CLOUDEX for Admins only

- Admin settings
  - Office365: How to get TenantId, ClientId and ClientSecret?
  - How to change APPLICATION Callback URL in Azure Portal?
- Shipment integrations
  - How to send shipment data to Cloudex TMS via API?
  - How to send shipment milestones to Cloudex TMS via API?
  - How to send shipment documents to Cloudex TMS via API?
- AI Plugin
  - Cloudex TMS assistant API

Admin settings

Office365: How to get TenantId, ClientId and ClientSecret?

If you use Microsoft office 365 for email sending. Your company Microsoft admin need to get following settings from **Azure AD Application**

- **Directory (Tenant) ID**
- **Application (Client) ID**
- **Client Secret Value**

These three values are required for integrating CLOUDEX TMS with Microsoft Graph using OAuth2.

---

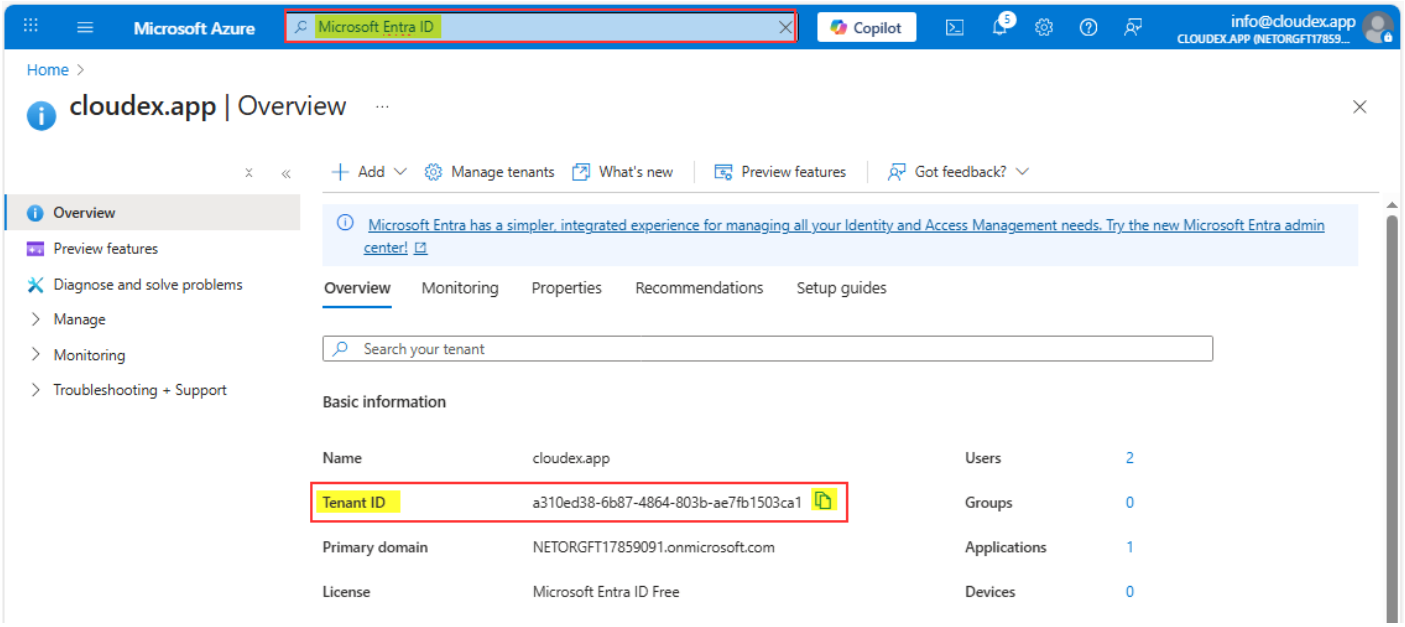
1 Sign in to Azure Portal

Open: https://portal.azure.comSign in with a Microsoft 365 **Global Admin** or **Application Administrator** account.

---

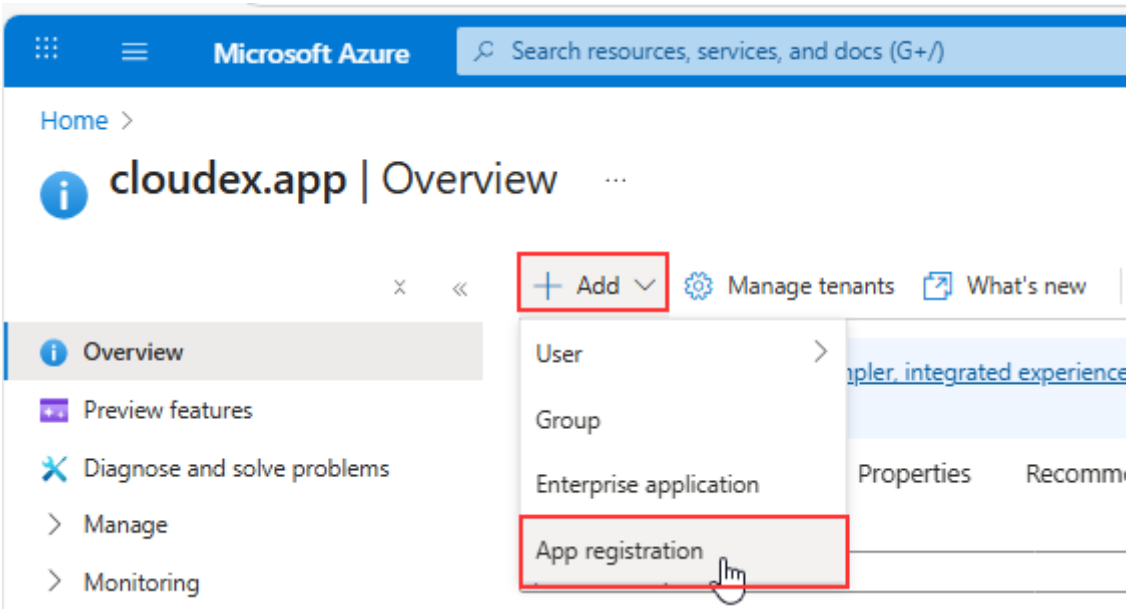
2 Open “Microsoft Entra ID” (Azure AD)

