

```
# Copyright (c) 2026 Oracle and/or its affiliates.  
# Licensed under the Universal Permissive License v 1.0 as shown at  
https://oss.oracle.com/licenses/upl/
```

# OCI GenAI Auth Documentation

Version: 2026-03-19

## Overview

OCI GenAI Auth is a lightweight Python helper library that signs HTTP requests to OCI Generative AI's OpenAI-compatible REST APIs. It lets you use the official OpenAI SDK while authenticating

with OCI IAM credentials (session, user, resource, or instance principals), instead of API keys.

## When to Use

Use OCI GenAI Auth when you want:

- IAM-based authentication (preferred for enterprise deployments)
- OCI-managed identity controls and auditability
- Direct use of the OpenAI Python SDK with OCI endpoints

## Core Concept

OCI GenAI Auth provides an `httpx` authentication class (`OciSessionAuth` and others) that signs requests. You attach it to the OpenAI client's `http_client` so every request is OCI-signed.

## Basic Usage (IAM Auth)

```
import httpx
from openai import OpenAI
from oci_genai_auth import OciSessionAuth

client = OpenAI(
    base_url="https://inference.generativeai.us-chicago-
1.oci.oraclecloud.com/openai/v1",
    api_key="not-used",

http_client=httpx.Client(auth=OciSessionAuth(profile_name="DEFAULT"))
,
```

## AgentHub (Non-Pass-Through)

AgentHub requires a project OCID and uses the non-pass-through endpoint. Only the project OCID is required.

```
import httpx
from openai import OpenAI
from oci_genai_auth import OciSessionAuth

client = OpenAI(
    base_url="https://inference.generativeai.us-chicago-
1.oci.oraclecloud.com/openai/v1",
    api_key="not-used",
    project="ocid1.generativeaiproject.oc1.us-chicago-
1.aaaaaaaexample",

http_client=httpx.Client(auth=OciSessionAuth(profile_name="DEFAULT"))
,
```

## Partner APIs (Pass-Through)

Pass-through endpoints require a compartment OCID header:

```
import httpx
from openai import OpenAI
from oci_genai_auth import OciSessionAuth

client = OpenAI(
    base_url="https://inference.generativeai.us-chicago-
1.oci.oraclecloud.com/v1",
    api_key="not-used",
    default_headers={
        "opc-compartment-id":
"ocid1.compartment.oc1..aaaaaaaexample",
    },
    http_client=httpx.Client(auth=OciSessionAuth(profile_name="DEFAULT"))
)
```

## Supported Auth Types

- OciSessionAuth: session token from OCI CLI profile
- OciUserPrincipalAuth: API key from OCI config
- OciResourcePrincipalAuth: workload identity (OCI services)
- OciInstancePrincipalAuth: OCI Compute instance identity

## Notes

- The library only signs requests; it does not wrap the OpenAI SDK.
  - Use base\_url for OCI endpoints exactly as provided.
  - API keys are not required when using IAM authentication.
-