Table of Contents

Overview

The Partner Portal is an API experience management platform designed to simplify and standardize the user experience, enabling our partners to publish and consume APIs more efficiently. It catalogs existing API services, hosts API information and documentation, and streamlines access management/client credentialing.

The Partner Portal is an experience manager only and will not support API design, development, deployment, security, testing, analysis, or maintenance.

Table of Contents

Getting Started

About the Partner Portal API

The Patner Portal API can be used to programmatically interact with Centene's partner portal platform streamlining developers' workflows by eliminating the need to interact via the user interface.

The Patner Portal API is organized around REST. The API has predictable resource-oriented URLs, accepts form-encoded request bodies, returns JSON-encoded responses, and uses standard HTTP response codes, authentication, and verbs.

Getting Access

Centene's APIs handle enormous amounts of data. The way we ensure this data is secured for developers and users alike is through authentication. The first step in the authentication process is creating a new EntryKeyID account if you don't already have one.

Creating a new EntryKeyID account is quick and easy. Just click here, fill out your information, verify you are who you say you are, and then you can begin interacting on the Centene Partner Portal.

Table of Contents

API Errors

Centene uses conventional HTTP response codes to indicate the success or failure of an API request. In general: Codes in the `2xx` range indicate success. Codes in the `4xx` range indicate an error that failed given the information provided (e.g., a required parameter was omitted). Codes in the `5xx` range indicate an error with Centene's servers. Some `4xx` errors that could be handled programmatically (e.g., scope not recognized) include an error code that briefly explains the error reported.

Table of Contents

API Catalog

The API catalog allows all users to browse published public APIs, download OAS files, view endpoints, and view any additional documentation provided by the developer.

List APIs

`GET /apis`

- Returns a list of APIs available to you along with the attributes associated with them such as API name, ID, description, publish date, authentication type, etc.
- Supports sorting and limiting of items in the response

Get API by ID

`GET /apis/{id}`

- Returns a single API and its attributes matching the Api# found in the SK and PK key-value pairs.
- **NOTE**: You must truncate the SK and PK values as to exclude `Api#` (for SK) or `API-Api#` (for PK) leaving only the unique API ID string.

Get OAS File

`GET /apis/oas/{id}`

- Returns a link allowing you to download the OAS file via your browser.

Table of Contents

Publish and Update

The following calls allow producers to publish thier APIs, get a list of the APIs they've published, and update their APIs as necissary.

In order to successully make any of the following calls, you must be a member of Centene's producer group. To request this designation, submit a request via Access Central to the "partnerportal_producers" active directory group.

Publish API

`POST /apis`

- Publishes a new API to the Partner Portal.
- The body of your request must include the following form-data:
  - *file*: The OAS file in a **.json**, **.yaml**, or **.yml** file format.
  - *doc*: The additional documentation in a **.md** file format.
  - *data*: The API detail passed as raw JSON that includes the following key-value pairs:

```
{
    "name": "",
    "description": "",
    "type": "",
    "path": "",
    "version": "",
    "category": "",
```

```
    "visibility": "",
    "authentication": "",
    "apiDocTitle": "",
    "hostServers": [
        {
            "hostName": "",
            "serverName": "",
            "id": ""
        },
    ],
}
```

Key	Description	Value/Options
`"name":`	This will be the searchable name of your API displayed or returned to consumers browsing the catalog.	`"string"`
`"description":`	Will be the description of your API displayed or returned to consumers browsing the catalog.	`"string"`
`"path":`	Indicates the type of API you're publishing.	`"REST"` , `"GraphQL"`
`"version":`	Indicates the version of your API.	`"string"`
`"category":`	Indicates what consumer group the API serves.	`"Patient"` , `"Provider"`
`"visibility":`	Determines who can see the API.	`"Public"` , `"Private"` , `"Partner"` , `"Application"`
`"authentication":`	Defines what authentication method the API uses.	`"API Keys"` , `"Basic Auth"` , `"OAUTH 2.0"`
`"apiDocTitle":`	Will be the title of your API documentation displayed or returned to consumers browsing the catalog.	`"string"`

Key	Description	Value/Options
`"hostServers":`	Indicates what server(s) and environment(s) are supported by your API. You must pass at least one array defining the Server URL (hostName), type (serverName), and ID (id).	See next three (3) rows for attributes passed in the `"hostServers:"`
- `"hostName":`	<>	`"string"`
- `"serverName":`	<>	`"Production"`, `"Sandbox"`
- `"id":`	<>	`"string"`

- A successful call will return a success message and the published APIs ID in the response body:

```
{
    "message": "Insert Successful",
    "id": "51069cbf-7e26-4ec6-b5a2-bc11fcb060d6"
}
```

Update API

`PUT /apis/{id}`

- Updates an existing API by replacing existing form-data with new form-data
- Your call can contain any of the form-data fields you defined during the Publish API step.
- Any form-data omitted from the call will not be changed (existing form-data will persist)
- A successful call will return a success message in the response body:

```
{
    "message": "Update Successful"
}
```

Get My APIs

`GET /apis/producer`

- Returns a list of the APIs published by the user identity.
- Supports sorting and limiting the response.

