

STEPS TO SETUP APIM

STEP 1: Create a Resource Group and select the subscription

Home > Resource groups >

Create a resource group ...

Basics

Tags

Review + create

Resource group

 - A container that holds related resources for an Azure solution. The resource group can include all the resources for the solution, or only those resources that you want to manage as a group. You decide how you want to allocate resources to resource groups based on what makes the most sense for your organization. [Learn more](#)

Subscription *

Free Trial

Resource group name *

Region *

(US) East US

Microsoft Azure

Upgrade

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

Copilot

Home >

Resource groups

Relanto

+ Create

Manage view

Refresh

Export to CSV

Open query

Assign tags

Create a new resource group.

Filter for any field...

Subscription equals all

Location equals all

Add filter

Showing 1 to 1 of 1 records.

No grouping

<input type="checkbox"/> Name ↑↓	Subscription ↑↓	Location ↑↓
<input type="checkbox"/> apim_poc	Free Trial	East US

Step 2: Create a Service in Azure AI services (OpenAI)

Home > Azure AI services | Azure OpenAI >

Create Azure OpenAI ...

- 1 Basics
- 2 Network
- 3 Tags
- 4 Review + submit

Azure OpenAI Service provides access to OpenAI's powerful language models, including all the latest OpenAI models. These models can be easily adapted to your specific tasks, including but not limited to content generation, summarization, image understanding, semantic search, and natural language to code translation. Top use cases include Call Centers, Virtual Assistants, Accessibility, Content Generation, and Code Development. The service also features the Assistants API, Fine Tuning capabilities and many ways to connect your data to the service for conversational experiences. The service can be scaled through Standard (tokens) and Provisioned (PTUs) deployment types.

[Learn more](#)

Project Details

Subscription * ⓘ

Free Trial

Resource group * ⓘ

Create new

Previous

Next

Home > Azure AI services

Azure AI services | Azure OpenAI ...

Search

Overview

All Azure AI services

Azure AI services

- Azure AI services
- Azure OpenAI**
- AI Search
- Computer vision
- Face API
- Custom vision

Create

Manage deleted resources

Manage view

Refresh

Export to CSV

Open query

Filter for any field...

Subscription equals all

Type equals all

Add filter

Showing 1 to 1 of 1 records.

No grouping

List

<input type="checkbox"/>	Name ↑↓	Kind ↑↓	Location ↑↓	Custom Domain ... ↑↓	Pricing tier
<input type="checkbox"/>	apim-poc-azure-ai-project	OpenAI	East US	apim-poc-azure-a...	S0

Step 3: Deploy models inside the created Azure open AI service

1. Open the Azure open AI service that you created.
2. Go to the endpoint section and Copy the Service Endpoint.
3. Click on “Go to AI Foundry portal”