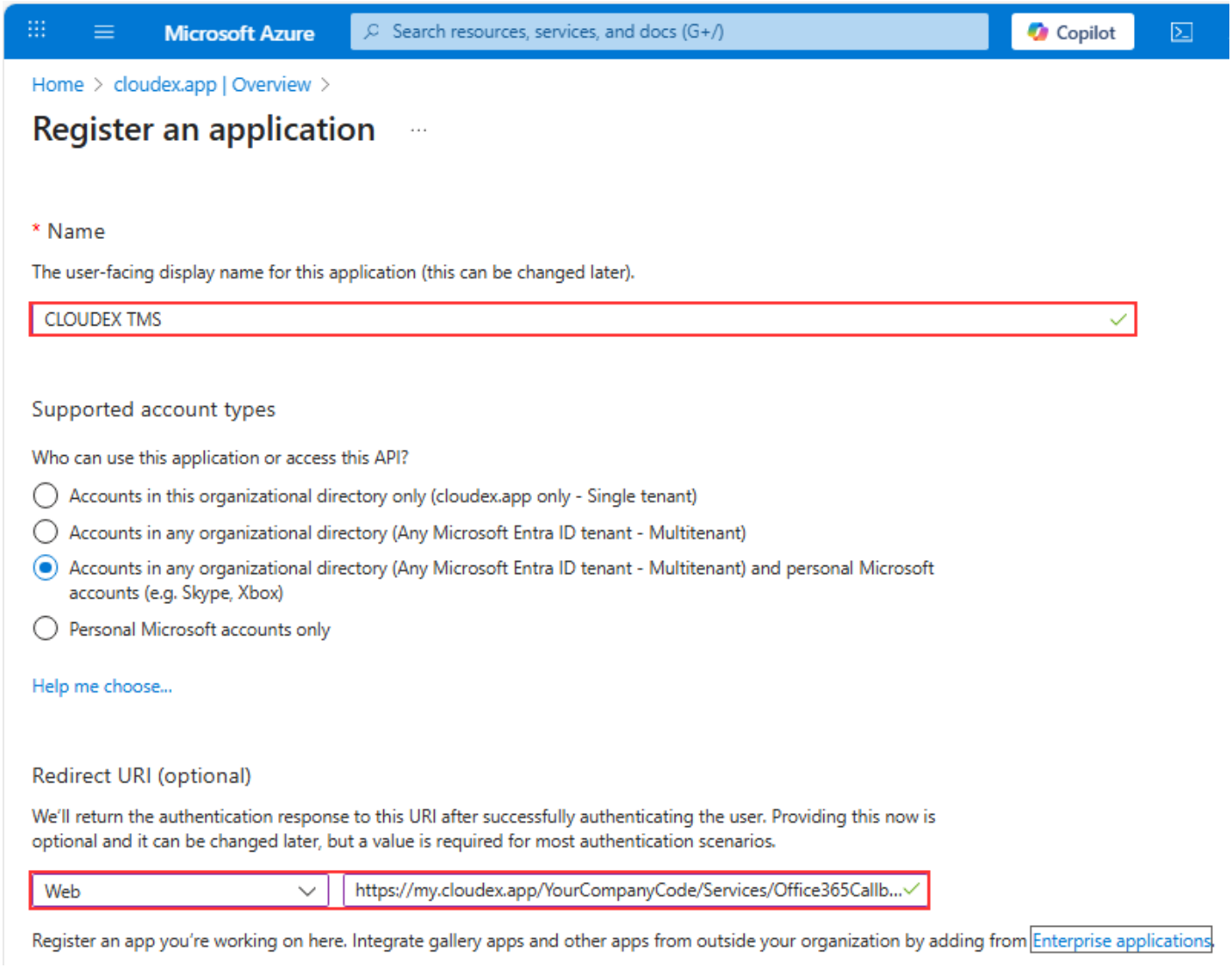
Left menu → **Microsoft Entra ID**
(or search “Entra ID” in the top search bar)



3 Create a new App Registration

1. Left menu: **App registrations**
2. Click **New registration**





URL type must be "Web" NOT "Single page application"
Looks like beter to register as Single tenant. If you register as multi tenant then can be so than you need to enter CLOUDEX MpnId 7086574

Fill the form:

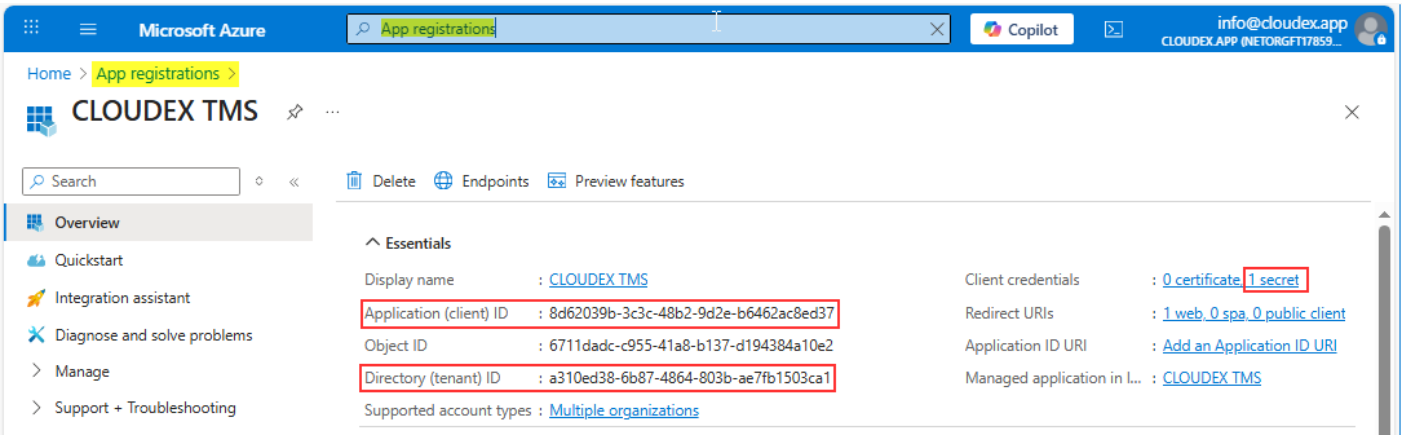
Field	Value
Name	CLOUDEX TMS Email Integration (or any name)
Supported account types	**✔ Accounts in any organizational directory (multitenant)**
Redirect URI	Select **Web** → Enter your redirect: `https://my.cloudex.app/YourCompanyCode/Services/Office365Callback.aspx` 1) Replace `YourCompanyCode` with your web app CompanyCode 2) Redirect URL is case-sensitive

Click **Register**.

---

4 Get the Tenant ID and Client ID

After creation, you will be redirected to the app's **Overview** page.
Here you will see:
✔ **Directory (Tenant) ID**
✔ **Application (Client) ID**



Copy them and save.

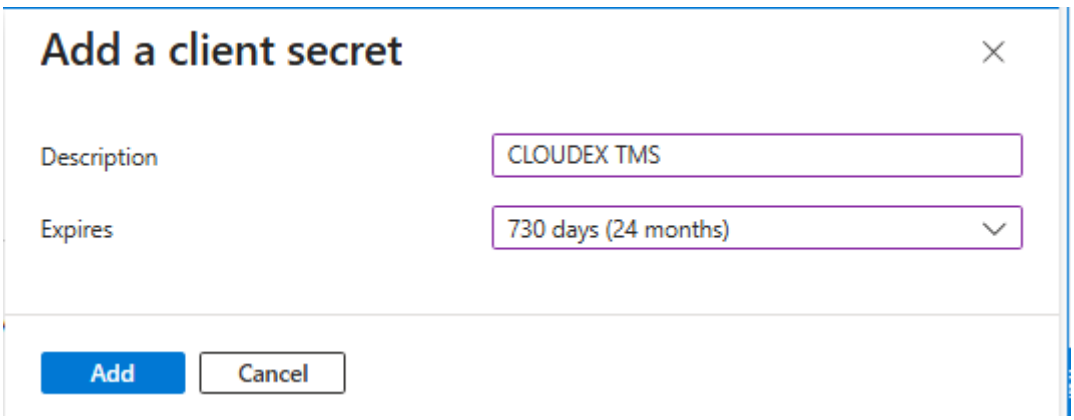
You already have **2/3 values**.