Upload OAS File

`POST /apis/oas/upload`

Table of Contents

Access Management

Access management allows consumers to request access and get a list of thier requests. It allows producers to approve or deny access requests and get a list of the access requests made.

One of the highlights of the platform is no longer having to manually generate and communicate access credentials. This not only significantly increases security of the information but also makes the process highly efficient for both consumers and producers.

Request Access

`POST /subscriptions`

- Submits an access request for an API
- On success, triggers an email notification to both consumer and producer
- The body of the request must contain the following information:

```
{
  "apiId": "",
  "environment": "",
  "url": "",
  "serverId": ""
}
```

Key	Description	Value/Options
`"apiId":`	The unique API ID of which you are requesting access (truncated SK and PK values from the Get API by ID section)	`"string"`
`"environment":`	The server type of which you are requesting access	`"Production"` , `"Sandbox"`
`"url":`	The API server URL of which you are requesting access	`"string"`

Key	Description	Value/Options
`"serverId":`	The API server ID of which you are requesting access	`"string"`

Get Access Requests

Consumer View `GET /subscriptions/{id}`

- Returns an array of ***your*** access request(s) and their status(es) for a specific API

Producer View `GET /subscriptions/requests/{id}`

- Only available to producers
- Returns an array of ***all*** access request(s) and their status(es) for a specific API
- Includes more information around approval history and profile information for accountability

Approve/Deny Access

`PUT /subscriptions`

- Only available to producers
- Responds to an access request from a consumer
- The body of the request must contain the following information:

```
{
  "id": "",
  "status": "",
  "reason": ""
}
```

Key	Description	Value/Options
`"id":`	The ID of the request to which you are responding	`"string"`
`"status":`	Indicate whether or not the access request is to be approved or denied	`"Approved"` , `"Denied"`

Key	Description	Value/Options
`"reason":`	Provide the denial reason that will be included on the notification sent to the consumer.	`"string"`

Table of Contents

Client Management

Client management allows consumers to generate and edit client credentials used to access APIs for which they have been approved, as well as get a list of those clients.

Add Client

`POST /clients`

- Generates new client credentials used for authentication and access
- The body of the request must contain the following information:

```
{
  "apiId": "",
  "serverId": "",
  "url": "",
  "environment": "",
  "requestId": "",
  "projectId: "",
  "redirectURI": [
    ""
  ],
  "grantType": ""
}
```

Key	Description	Value/Options
`"apiId":`	The unique API ID of which you are requesting access (truncated SK and PK values from the Get API by ID section)	`"string"`

Key	Description	Value/Options
`"serverId":`	The API server ID of which you are requesting access	`"string"`
`"url":`	The API server URL of which you are requesting access	`"string"`
`"environment":`	The server type of which you are requesting access	`"Production"` , `"Sandbox"`
`"requestId":`	The ID of the request approved by the producer	`"string"`
`"projectId":`	The project ID returned when the request was approved	`"string"`
`"redirectURI":`	The URL the user will be redirected to…	`"string"`
`"grantType":`	The grant type that will be used to authenticate the client	`"string"`

Edit/Disable Client

`PUT /clients/{clientID}`

- This functionality is currently not working for backend interactions. To edit or disable a client you have added, use the partal portal website at https://partners.centene.com.

Get Client by Server ID

`GET /clients`

- Returns client details for a the server specified in the request

Table of Contents

Supporting Services

Send Email

`POST /sparkPostEmail`

- Sends an access request response email
- The body of the request must contain the following information:

```
{
    "templateId": "",
    "templateName": "",
    "email": "",
    "substitutionData": {
        "producerName": "",
        "consumerName": "",
        "serverName": "",
        "apiName": "",
        "reasonForDenial": "",
        "contactUs": ""
    }
}
```

Key	Description	Value/Options
`"templateId":`		
`"templateName":`		
`"email":`		
`"substitutionData":`		See next six (6) rows for attributes passed in the `"substitutionData":`
- `"producerName":`		
- `"consumerName":`		
- `"serverName":`		
- `"apiName":`		
- `"reasonForDenial":`		
- `"contactUs":`		

Get Catagories

- Returns a list of valid catagories

Health Check

API Sort

Get Grant Types

Get Methods

Visibility

Table of Contents

Resources

Table of Contents

Terms of Use

Table of Contents

Change Log

Table of Contents

Accessibility and Compatibility Features

Table of Contents

Missing Endpoints

- Publish and Update
  - View Endpoints
  - View Additional Documentation

Table of Contents