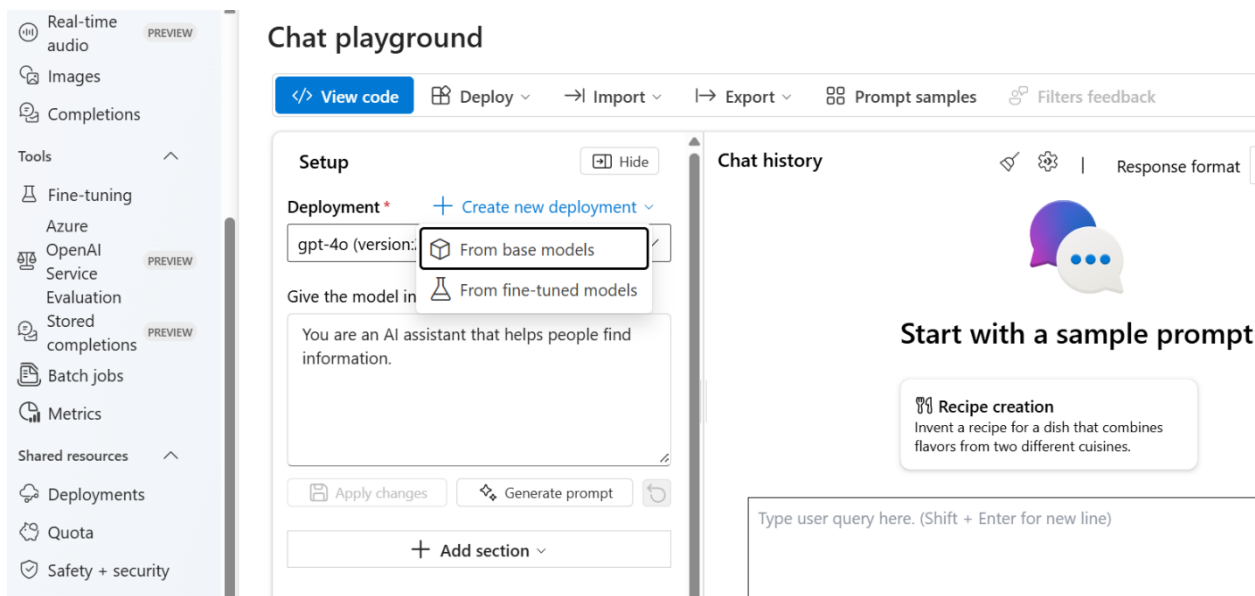
Home > Azure AI services | Azure OpenAI >



4. Go to the Deployments Section

The screenshot shows the Azure AI Foundry Chat playground interface. The top bar displays 'Azure AI Foundry | Azure OpenAI Service / apim-poc-azure-ai-project / Chat playground' and 'All resources'. A left sidebar lists various tools and resources: Real-time audio (PREVIEW), Images, Completions, Tools (Fine-tuning, Azure OpenAI Service (PREVIEW), Evaluation, Stored completions (PREVIEW), Batch jobs, Metrics), and Shared resources (Deployments (selected), Quota, Safety + security, Data files, Vector stores (PREVIEW)). The main area is titled 'Chat playground' and includes a 'View code' button, 'Deploy', 'Import', 'Export', 'Prompt samples', and 'Filters feedback' options. The 'Setup' section shows the deployment 'gpt-4o (version:2024-11-20)'. Below this, there's a text box for 'Give the model instructions and context' with the prompt 'You are an AI assistant that helps people find information.' and buttons for 'Apply changes', 'Generate prompt', and 'Add section'. A 'Chat history' section on the right shows a sample prompt 'Recipe creation' with the instruction 'Invent a recipe for a dish that combines flavors from two different cuisines.' At the bottom, there's a text input field for 'Type user query here. (Shift + Enter for new line)' and a token count '11/128000 to'.

5. Create a new Deployment Model and Deploy the model you need in that section by selecting the model and version.



Step 4: Create an APIM:

1. Go to **API Management services** and create APIM:

[Home](#) >

API Management services

Relanto

[+ Create](#) [↶ Recover](#) [⚙️ Manage view](#) [↺ Refresh](#) [⬇️ Export to CSV](#) [🔗 Open query](#) [🏷️ Assign tags](#)

Filter for any field... [Subscription equals all](#) [Resource group equals all](#) [Location equals all](#) [+ Add filter](#)

⚠️ Support for the single-tenant v1 (STv1) platform ends on 8/31/24. Migrate instances before that date to the new platform version (STv2) for continued support and access to new document for more information.

Showing 1 to 2 of 2 records. No grouping

<input type="checkbox"/>	Name ↑↓	Status ↑↓	Tier ↑↓	Platform... ↑↓	Type ↑↓	Location ↑↓	Resource group ↑↓	Si
<input type="checkbox"/>	apim-swedencentral-poc	🟢 Online	Consumption	mtv1	API Management servi...	Sweden Central	apim_poc	F
<input type="checkbox"/>	apim-swedencentral-poc-st...	🟢 Online	Standard	stv2	API Management servi...	Sweden Central	apim_poc	F

Create API Management service ...

