# Documentação da Integração com Microsoft Fabric API

## 1. Introdução

Este documento descreve o processo de autenticação e consumo da API do Microsoft Fabric utilizando Python e OAuth2. O objetivo é fornecer um guia passo a passo para registrar um aplicativo no Azure AD, gerar um token de acesso e realizar chamadas para a API.

## 2. Configuração no Azure AD

1. Acesse o portal do Azure: **https://portal.azure.com/**

2. Vá para "Azure Active Directory" > "**Registros de Aplicativo**" e clique em "**Novo registro**".

3. Configure os seguintes campos:

* Nome: FabricAPI
* Tipo de conta: Contas na organização

Interface gráfica do usuário, Texto, Aplicativo, Email

O conteúdo gerado por IA pode estar incorreto.

* Redirecionamento: <http://localhost:5000/callback>

Interface gráfica do usuário, Aplicativo

O conteúdo gerado por IA pode estar incorreto.

* Interface gráfica do usuário, Texto, Aplicativo, Email

  O conteúdo gerado por IA pode estar incorreto.AUTENTICAÇÃO:

4. Após criar o aplicativo, copie os seguintes valores:

* Application (Client) ID
* Directory (Tenant) ID

Nome de exibição

**FabricAPI**

ID do aplicativo (cliente) / Application (Client) ID

**03f12ddb-c47a-420e-bceb-0e6a8ccdfaad**

ID do Objeto

**597a0102-1caf-4f20-a51a-221f91e1f4de**

ID do diretório (locatário) / Directory (Tenant) ID

**6653b20a-edb6-44e9-b25f-e28c14679f1d**

Valor secret

**O2L8Q~TT6T3pecCHFGRinXB.rqm75a0ib4GfYcLt**

ID Secreto

**dbbae66e-6165-44e3-a106-6d66ef0db408**

5. Vá para "Permissões de API" e adicione as permissões:

**- Workspace.Read.All**

**- Workspace.ReadWrite.All**

**- Dataset.Read.All**

**- Dataset.ReadWrite.All**

**- Report.Read.All**

**- Report.ReadWrite.All**

6. Conceda o consentimento do administrador.

Interface gráfica do usuário, Aplicativo, Tabela

O conteúdo gerado por IA pode estar incorreto.

## 3. Autenticação e Geração do Token

Para autenticação, utilizamos o fluxo de "authorization\_code" via OAuth2, permitindo que o usuário faça login via navegador e obtenha um token válido.

O código Python abaixo inicializa um servidor Flask para capturar o código de autorização e trocar pelo token de acesso:

Tela de computador com texto preto sobre fundo branco

O conteúdo gerado por IA pode estar incorreto.**LoginFabric.py**

import requests

import webbrowser

from flask import Flask, request

# Configurações do Azure AD

TENANT\_ID = "6653b20a-edb6-44e9-b25f-e28c14679f1d"

CLIENT\_ID = "03f12ddb-c47a-420e-bceb-0e6a8ccdfaad"

CLIENT\_SECRET = "O2L8Q~TT6T3pecCHFGRinXB.rqm75a0ib4GfYcLt"

REDIRECT\_URI = "http://localhost:5000/callback"

SCOPE = "https://api.fabric.microsoft.com/.default"

# Criar aplicação Flask para capturar o token

app = Flask(\_\_name\_\_)

@app.route('/')

def login():

    auth\_url = (

        f"https://login.microsoftonline.com/{TENANT\_ID}/oauth2/v2.0/authorize"

        f"?client\_id={CLIENT\_ID}"

        f"&response\_type=code"

        f"&redirect\_uri={REDIRECT\_URI}"

        f"&response\_mode=query"

        f"&scope={SCOPE}"

        f"&state=12345"

    )

    webbrowser.open(auth\_url)  # Abre o navegador automaticamente

    return "Por favor, faça login e autorize o aplicativo."

@app.route('/callback')

def callback():

    code = request.args.get("code")

    token\_url = f"https://login.microsoftonline.com/{TENANT\_ID}/oauth2/v2.0/token"

    # Troca o código de autorização pelo Access Token

    data = {

        "grant\_type": "authorization\_code",

        "client\_id": CLIENT\_ID,

        "client\_secret": CLIENT\_SECRET,

        "code": code,

        "redirect\_uri": REDIRECT\_URI,

        "scope": SCOPE

    }

    response = requests.post(token\_url, data=data)

    if response.status\_code == 200:

        token\_info = response.json()

        access\_token = token\_info["access\_token"]

        print("\n✅ Token obtido com sucesso! Copie e use no seu código:")

        print(access\_token, "\n")  # Exibe o token completo no terminal

        return f"✅ Token obtido com sucesso! Use esse token nas requisições: {access\_token}"

        # return "✅ Token obtido com sucesso! O token completo foi impresso no terminal."

    else:

        return f"❌ Erro ao obter token: {response.text}"

# Iniciar o servidor Flask

if \_\_name\_\_ == "\_\_main\_\_":

    app.run(port=5000, debug=True)

## 4. Uso do Token para Acessar a API

Texto

O conteúdo gerado por IA pode estar incorreto.Com o token gerado, podemos realizar chamadas para a API do Microsoft Fabric. O exemplo abaixo lista os workspaces disponíveis:

**FabricWorkspaces.py**

import requests

# Substitua pelo token que você obteve na autenticação web

access\_token = "eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiIsIng1dCI6IkpETmFfNGk0cjdGZ2lnTDNzSElsSTN4Vi1JVSIsImtpZCI6IkpETmFfNGk0cjdGZ2lnTDNzSElsSTN4Vi1JVSJ9..cturmVFZMNtp5ZFVrk1jLbPoD\_XTeyf6IX26aai4oHQSZgstqVPkP-Y9CJmn0pH7ZkD1C5oB2FEZZw77bAcysrc6WXQN3k8nOlwc0FVFEtvN9vQFGr1mTkW0iDrEvpvneT0wxY9SHciErOittZbi5Say28RmU0WzfbpRpCzWPLxuxgdepsIHSc2kqzdPVi4W\_6UHfEE7r1v\_GaxvySqW0wHKCie2\_s48Bn9WKDTZTflR6qlT\_B0LGBBxcM54ZN-VapwB7IiGyB0lkAc2znIVXxxeU200AtMtHjAEn\_d5VLTiLYEDaHhO0lwOUWIjBxiBENOZHL-2vGdWDAkoNZE5rw"

# URL da API para listar os workspaces

url = "https://api.fabric.microsoft.com/v1/workspaces"

# Cabeçalhos com o token de autorização

headers = {

    "Authorization": f"Bearer {access\_token}",

    "Content-Type": "application/json"

}

# Fazendo a requisição GET

response = requests.get(url, headers=headers)

# Exibindo os dados dos workspaces

if response.status\_code == 200:

    print("🔹 Workspaces disponíveis:")

    print(response.json())

else:

    print(f"❌ Erro ao obter workspaces: {response.status\_code}")

    print(response.text)

## 5. Solução de Problemas

Se ocorrerem erros, consulte os possíveis problemas abaixo:

🔹 \*\*Erro 401 Unauthorized\*\*: Verifique se o token foi gerado corretamente e se o escopo está correto.

🔹 \*\*Erro 403 Forbidden\*\*: Confirme que o usuário tem acesso ao workspace no Power BI.

🔹 \*\*Erro 500 Internal Server Error\*\*: Verifique a configuração do Flask e tente novamente.