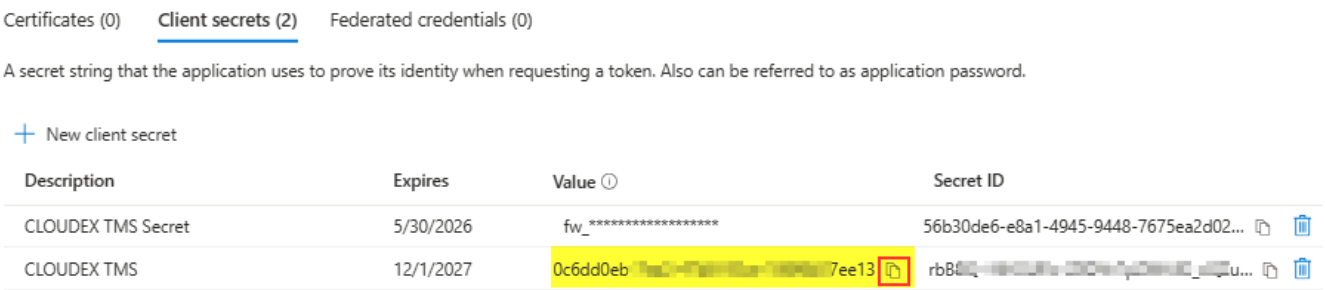
---

5 Create Client Secret

Side menu → **Certificates & secrets**

1. Click **New client secret**
2. Enter a name: `CLOUDEX Secret`
3. Choose expiration:
   - 6 months (not recommended)
   - 12 months
   - **24 months** (recommended)
   - Or "Custom"
4. Click **Add**





IMPORTANT ⚠

Copy the Client Secret VALUE immediately.
You will never be able to see it again later.

Store it securely (Azure Key Vault, password manager, etc).

Now you have:

- **ClientId**
- **TenantId**
- **ClientSecret Value**

---

6 Add Required API Permissions

Side menu → **API permissions**

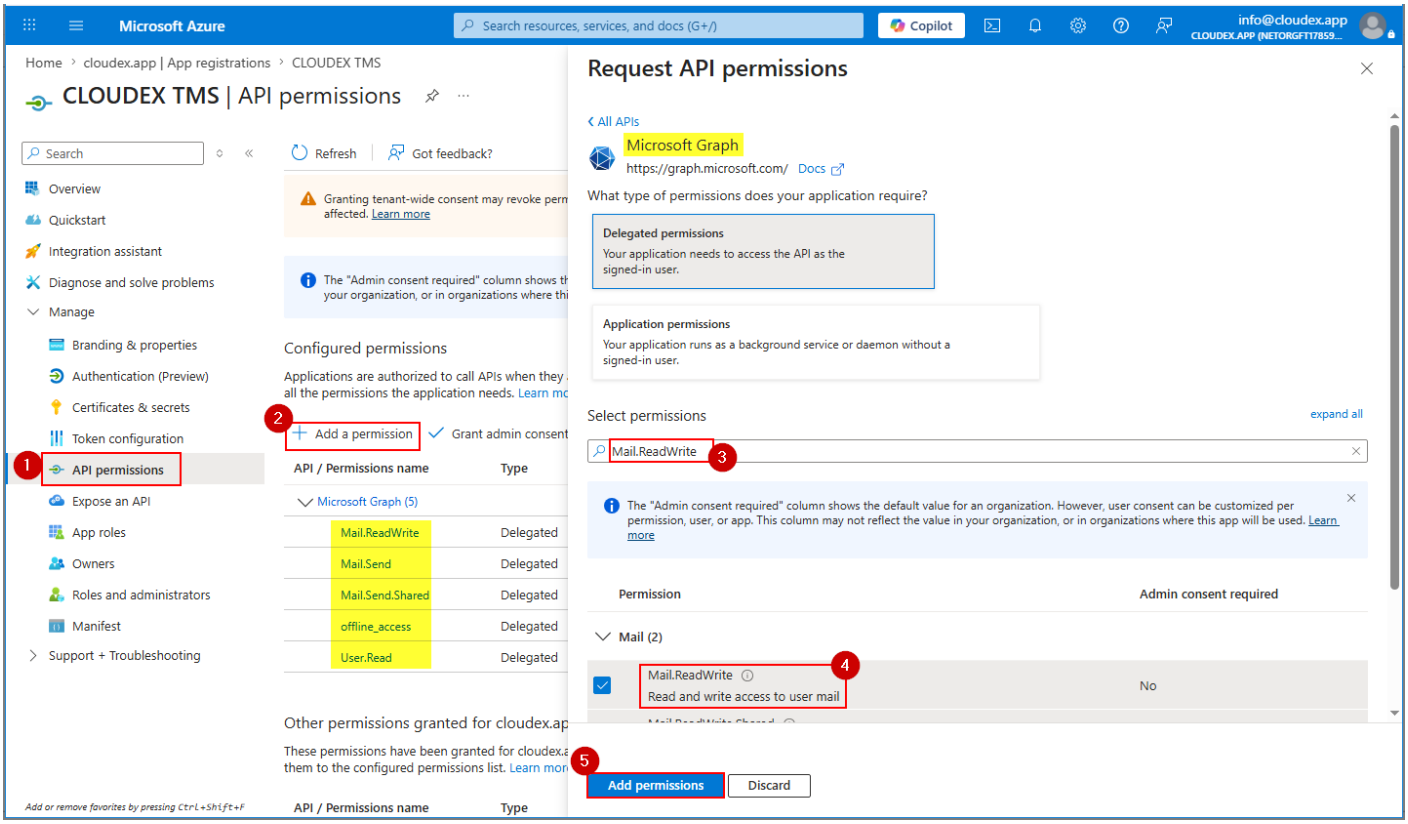
Click:

- **Add a permission**
- **Microsoft Graph**
- **Delegated permissions**

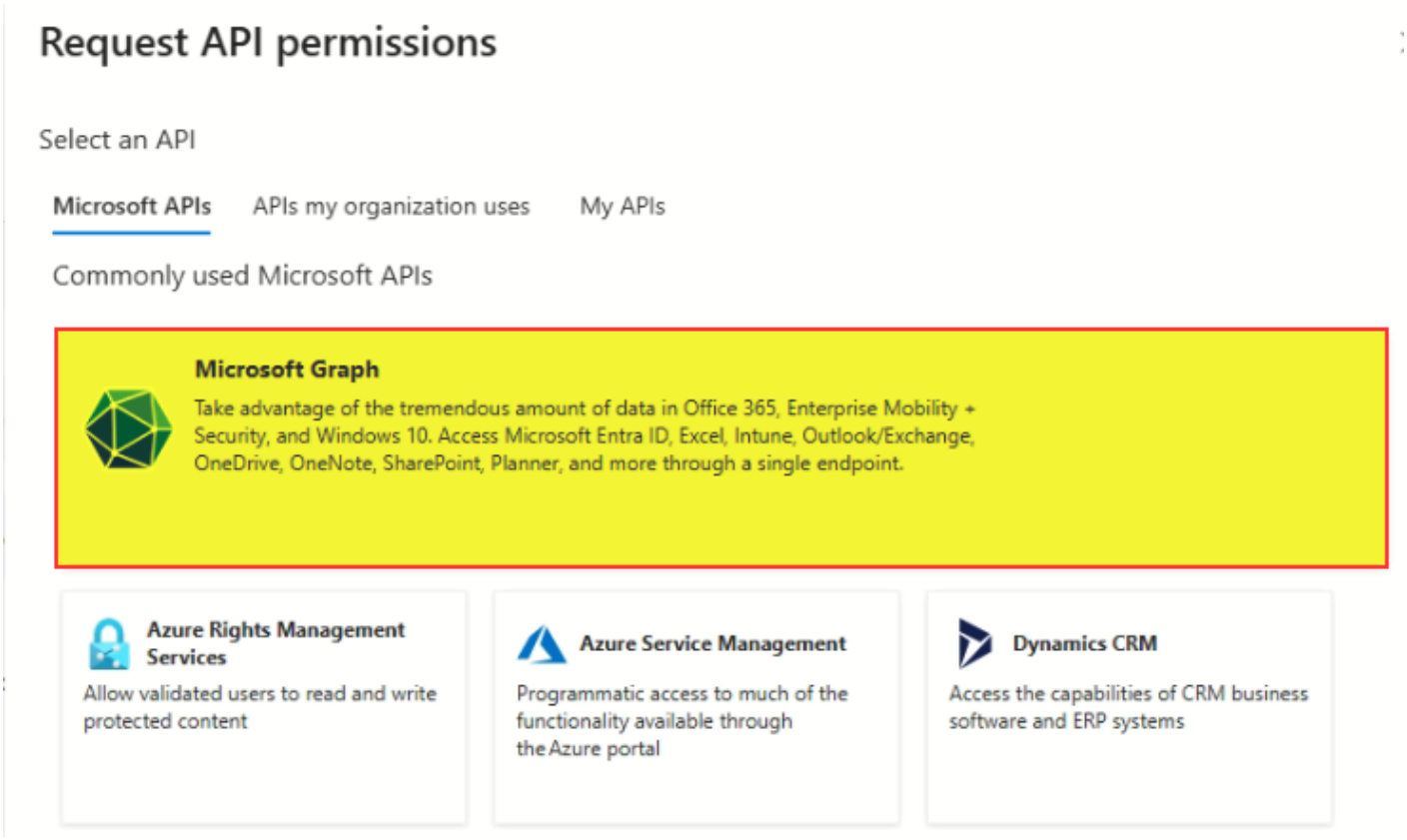
Search + select:

✔ `Mail.ReadWrite`
✔ `Mail.Send`
✔ `Mail.SendShared`
✔ `offline_access`
✔ `User.Read`
!!! If email sending does not work, then add also following permissions
✔ `email`
✔ `openid`
✔ `profile`

Then click **Add permissions**.
Then click **Grant admin consent**.



Can be in new interface need to choose Microsoft Graph API

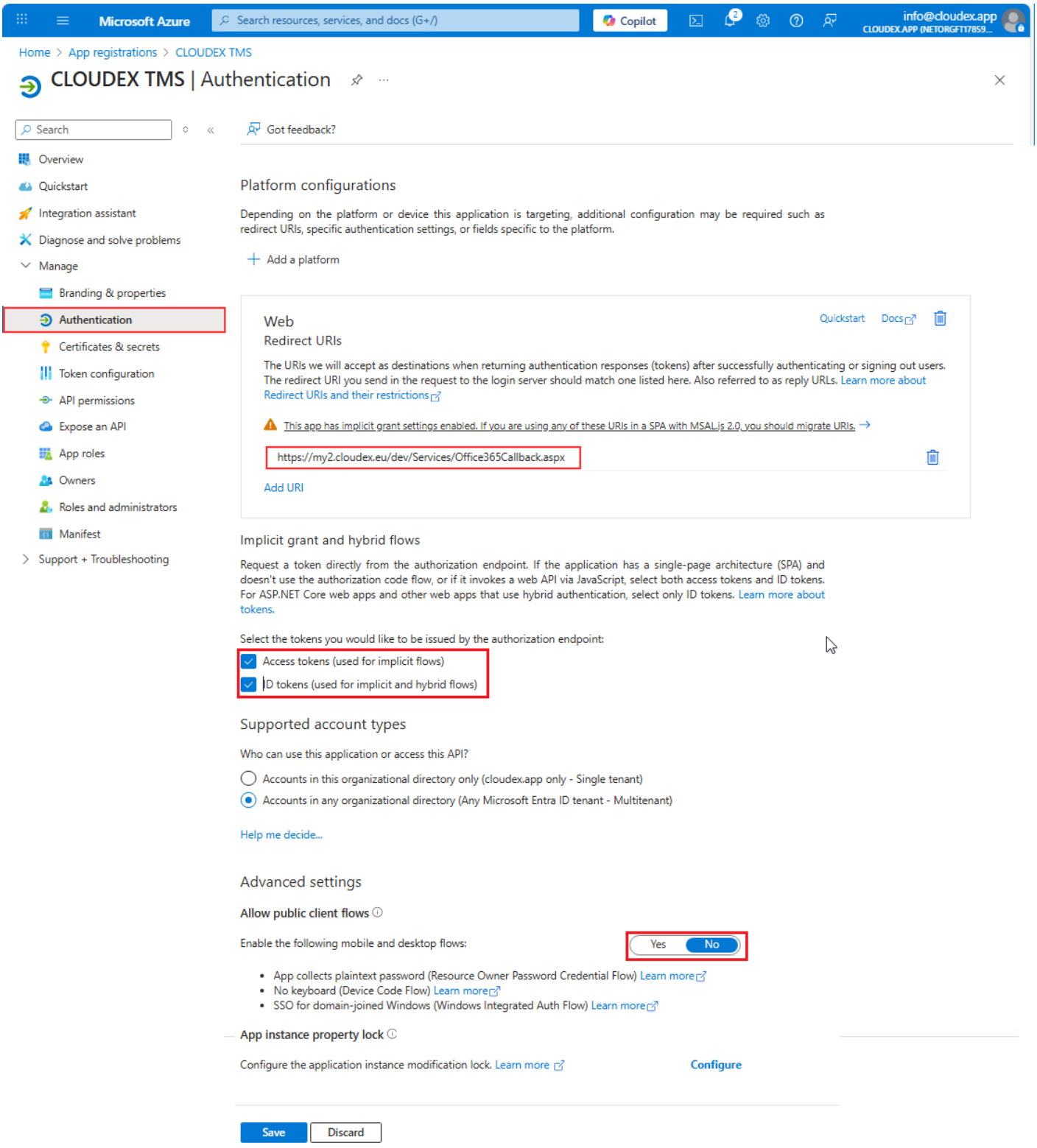


7 Final Check: Authentication Settings

Side menu → **Authentication**

Ensure:

✔ Your redirect URI is correct
✔ “Allow public client flows” is **OFF**
✔ “Access tokens” and “ID tokens” are **ON**