API Management service

Basics Monitor + secure Virtual network Managed identity Tags Review + install

Project details

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

Subscription * ⓘ ▼

Resource group * ⓘ ▼

[Create new](#)

Instance details

Region * ⓘ ▼

Resource name *

Organization name * ⓘ

Review + create

< Previous

Next: Monitor + secure >

Note: 1. Use standard tier to meet the requirements

2. Enable status of Managed identity section

Create API Management service ...

API Management service

✖ Basics Monitor + secure Virtual network **Managed identity** Tags Review + install

A system assigned managed identity enables Azure resources to authenticate to cloud services (e.g., Azure Key Vault) without storing credentials in code. Once enabled, all necessary permissions can be granted via Azure role-based access control. The lifecycle of this type of managed identity is tied to the lifecycle of this resource. Additionally, each resource (e.g., Virtual Machine) can only have one system assigned managed identity. [Learn more](#)

System assigned managed identity

Enable system assigned identity to grant the resource access to other existing resources.

Status ☒

Review + create

< Previous

Next: Tags >

Step 5: Authenticating Azure Openai service with APIM using Managed identity

1. Now after creating APIM go to the Azure openAI service's Access control and click on Add role assignment

The screenshot shows the Microsoft Azure portal interface. At the top, there's a navigation bar with 'Microsoft Azure', an 'Upgrade' button, a search bar, and a 'Copilot' icon. Below this, the breadcrumb trail reads 'Home > Azure AI services | Azure OpenAI > apim-poc-azure-ai-project'. The main heading is 'apim-poc-azure-ai-project | Access control (IAM)'. On the left, a sidebar lists navigation options: Overview, Activity log, Access control (IAM) (selected), Tags, Diagnose and solve problems, Resource Management, Security, Monitoring, Automation, and Help. The main content area has a top bar with '+ Add', 'Download role assignments', 'Edit columns', and a refresh icon. Below this, there are tabs: 'Add role assignment' (selected), 'Check access', 'Roles', and 'Deny assignments'. The 'Add role assignment' tab shows 'My access' with a 'View my access' button, and 'Check access' with a 'Check access' button. At the bottom, there are two cards: 'Grant access to this resource' and 'View access to this resource'.

2. Add Cognitive Services OpenAI User(Built in) role:

The screenshot shows the 'Add role assignment' page in the Microsoft Azure portal. The breadcrumb trail is 'Home > Azure AI services | Azure OpenAI > apim-poc-azure-ai-project | Access control (IAM) > Add role assignment'. The page has two tabs: 'Job function roles' (selected) and 'Privileged administrator roles'. Below the tabs, there's a description: 'Grant access to Azure resources based on job function, such as the ability to create virtual machines.' A search bar contains 'Cognitive Services OpenAI User'. To the right of the search bar are filters: 'Type : All' and 'Category : All'. Below the filters is a table with the following data:

Name ↑↓	Description ↑↓	Type ↑↓	Category
Cognitive Services Data Contributor (Preview)	Allows to call data plane APIs, but no...	BuiltInRole	No
Cognitive Services Data Reader	Lets you read Cognitive Services data.	BuiltInRole	Pre
Cognitive Services OpenAI User	Ability to view files, models, deploym...	BuiltInRole	No
Cognitive Services User	Lets you read and list keys of Cogniti...	BuiltInRole	AI

Below the table, it says 'Showing 1 - 4 of 4 results.' At the bottom, there are three buttons: 'Review + assign', 'Previous', and 'Next'.

Note:

- (1) Assign access to Managed identity
- (2) Select the APIM according to the image below

The screenshot shows the Azure portal interface. On the left, the 'Add role assignment' page is visible, with the 'Members' tab selected. The 'Selected role' is 'Cognitive Services OpenAI User'. Under 'Assign access to', 'Managed identity' is selected. The 'Members' section shows '+ Select members'. Below this is a table with columns 'Name' and 'Object ID', currently empty with the text 'No members selected'. The 'Description' field is optional. At the bottom are buttons for 'Review + assign', 'Previous', and 'Next'.

