secret value**

The screenshot shows the 'Create a secret' form in the Azure portal. The 'Name' field is filled with 'my-Azure-OpenAI-API-Key'. The 'Secret value' field is highlighted with a red box and contains a masked value. The 'Expiration date' is set to 10/07/2027. The 'Create' button is highlighted with a red box.

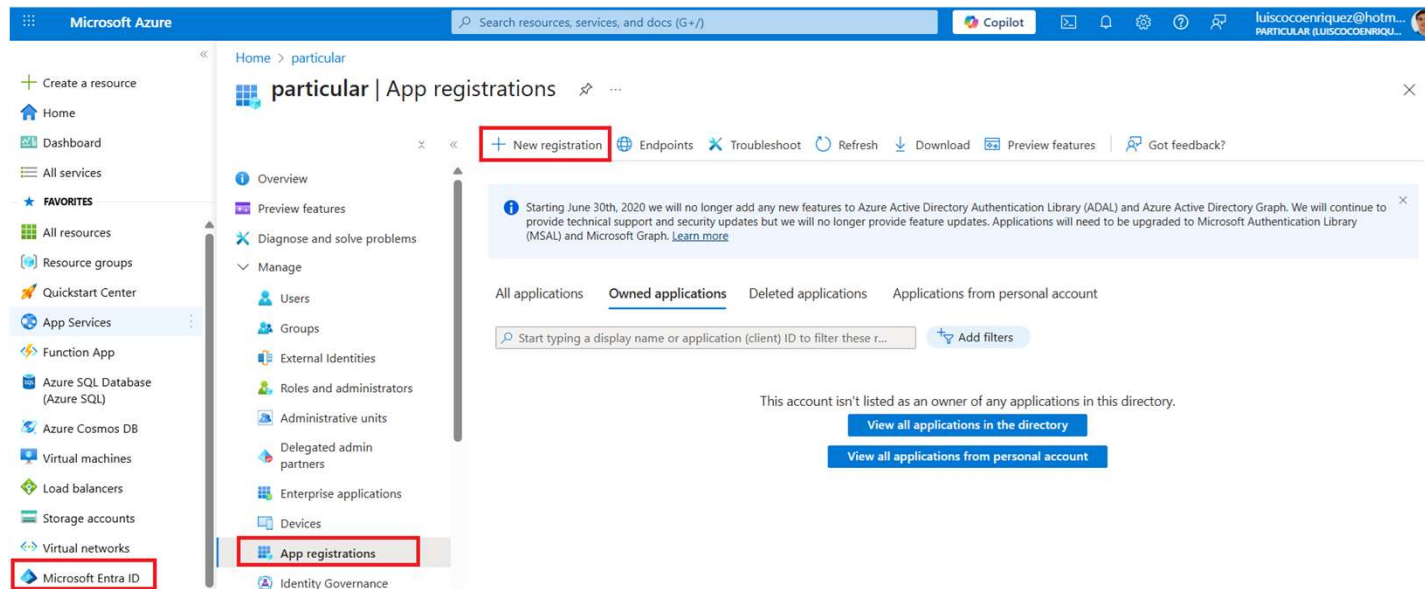
Field	Value
Upload options	Manual
Name *	my-Azure-OpenAI-API-Key
Secret value *
Content type (optional)	
Set activation date	<input type="checkbox"/>
Set expiration date	<input checked="" type="checkbox"/>
Expiration date	10/07/2027 4:33:53 PM
Enabled	Yes
Tags	0 tags

<https://github.com/microsoft/agent-framework/tree/main/dotnet/samples/GettingStarted/Agents>

Securing with Azure KeyVault service



We Register a new Application in Microsoft Entra ID



<https://github.com/microsoft/agent-framework/tree/main/dotnet/samples/GettingStarted/Agents>

Securing with Azure KeyVault service



We input the new application information

A screenshot of the Microsoft Azure portal's "Register an application" page. The page is titled "Register an application" and has a breadcrumb trail "Home > particular | App registrations >". The "Name" field is filled with "mynewapplicationluiscoco" and is highlighted with a red box. The "Supported account types" section has the option "Accounts in this organizational directory only (particular only - Single tenant)" selected, also highlighted with a red box. The "Redirect URI (optional)" section has a dropdown menu set to "Select a platform" and a text box containing "e.g. https://example.com/auth". At the bottom, there is a blue "Register" button highlighted with a red box. The left sidebar shows the Azure portal navigation menu with various services listed.