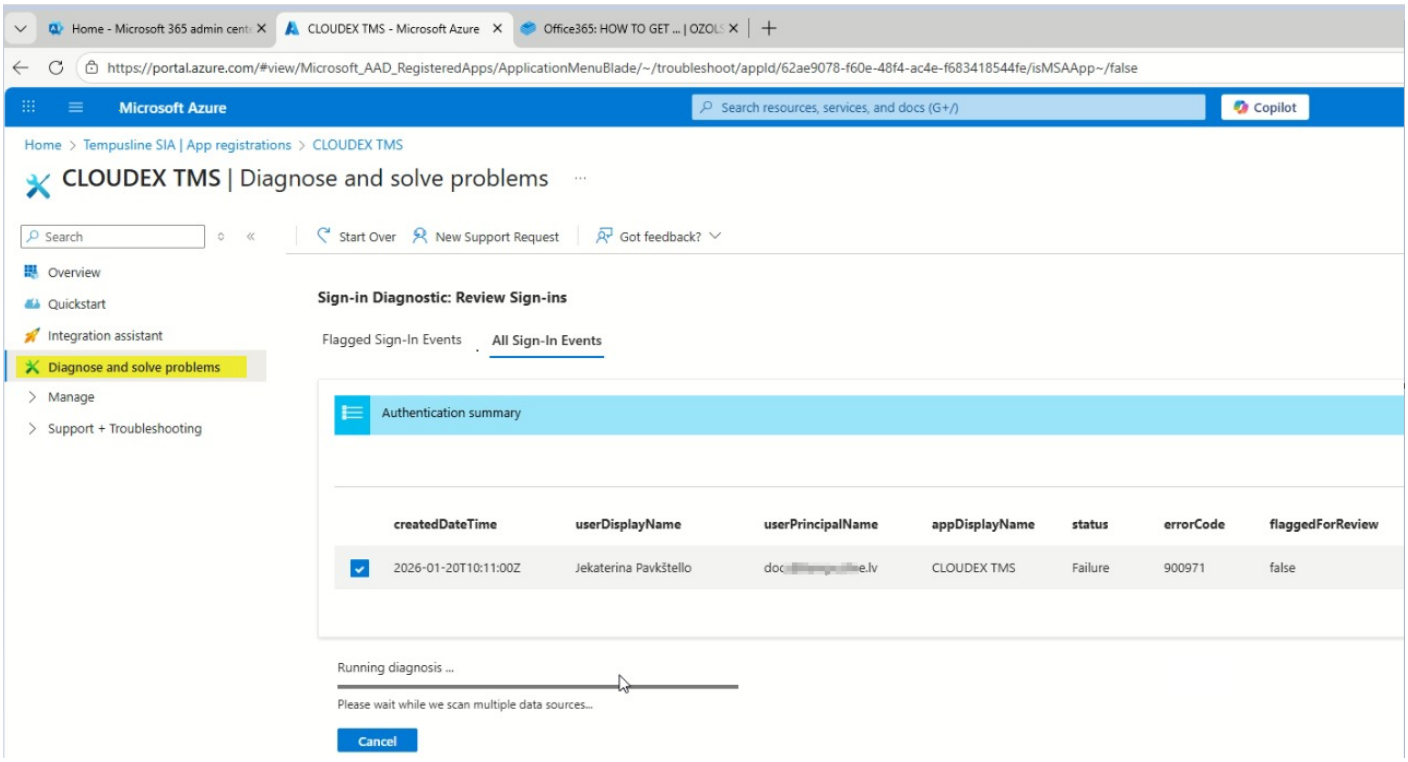
◻◻ DONE — Values ready to use

You now have everything:

Parameter	Where to find it
TenantId	App → Overview → Directory (tenant) ID
ClientId	App → Overview → Application (client) ID
ClientSecret	App → Certificates & Secrets → **Value**

Problem solution

In case you have any problems sending out emails from CLOUDEX TMS or Ozols you can run **"Diagnose and solve problems"** tool



Admin settings

How to change APPLICATION Callback URL in Azure Portal?

1 Open Azure Portal

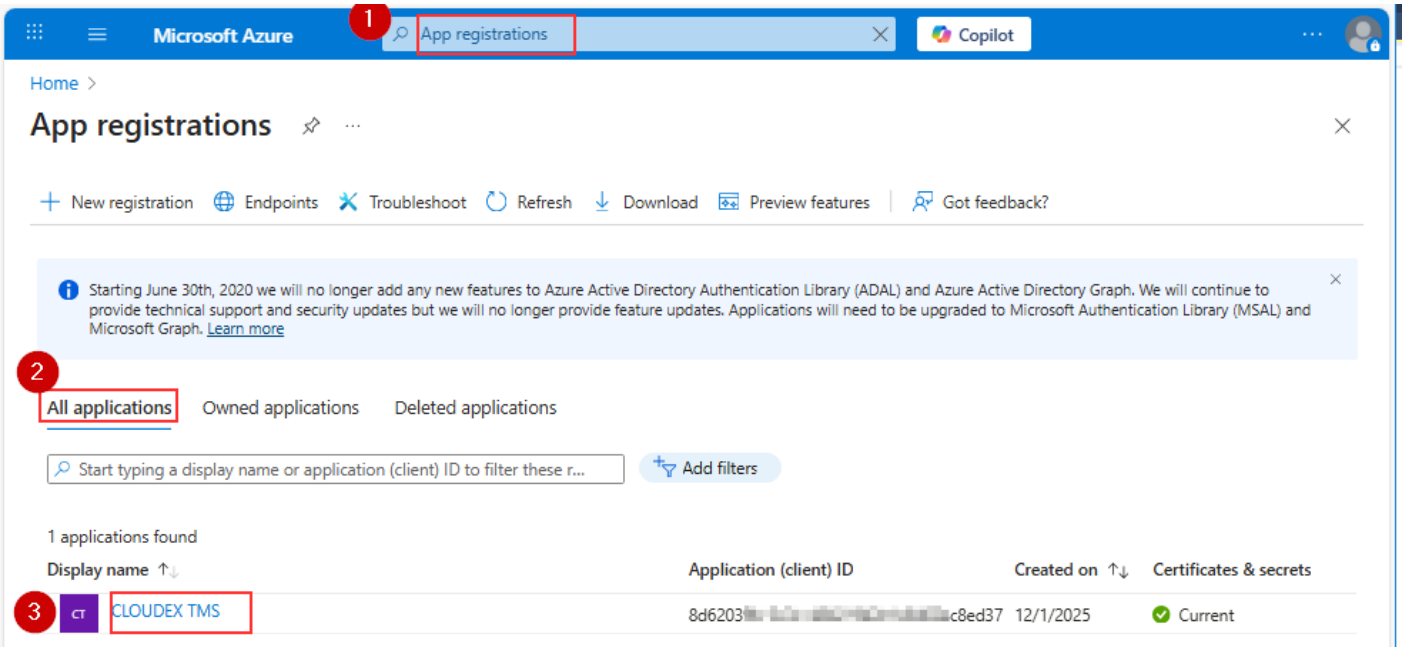
Go to: https://portal.azure.com

Sign in using a **Global Administrator** or **Application Administrator** account.