On the right, the 'Select managed identities' dialog is open. It has a warning: 'Some results might be hidden due to your ABAC condition.' The 'Subscription' is set to 'Free Trial'. The 'Managed identity' dropdown shows 'All system-assigned managed identities (1)'. Below is a search bar and a list of results. One result is visible: 'apim-swedencentral-poc-std-002' with a long Object ID. Below the list, it says 'Selected members: No members selected. Search for and add one or more members you want to assign to the role for this resource.' There is a link 'Learn more about RBAC'. At the bottom are 'Select' and 'Close' buttons, and a 'Feedback' link.

Step 6: Go to API's Section of the APIM and Import OpenAPI(Swagger) documentation for the specific model and version you are using

Download the json file from below link and upload it in the next step(This is for openai gpt-4o model)

Ref: https://github.com/HoussemDellai/ai-course/blob/main/300_apim_genai_openai/openapi-AzureOpenAI-2024-10-21-

The screenshot shows the Azure portal page for the API Management service 'apim-swedencentral-poc-std-002'. The breadcrumb is 'Home > API Management services > apim-swedencentral-poc-std-002'. The page title is 'apim-swedencentral-poc-std-002 | APIs'. Below the title is a search bar and links for 'Developer portal' and 'Send us your feedback'.

On the left is a sidebar with navigation options: Tags, Diagnose and solve problems, Events, Settings, APIs (selected), Workspaces, Products, Subscriptions, Named values, Backends, and Policy fragments.

The main content area is divided into two sections. The top section is 'Create from definition' with four options: HTTP (Manually define an HTTP API), WebSocket (Streaming, full-duplex communication with a WebSocket server), OpenAPI (Standard, language-agnostic interface to REST APIs), and WADL (Standard XML representation of your RESTful API).

The bottom section is 'All APIs' and shows a list of APIs, with 'Azure OpenAI Service A...' visible.

Note: Add a suffix for your APIM URL endpoint.

The screenshot shows a dialog box titled "Create from OpenAPI specification". It has two tabs: "Basic" (selected) and "Full". The "Basic" tab contains the following fields and options:

- OpenAPI specification:** A text input field containing "https://" followed by "or" and a "Select a file" button with the note "(maximum size 4 MiB)".
- Include required query parameters in operation templates:** A checkbox that is checked.
- Display name:** A text input field containing "GPT endpoints".
- Name:** A text input field containing "gpt-endpoints".
- API URL suffix:** A text input field containing "openai".
- Base URL:** A text input field containing "http(s)://apim-swedencentral-poc-std-002.azure-api.net/openai".

At the bottom right, there are two buttons: "Create" (in a dark blue box) and "Cancel" (in a light gray box).

Step 7: Configuring APIM's API section:

1. Go to the API which was just imported and go to the design section

The screenshot shows the Azure API Management console for the API "apim-swedencentral-poc-std-002 | APIs". The "Design" tab is selected, showing the API's structure:

- Frontend:** A section for defining the API's frontend, including a "Search operations" field and a "Filter by tags" field.
- Inbound processing:** A section for modifying the request before it is sent to the backend service. It includes a "Policies" section with an "Add policy" button.
- Backend:** A section for defining the backend service, including an "HTTP(s) endpoint" field and a "Policies" section with a "retry" button.
- Outbound processing:** A section for modifying the response before it is sent to the client.

The console also shows a list of "All APIs" on the left, including "Azure OpenAI Service A...".