<https://github.com/microsoft/agent-framework/tree/main/dotnet/samples/GettingStarted/Agents>

Securing with Azure KeyVault service



We copy the ClientID and the Tenant values to create environmental variables

The screenshot shows the Azure Portal interface for an application registration named "mynewapplicationluiscoco". The "App registrations" tab is selected, and the "Overview" page is displayed. The "Application (client) ID" and "Directory (tenant) ID" are highlighted with red boxes. The "Variables de entorno" (Environment Variables) window is open, showing the "Variables de usuario para luisc" (User variables for luisc) and "Variables de sistema" (System variables) tabs. The "AZURE_CLIENT_ID" and "AZURE_TENANT_ID" variables are being created or edited, with their values copied from the Azure Portal. The "AZURE_CLIENT_ID" value is "33cb6c0-0bb4-4a4b-a740-793f93ede173" and the "AZURE_TENANT_ID" value is "e099cebd-5eea-41a3-88db-bcb9a9c8a83e".

Microsoft Azure

Home > particular > App registrations >

mynewapplicationluiscoco

Search resources, services, and docs (G+/I)

Home

Dashboard

All services

FAVORITES

All resources

Resource groups

Quickstart Center

App Services

Function App

Azure SQL Database (Azure SQL)

Azure Cosmos DB

Virtual machines

Load balancers

Storage accounts

Virtual networks

Microsoft Entra ID

Monitor

Overview

Quickstart

Integration assistant

Diagnose and solve problems

Manage

Branding & properties

Authentication

Certificates & secrets

Token configuration

API permissions

Expose an API

App roles

Owners

Roles and administrators

Manifest

Essentials

Display name

mynewapplicationluiscoco

Application (client) ID

33cb6c0-0bb4-4a4b-a740-793f93ede173

Object ID

577f56dd-21c1-4888-92d2-479ccfd762ee

Directory (tenant) ID

e099cebd-5eea-41a3-88db-bcb9a9c8a83e

Supported account types

My organization only

Welcome to the new and improved App registrations. Looking to learn how it's changed from App registrations (Legacy)? [Learn more](#)

Starting June 30th, 2020 we will no longer add any new features to Azure Active Directory Authentication Library (ADAL) and Azure Active Directory Graph. We will continue to provide technical support and security updates but we will no longer provide feature updates. Applications will need to be upgraded to Microsoft Authentication Library (MSAL) and Microsoft Graph. [Learn more](#)

Get Started

Documentation

Propiedades del sistema

Nombre de equipo

Opciones avanzadas

Protección del sistema

Hardware

Acceso remoto

Para realizar la mayoría de estos cambios, inicie sesión como administrador.

Rendimiento

Efectos visuales, programación del procesador, uso de memoria y memoria virtual

Perfiles de usuario

Configuración del escritorio correspondiente al inicio de sesión

Inicio y recuperación

Inicio del sistema, errores del sistema e información de depuración

Variables de entorno...

Variables de entorno

Variable

Valor

ANDROID_HOME

C:\Users\luisc\AppData\Local\Android\Sdk

AZURE_CLIENT_ID

33cb6c0-0bb4-4a4b-a740-793f93ede173

AZURE_CLIENT_SECRET

oGp8Q-gkflgHRi--pT5yXFRtY5-inYmH54bNW

AZURE_OPENAI_DEPLOYM...

gpt-4o-mini

AZURE_OPENAI_ENDPOINT

https://myaiservice.luiscoco.openai.azure.com/

AZURE_TENANT_ID

e099cebd-5eea-41a3-88db-bcb9a9c8a83e

ChocolateyLastPathUpdate

133304556037711650

Nueva...

Editar...