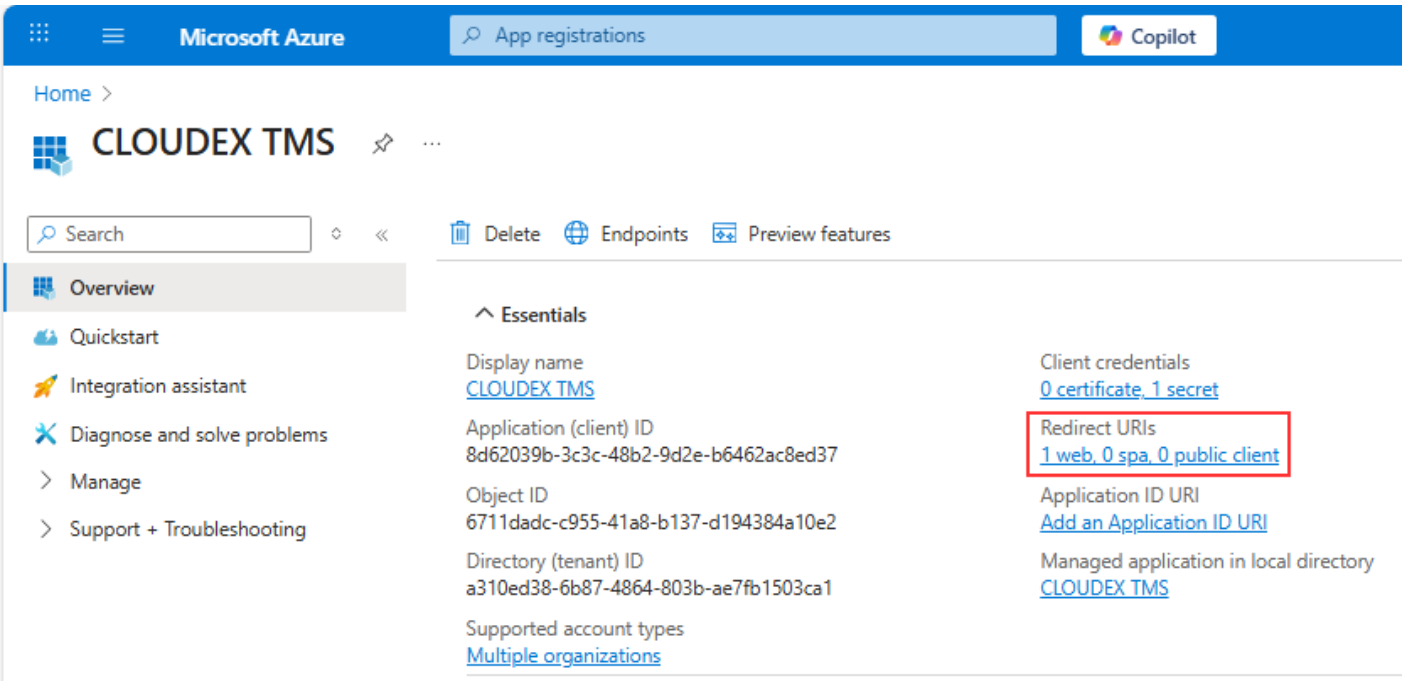
---

2 Open *Microsoft Entra ID*

1) In the left menu, click: **Microsoft Entra ID or Search for App registrations**
2) Open All Applications
3) Open CLOUDEX TMS application

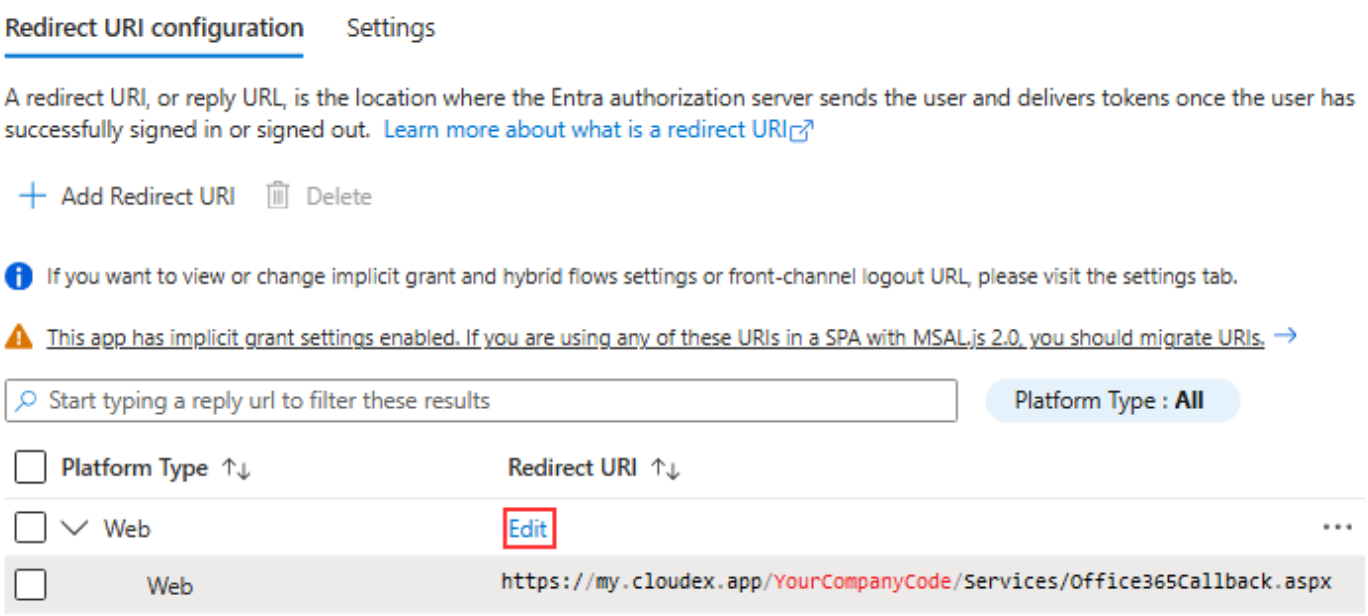


Choose Redirect URLs



Press edit button and set correct URL

!!!! In YourCompanyCode you should place your company code
https://my.cloudex.app/YourCompanyCode/Services/Office365Callback.aspx



Redirect URL should be under Web NOT under SIngle-application

Shipment integrations

How to send shipment data to Cloudex TMS via API?

Use the **Cloudex TMS API** when shipments created in another system need to be automatically imported into Cloudex TMS.

This is typically the first step of a shipment integration between an external TMS, ERP, customer system, or another application and Cloudex TMS.

Before you start

You need:

- access to the Cloudex TMS API;
- a valid **Cloudex API key**;
- access to the Cloudex TMS API documentation.