2. Go to the policy in the inbound processing and add the below policy

The screenshot shows the Azure OpenAI Service API management console. On the left, there's a sidebar with 'Search APIs', 'Filter by tags', and 'Group by tag' options. Below this, there's a list of APIs, including 'Azure OpenAI Service A...'. The main area displays the 'All operations' list, showing several POST operations. The right pane shows the 'Policies' tab for a specific API, with a list of operations. The 'inbound' policy is selected, and its configuration is shown in the right pane. The configuration includes a rate limit, a set of backend URLs, and a logic to select a backend based on a random index.

```
1 <policies>
2   <inbound>
3     <base />
4     <rate-limit-by-key calls="100" renewal-period="10" counter-key="@{(context.Request.IpAddress)}" />
5     <set-variable name="backendArray" value="@{
6       new JSONArray(
7         "https://ai-services-eastus-demo-prod.cognitiveservices.azure.com/openai",
8         "https://ai-services-eastus-demo-prod-002.cognitiveservices.azure.com/openai",
9         "https://apim-poc-azure-ai-project.openai.azure.com/openai"
10      )" />
11     }" />
12     <set-variable name="backendIndex" value="@{
13       var backends = (JSONArray)context.Variables["backendArray"];
14       return (int)(DateTime.UtcNow.Ticks % backends.Count);
15     }" />
16     <set-variable name="backendSelection" value="@{
17       var backends = (JSONArray)context.Variables["backendArray"];
18       return (string)backends[(int)context.Variables["backendIndex"]];
19     }" />
20     <set-backend-service base-url="@{((string)context.Variables["backendSelection"])}" />
21     <authentication-managed-identity resource="https://cognitiveservices.azure.com/.default" />
22   </inbound>
23 </policies>
```

Policy used (Replace your backend URL's in the JArray section):

<policies>

<inbound>

<base />

<rate-limit-by-key calls="100" renewal-period="10" counter-key="@{(context.Request.IpAddress)}" />

<set-variable name="backendArray" value="@{

new JSONArray(

"https://ai-services-eastus-demo-prod.cognitiveservices.azure.com/openai",

"https://ai-services-eastus-demo-prod-002.cognitiveservices.azure.com/openai",

"https://apim-poc-azure-ai-project.openai.azure.com/openai"

)

}" />

<set-variable name="backendIndex" value="@{

var backends = (JSONArray)context.Variables["backendArray"];

return (int)(DateTime.UtcNow.Ticks % backends.Count);

```

    }" />

    <set-variable name="backendSelection" value="@{
        var backends = (JArray)context.Variables["backendArray"];
        return (string)backends[(int)context.Variables["backendIndex"]];
    }" />

    <set-backend-service base-url="@((string)context.Variables["backendSelection"])" />

    <authentication-managed-identity resource="https://cognitiveservices.azure.com"
output-token-variable-name="msi-access-token" ignore-error="false" />

    <set-header name="Authorization" exists-action="override">
        <value>@"Bearer " + (string)context.Variables["msi-access-token"]</value>
    </set-header>

</inbound>

<backend>

    <retry condition="@((context.Response == null || context.Response.StatusCode == 500
|| context.Response.StatusCode == 429 || context.Response.StatusCode != 200)"
count="20" interval="1">

        <forward-request timeout="@((100 +
(context.Variables.ContainsKey(&quot;retryCount&quot;)) ?
(int)context.Variables[&quot;retryCount&quot;] * 100 : 0)))" buffer-request-body="true" />

        <set-variable name="backendIndex" value="@{
            var backends = (JArray)context.Variables["backendArray"];
            return ((int)context.Variables["backendIndex"] + 1) % backends.Count;
        }" />

        <set-variable name="backendSelection" value="@{
            var backends = (JArray)context.Variables["backendArray"];
            return (string)backends[(int)context.Variables["backendIndex"]];
        }" />

```

```

        <set-backend-service base-url="@((string)context.Variables["backendSelection"])" />

    </retry>

</backend>

<outbound>

    <base />

    <set-header name="X-Selected-Backend" exists-action="override">

        <value>@((string)context.Variables["backendSelection"]</value>

    </set-header>

</outbound>

<on-error>

    <base />

</on-error>

</policies>

```

3. Specify the header and query parameter name and enable the checkbox(This will be used while making an API call)

☐ Group by tag

[+ Add API](#)

All APIs

Azure OpenAI Service A...