Eliminar

Variables de sistema

Variable

Valor

ANDROID_HOME

C:\Users\luisc\AppData\Local\Android\Sdk

AZURE_CLIENT_ID

4040303e-9579-4f08-8eb8-4045c7f8bd7

AZURE_CLIENT_SECRET

4ii8Q-6q1Y024qv8AQo4OINU09diZL2oh78bBu

AZURE_TENANT_ID

e099cebd-5eea-41a3-88db-bcb9a9c8a83e

ChocolateyInstall

C:\ProgramData\chocolatey

ComSpec

C:\WINDOWS\system32\cmd.exe

DOTNET_WORKER_DIR

C:\bin\

Nueva...

Editar...

Eliminar

<https://github.com/microsoft/agent-framework/tree/main/dotnet/samples/GettingStarted/Agents>

Securing with Azure KeyVault service



We create a new **Secret** and we also store in an environmental variable

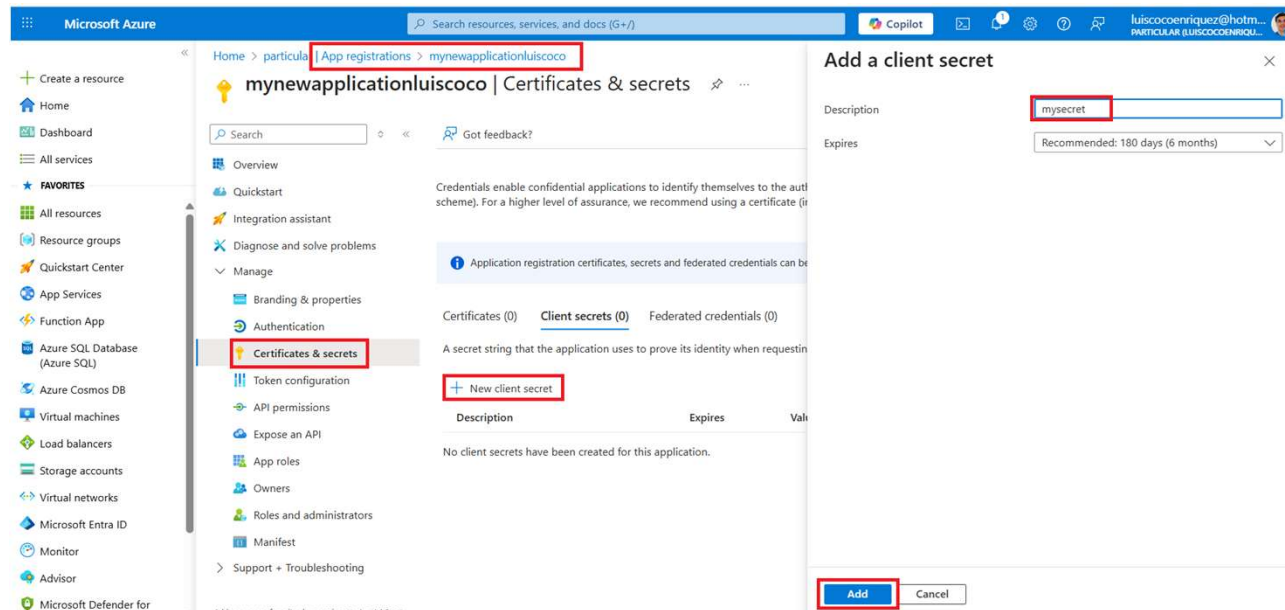
A screenshot of the Microsoft Azure portal interface. The top navigation bar shows "Microsoft Azure" and a search bar. The left sidebar contains a list of services, with "App registrations" highlighted in the "FAVORITES" section. The main content area shows the "mynewapplicationluisco" application registration. The "Overview" tab is selected, and the "Certificates & secrets" sub-tab is highlighted in the left sidebar. The "Essentials" section displays various application details, including the display name, application ID, object ID, and directory ID. A notification banner at the bottom states: "Starting June 30th, 2020 we will no longer add any new features to Azure Active Directory Authentication Library (ADAL) and Azure Active Directory Graph. We will continue to provide technical support and security updates but we will no longer provide feature updates. Applications will need to be upgraded to Microsoft Authentication Library (MSAL) and Microsoft Graph. Learn more".