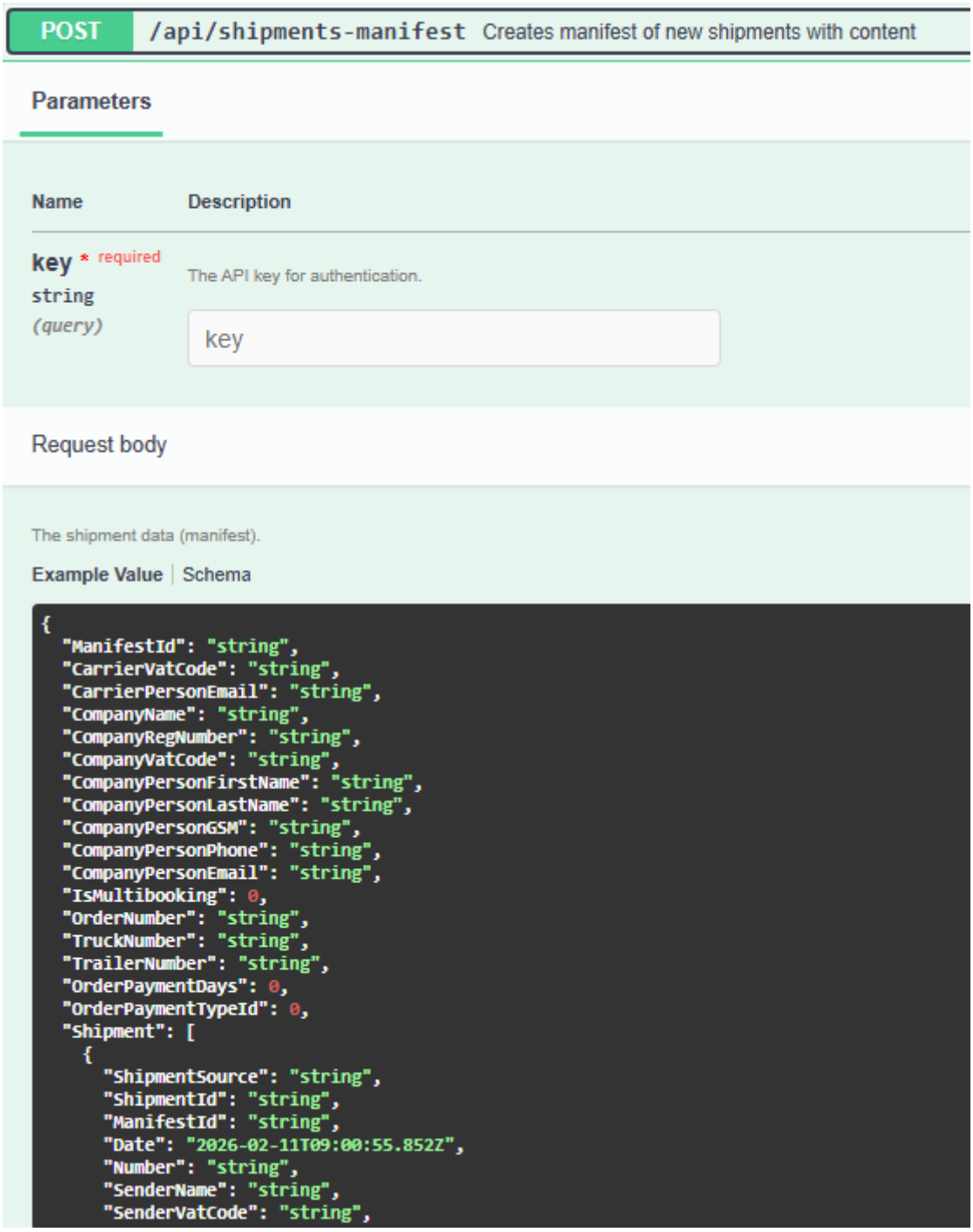
See: **How to create API key**

Full Cloudex TMS API documentation: https://my.cloudex.lv/cloudex-api/swagger/index.html

Which endpoint should be used?

To create/import a shipment in Cloudex TMS, use the: **shipment-manifest** endpoint.

The external system sends the shipment information to Cloudex TMS using this endpoint.



Integration flow

A typical shipment integration works as follows: **External system → Shipment manifest → Cloudex TMS**

Related articles

- **How to create API key**

How to send shipment milestones to Cloudex TMS via API?

Use the **Cloudex TMS API milestones endpoint** when an external system needs to update Cloudex TMS about the progress of an existing shipment.

Examples include shipment milestone/status updates such as:

- Shipment loaded;
- Shipment delivered.

Before you start

The shipment must already exist in Cloudex TMS and the external system must be able to identify which Cloudex TMS shipment the milestone belongs to.

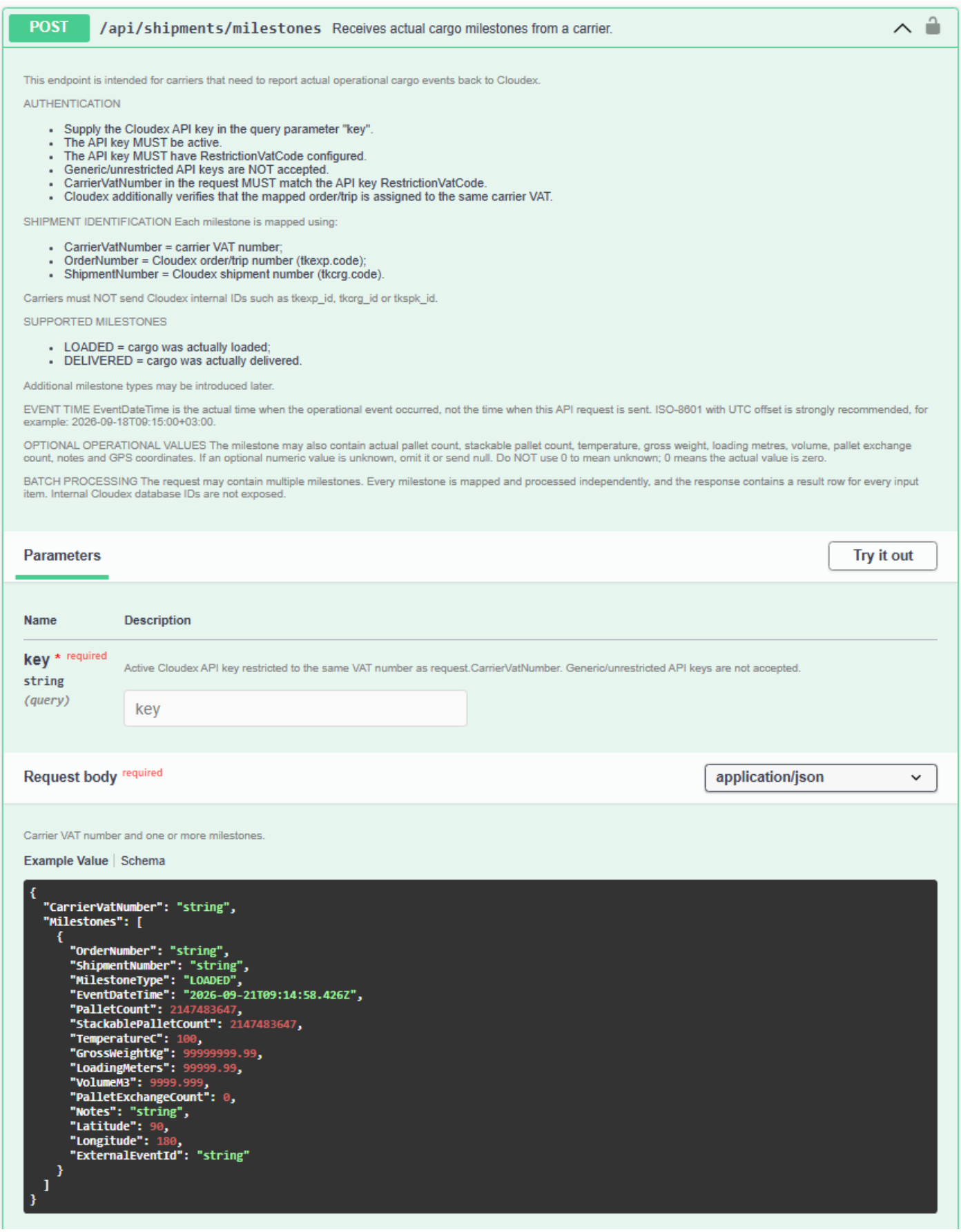
You also need:

- access to the Cloudex TMS API;
- a valid Cloudex API key.

Full Cloudex TMS API documentation: https://my.cloudex.lv/cloudex-api/swagger/index.html

Which endpoint should be used?

To send shipment milestone/status updates to Cloudex TMS, use the: **milestones** endpoint.



POST /api/shipments/milestones Receives actual cargo milestones from a carrier.

This endpoint is intended for carriers that need to report actual operational cargo events back to Cloudex.

AUTHENTICATION

- Supply the Cloudex API key in the query parameter "key".
- The API key MUST be active.
- The API key MUST have RestrictionVatCode configured.
- Generic/unrestricted API keys are NOT accepted.
- CarrierVatNumber in the request MUST match the API key RestrictionVatCode.
- Cloudex additionally verifies that the mapped order/trip is assigned to the same carrier VAT.

SHIPMENT IDENTIFICATION Each milestone is mapped using:

- CarrierVatNumber = carrier VAT number;
- OrderNumber = Cloudex order/trip number (tkexp.code);
- ShipmentNumber = Cloudex shipment number (tkcrg.code).

Carriers must NOT send Cloudex internal IDs such as tkexp_id, tkcrg_id or tkspk_id.