...

REVISION 1
CREATED Mar 6, 2025, 5:35:48 PM

[Design](#)
[Settings](#)
[Test](#)
[Revisions \(1\)](#)
[Change log](#)

Subscription

Subscription required
☒

Header name

Query parameter name

Step 8: Subscription Key Creation

- 1. Create a subscription key in the Subscription section of APIs under APIM

Home > API Management services > apim-swedencentral-poc-std-002

apim-swedencentral-poc-std-002 | Subscriptions

API Management service

Search

+ Add subscription Columns Refresh

Tags

Diagnose and solve problems

Events

Settings

APIs

Workspaces

APIs

Products

Subscriptions

Named values

Backends

Policy fragments

API Tags

Schemas

API consumers can subscribe to Products to start using your APIs. [Learn more](#)

Search

State

All Pending approval

Display name	Primary key	Secondary key	Scope
	*****	*****	Product: Starter
	*****	*****	Product: Unlimited
Built-in all-access su...	*****	*****	Service
All APIs subscription	*****	*****	All APIs
dummy	*****	*****	All APIs
ProdUser	*****	*****	All APIs
DevUser	*****	*****	All APIs

New subscription

Subscription

Name *

Testing

Display name

Testing

Allow tracing

Scope

API

API

Azure OpenAI Service API(Custom p...

Product

User

Administrator (Shreeraj.Shetty@Rel...

Send notification for

Do not send

- 2. Retrieve the key and copy

Search

+ Add subscription Columns Refresh

Tags

Diagnose and solve problems

Events

Settings

APIs

Workspaces

APIs

Products

Subscriptions

Named values

Backends

Policy fragments

API Tags

API consumers can subscribe to Products to start using your APIs. [Learn more](#)

Search

State

All Pending approval

Primary key	Secondary key	Scope	State	Owner
*****	*****	Product: Starter	Active	Administrator
*****	*****	Product: Unlimited	Active	Administrator
*****	*****	Service	Active	
*****	*****	All APIs	Active	
*****	*****	All APIs	Active	Administrator
*****	*****	All APIs	Active	Administrator
6ab3a7e5160a...	77da4048619a...	All APIs	Active	Administrator

Show/hide keys

Activate subscription

Submit subscription

Suspend subscription

Reject subscription

Cancel subscription

Delete subscription

Regenerate primary key

Regenerate secondary key

Step 9: Testing

This Apim's performance can be tested using Apache benchmark docker image

Run the below Docker command:

```
docker run --rm -v <Path to your payload file>:/data jordi/ab -n 10 -c 10 -l -T  
"application/json" -H "api-key: <your-subscription-key>" -p /data/apim-body.json -v 2 -  
e /data/apim-errors.csv -g /data/apim-results.tsv https://apim-swedencentral-poc-  
std-002.azure-api.net/openai/deployments/gpt-4o/chat/completions?api-  
version=2024-08-01-preview
```

Note:

-n represents the number of requests

-c represents the concurrency

That means in every second c number of requests are sent parallelly.

Step 10: Checking the logs

Go to the Monitoring section of the APIM and query to get the logs.

Query:

1. To get the Details:

ApiManagementGatewayLogs

| where TimeGenerated > ago(1d)

| where IsRequestSuccess == true

| where ResponseCode == 200

| project TimeGenerated, BackendUrl| top 10 by TimeGenerated desc

2. To get the Count of the Requests hit to particular Backend:

```
ApiManagementGatewayLogs
| where TimeGenerated > ago(1d)
| where IsRequestSuccess == true
| where ResponseCode == 200
| project TimeGenerated, BackendUrl
| top 50 by TimeGenerated desc // Fetch last 30 requests
| summarize HitCount = count() by BackendUrl
| order by HitCount desc
```