<https://github.com/microsoft/agent-framework/tree/main/dotnet/samples/GettingStarted/Agents>

Securing with Azure KeyVault service



We create a new **Secret** and we also store in an environmental variable



<https://github.com/microsoft/agent-framework/tree/main/dotnet/samples/GettingStarted/Agents>

Securing with Azure KeyVault service



We copy the **Secret Value** and we store it in an environmental variable

The screenshot shows the Microsoft Azure portal interface. The left sidebar contains navigation options like 'Create a resource', 'Home', 'Dashboard', 'All services', and 'FAVORITES'. The main content area is titled 'mynewapplicationluisccoco | Certificates & secrets'. It includes a search bar, a 'Got feedback?' link, and a list of management options: Overview, Quickstart, Integration assistant, Diagnose and solve problems, and Manage. Under 'Manage', there are links for Branding & properties, Authentication, and Certificates & secrets (which is selected). The 'Certificates & secrets' section shows a table with columns: Description, Expires, Value, Copied, and Secret ID. A single entry is visible with the description 'mysecret', an expiration date of '4/5/2026', and a value starting with 'xxJ8Q-6uKhBKmkypMjV2DVAUnfChYbL'. The 'Value' column has a 'Copied' button and a 'Secret ID' column with the value '08c1ded3-a7fa-4453-ae85-10b1b8ad116f'. A notification at the top right says 'Update application credentials' and 'Successfully updated application mynewapplicationluisccoco credentials'.

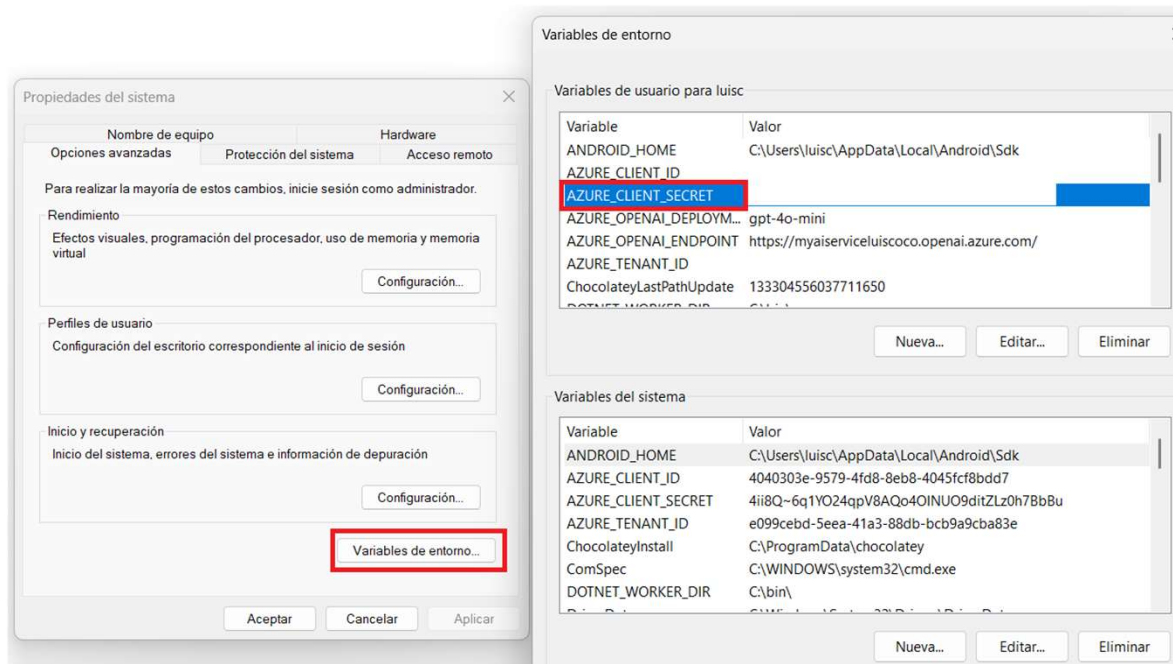
Description	Expires	Value	Copied	Secret ID
mysecret	4/5/2026	xxJ8Q-6uKhBKmkypMjV2DVAUnfChYbL	<input checked="" type="checkbox"/>	08c1ded3-a7fa-4453-ae85-10b1b8ad116f