SUPPORTED MILESTONES

- LOADED = cargo was actually loaded;
- DELIVERED = cargo was actually delivered.

Additional milestone types may be introduced later.

EVENT TIME EventDateTime is the actual time when the operational event occurred, not the time when this API request is sent. ISO-8601 with UTC offset is strongly recommended, for example: 2026-09-18T09:15:00+03:00.

OPTIONAL OPERATIONAL VALUES The milestone may also contain actual pallet count, stackable pallet count, temperature, gross weight, loading metres, volume, pallet exchange count, notes and GPS coordinates. If an optional numeric value is unknown, omit it or send null. Do NOT use 0 to mean unknown; 0 means the actual value is zero.

BATCH PROCESSING The request may contain multiple milestones. Every milestone is mapped and processed independently, and the response contains a result row for every input item. Internal Cloudex database IDs are not exposed.

Parameters Try it out

Name Description

key * required
string
(query)

Active Cloudex API key restricted to the same VAT number as request.CarrierVatNumber. Generic/unrestricted API keys are not accepted.

key

Request body required application/json

Carrier VAT number and one or more milestones.

Example Value | Schema


```
{
  "CarrierVatNumber": "string",
  "Milestones": [
    {
      "OrderNumber": "string",
      "ShipmentNumber": "string",
      "MilestoneType": "LOADED",
      "EventDateTime": "2026-09-21T09:14:58.426Z",
      "PalletCount": 2147483647,
      "StackablePalletCount": 2147483647,
      "TemperatureC": 100,
      "GrossWeightKg": 99999999.99,
      "LoadingMeters": 99999.99,
      "VolumeM3": 9999.999,
      "PalletExchangeCount": 0,
      "Notes": "string",
      "Latitude": 90,
      "Longitude": 180,
      "ExternalEventId": "string"
    }
  ]
}
```


Integration flow

A typical integration works as follows: **External system → Shipment milestone/status → Cloudex TMS**

For example, when the shipment is marked as delivered in the connected external system, the corresponding milestone can be sent to Cloudex TMS.

This allows Cloudex TMS to keep the shipment status synchronized with the connected system without requiring users to update the shipment manually.

Related articles

- **How to send shipment documents to Cloudex TMS via API?**
- **How to create API key**

How to send shipment documents to Cloudex TMS via API?

Use the **Cloudex TMS API documents endpoint** when documents from an external system need to be attached to a shipment in Cloudex TMS.

Typical examples are:

- POD (Proof of Delivery);
- CMR;
- other shipment-related documents.

Before you start

The shipment must already exist in Cloudex TMS and the external system must be able to identify which shipment the document belongs to.

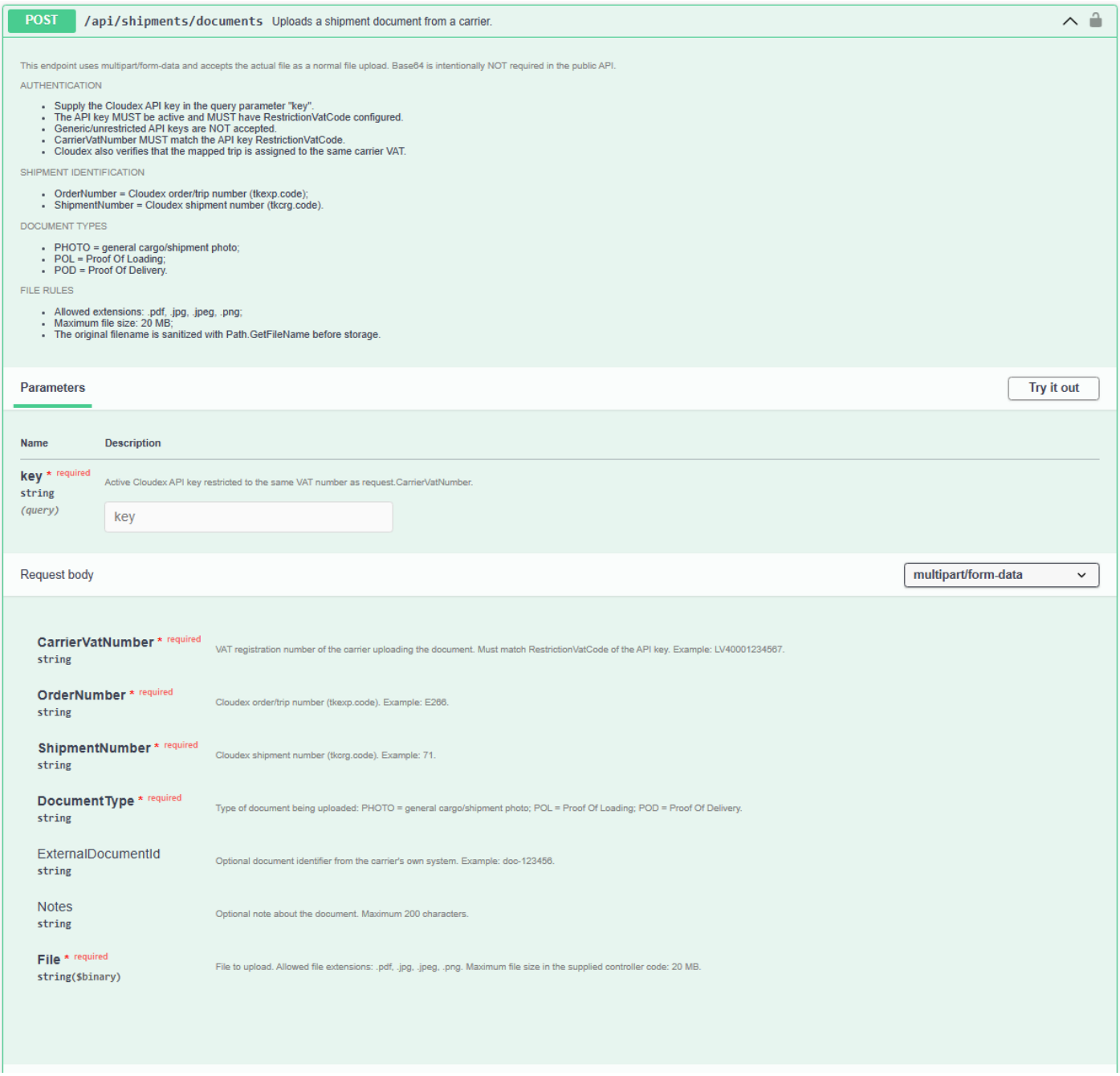
You also need:

- access to the Cloudex TMS API;
- a valid Cloudex API key.

Full Cloudex TMS API documentation: https://my.cloudex.lv/cloudex-api/swagger/index.html

Which endpoint should be used?

To send shipment documents to Cloudex TMS, use the: **documents** endpoint.



Integration flow

A typical document integration works as follows: **External system → Shipment document → Cloudex TMS shipment**

For example, once a signed POD or CMR becomes available in the connected system, it can be sent automatically to Cloudex TMS and linked to the corresponding shipment.

This reduces the need to manually download documents from one system and upload them again into Cloudex TMS.

Related articles

- **How to send shipment milestones to Cloudex TMS via API?**
- **How to create API key**

AI Plugin

AI Plugin

Cloudex TMS assistant API

For making Cloudex TMS assistant AI Plugin you need to use CLOUDEX TMS API

https://my.cloudex.lv/cloudex-api/swagger-ai-integrations/index.html