<https://github.com/microsoft/agent-framework/tree/main/dotnet/samples/GettingStarted/Agents>

Securing with Azure KeyVault service



We store the **Secret** in an environmental variable



<https://github.com/microsoft/agent-framework/tree/main/dotnet/samples/GettingStarted/Agents>



Microsoft
Agent Framework

Securing with Azure KeyVault service

In the Azure KeyVault, we assign a new role “Key Vault Secrets User” to the Registered application

The screenshot shows the Azure portal interface for adding a role assignment. The breadcrumb path is 'Home > mykeyvaultcoenriquez | Access control (IAM) >'. The 'Add role assignment' button is highlighted. Below it, the 'Role' tab is selected, and the 'Job function roles' section is expanded. A search filter 'Key Vault Secrets User' is applied, resulting in a table of roles. The 'Key Vault Secrets User' role is highlighted in the table.

Name	Description	Type	Category	Details
Key Vault Certificate User	Read certificate contents. Only works for key vaults that use the 'Azure role-based access control'...	BuiltInRole	None	View
Key Vault Data Access Administrator	Manage access to Azure Key Vault by adding or removing role assignments for the Key Vault Ad...	BuiltInRole	None	View
Key Vault Secrets User	Read secret contents. Only works for key vaults that use the 'Azure role-based access control' per...	BuiltInRole	Security	View

Showing 1 - 3 of 3 results.

<https://github.com/microsoft/agent-framework/tree/main/dotnet/samples/GettingStarted/Agents>

Securing with Azure KeyVault service



Assign access to: **User, group, or service principal**.
Select members: search your app by **Application Name**

The screenshot shows the Microsoft Azure portal interface. On the left is a navigation pane with various service icons. The main area displays the 'Add role assignment' wizard for the 'mykeyvaultcocoenriquez' resource group, specifically the 'Access control (IAM)' section. The 'Role' is 'Key Vault Secrets User'. Under 'Assign access to', the 'User, group, or service principal' option is selected. The 'Members' section shows 'No members selected' with a '+ Select members' button. A 'Description' field contains the GUID '33cbe6c0-0bb4-4a4b-a740-793f93ede173'. On the right, a 'Select members' dialog box is open, showing a search bar with 'mynewapplicationluisco' and a list of results including 'mynewapplicationluisco Application'. The 'Selected members' section at the bottom of the dialog shows the same application. The 'Select' button in the dialog is highlighted with a red box.

<https://github.com/microsoft/agent-framework/tree/main/dotnet/samples/GettingStarted/Agents>



Microsoft
Agent Framework

Securing with Azure KeyVault service

We input the Azure **OpenAI URI** and the **Deployment name** in the **appsettings.json**

```
{  "AzureOpenAI": {    "Endpoint": "https://myaiserviceluisccoco.openai.azure.com/",    "DeploymentName": "gpt-4o-mini",    "KeyVault": {  }
```

Azure AI Foundry | Azure OpenAI / myaiserviceluisccoco / Implementaciones / gpt-4o-mini

← **gpt-4o-mini**

Detalles Métricas Riesgos y seguridad

[Abrir en el área de juegos](#) [Solicitar cuota](#) [Editar](#) [Eliminar](#)

Punto de conexión

URI de destino
https://myaiserviceluisccoco.openai.azure.com/openai/de...

Clave
.....

Idioma
Python

[Abrir en VS Code](#)

Introducción

<https://github.com/microsoft/agent-framework/tree/main/dotnet/samples/GettingStarted/Agents>



Microsoft
Agent Framework

Securing with Azure KeyVault service

We input the Azure **KeyVault URI** in the **appsettings.json** file

```
{
  "AzureOpenAI": {
    "Endpoint": "https://myaiserviceluisccoco.openai.azure.com/",
    "DeploymentName": "gpt-4o-mini",
    "KeyVault": {
      "VaultUri": "https://mykeyvaultcocoenriquez.vault.azure.net/",
      "SecretName": "my-Azure-OpenAI-API-Key"
    }
  }
}
```

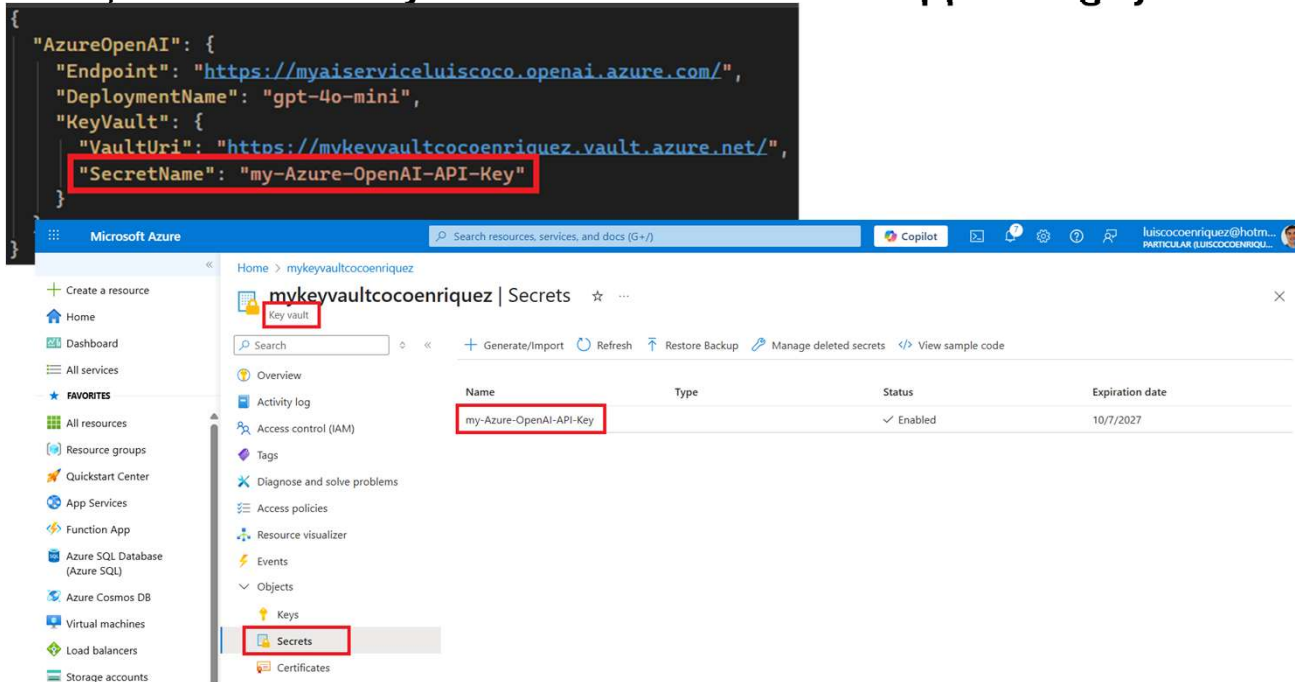
The screenshot shows the Azure portal interface for a Key Vault named 'mykeyvaultcocoenriquez'. The 'Overview' tab is selected, and the 'Vault URI' is highlighted in red, showing the value 'https://mykeyvaultcocoenriquez.vault.azure.net/'. Other details visible include the Resource group 'myRG', Location 'West Europe', and Subscription 'Subscription 2'.

<https://github.com/microsoft/agent-framework/tree/main/dotnet/samples/GettingStarted/Agents>

Securing with Azure KeyVault service

We input the Azure **KeyVault Secret Name** in the **appsettings.json** file

```
"AzureOpenAI": {  
  "Endpoint": "https://myaiserviceluiscoenriquez.openai.azure.com/",  
  "DeploymentName": "gpt-4o-mini",  
  "KeyVault": {  
    "VaultUri": "https://mykeyvaultcocoenriquez.vault.azure.net/",  
    "SecretName": "my-Azure-OpenAI-API-Key"  
  }  
}
```



Name	Type	Status	Expiration date
my-Azure-OpenAI-API-Key		✓ Enabled	10/7/2027

<https://github.com/microsoft/agent-framework/tree/main/dotnet/samples/GettingStarted/Agents>



Microsoft
Agent Framework

Securing with Azure KeyVault service

program.cs

```
using Azure;
using Azure.AI.OpenAI;
using Azure.Core;
using Azure.Identity;
using Azure.Security.KeyVault.Secrets;
using Microsoft.Agents.AI;
using Microsoft.Extensions.Configuration;
using OpenAI;
using System;
using System.IO;
```

1 referencia

```
static TokenCredential CreateCredential(out string used)
{
    var tenantId = Environment.GetEnvironmentVariable("AZURE_TENANT_ID");
    var clientId = Environment.GetEnvironmentVariable("AZURE_CLIENT_ID");
    var clientSecret = Environment.GetEnvironmentVariable("AZURE_CLIENT_SECRET");

    if (!string.IsNullOrEmpty(tenantId) &&
        !string.IsNullOrEmpty(clientId) &&
        !string.IsNullOrEmpty(clientSecret))
    {
        used = "ClientSecretCredential (env)";
        return new ClientSecretCredential(tenantId, clientId, clientSecret);
    }

    used = "DefaultAzureCredential";
    return new DefaultAzureCredential();
}
```

```
try
{
    // 1) Load non-secret config
    var cfg = new ConfigurationBuilder()
        .SetBasePath(Directory.GetCurrentDirectory())
        .AddJsonFile("appsettings.json", optional: false, reloadOnChange: true)
        .Build();

    var endpoint = cfg["AzureOpenAI:Endpoint"]
        ?? throw new InvalidOperationException("Missing AzureOpenAI:Endpoint.");
    var deploymentName = cfg["AzureOpenAI:DeploymentName"] ?? "gpt-4o-mini";

    var vaultUri = cfg["AzureOpenAI:KeyVault:VaultUri"]
        ?? throw new InvalidOperationException("Missing AzureOpenAI:KeyVault:VaultUri.");
    var secretName = cfg["AzureOpenAI:KeyVault:SecretName"] ?? "AzureOpenAI--ApiKey";

    // 2) Credential
    var credential = CreateCredential(out var usedCred);
    Console.WriteLine($"[Auth] Using: {usedCred}");

    // 3) Fetch the secret directly (no listing, no readMetadata)
    var secretClient = new SecretClient(new Uri(vaultUri), credential);
    KeyVaultSecret secret = await secretClient.GetSecretAsync(secretName);
    var apiKey = secret.Value
        ?? throw new InvalidOperationException($"Secret '{secretName}' has no value.");
}
```

<https://github.com/microsoft/agent-framework/tree/main/dotnet/samples/GettingStarted/Agents>



Microsoft
Agent Framework

Securing with Azure KeyVault service

program.cs

```
try
{
    // 4) Build the agent
    const string JokerName = "Joker";
    const string JokerInstructions = "You are good at telling jokes.";

    AIAgent agent = new AzureOpenAIClient(new Uri(endpoint), new AzureKeyCredential(apiKey))
        .GetChatClient(deploymentName)
        .CreateAIAgent(JokerInstructions, JokerName);

    Console.WriteLine(await agent.RunAsync("Tell me a joke about a pirate."));
    await foreach (var chunk in agent.RunStreamingAsync("Another pirate joke, please."))
    {
        Console.WriteLine(chunk);
    }
}
catch (AuthenticationFailedException ex)
{
    Console.Error.WriteLine("Authentication failed: " + ex.Message);
    Console.Error.WriteLine("If using ClientSecretCredential, ensure the client secret is valid and not expired.");
    throw;
}
catch (RequestFailedException ex)
{
    Console.Error.WriteLine($"Key Vault error ({ex.Status}): {ex.Message}");
    Console.Error.WriteLine("Ensure this identity has at least Secret GET permission (RBAC role or access policy).");
    throw;
}
```

<https://github.com/microsoft/agent-framework/tree/main/dotnet/samples/GettingStarted/Agents>