

OAuth 2.0 authorization

Authorize external requests to Creatio using OAuth 2.0 authorization

Version 8.0



This documentation is provided under restrictions on use and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this documentation, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

Table of Contents

Authorize external requests to Creatio using OAuth 2.0 authorization	4
1. Set up the integration of a third-party app with Creatio using Identity Service	4
2. Retrieve a Bearer token	6
3. Retrieve data from Creatio	8
Retrieve data from Creatio using OAuth 2.0 authorization and OData 4	10
1. Set up the integration of a third-party app with Creatio using Identity Service	12
2. Retrieve a Bearer token	12
3. Retrieve data from Creatio	13
Retrieve data from Creatio using OAuth 2.0 authorization and OData 3	13
1. Set up the integration of a third-party app with Creatio using Identity Service	14
2. Retrieve a Bearer token	14
3. Retrieve data from Creatio	14
Retrieve data from Creatio using OAuth 2.0 authorization and DataService	15
1. Set up the integration of a third-party app with Creatio using Identity Service	16
2. Retrieve a Bearer token	17
3. Retrieve data from Creatio	17
Retrieve data from Creatio using OAuth 2.0 authorization and web service	18
1. Set up the integration of a third-party app with Creatio using Identity Service	19
2. Retrieve a Bearer token	19
3. Retrieve data from Creatio	19

Authorize external requests to Creatio using OAuth 2.0 authorization



Use OAuth 2.0 protocol to securely authorize third-party apps and web services you integrate with Creatio. This technology does not pass Creatio logins and passwords to third-party apps. OAuth 2.0 also lets you restrict Creatio permissions for the integrated apps.

General procedure to authorize external requests to Creatio using OAuth 2.0 authorization:

1. Set up the integration of a third-party app with Creatio using Identity Service. [Read more >>>](#)
2. Retrieve a `Bearer` token. [Read more >>>](#)
3. Retrieve data from Creatio. [Read more >>>](#)

You can use Postman to test requests.

1. Set up the integration of a third-party app with Creatio using Identity Service

1. **Set up the Identity Service.** Instructions: [Set up the Identity Service](#).
2. **Set up the OAuth 2.0 authorization.** Instructions: [Set up the OAuth 2.0 authorization](#).
3. **Make sure the Identity Service is running.**
 - a. Add a request collection. Instructions: [Adding a collection of requests](#).
For example, add the `Identity Service` request collection.
 - b. Create a request.
 - a. Add a request to the collection. Instructions: [Adding requests to the collection](#).
 - b. Fill out the request parameters.

Parameter	Parameter description	Parameter value
Request method		GET
Request URL	The URL of the Identity Service website that lets you view the settings.	http://localhost:8090/.well-known/openid-configuration

```
Request

GET http://myidentityservice/.well-known/openid-configuration
```

- c. Save the request.
- c. Execute the request.

As a result, you will ensure the Identity Service is running.

Response

Status: ● 200 OK

```
{
  "issuer": "creatio.com",
  "jwks_uri": "http://localhost:8090/.well-known/openid-configuration/jwks",
  "authorization_endpoint": "http://localhost:8090/connect/authorize",
  "token_endpoint": "http://localhost:8090/connect/token",
  "userinfo_endpoint": "http://localhost:8090/connect/userinfo",
  "end_session_endpoint": "http://localhost:8090/connect/endsession",
  "check_session_iframe": "http://localhost:8090/connect/checksession",
  "revocation_endpoint": "http://localhost:8090/connect/revocation",
  "introspection_endpoint": "http://localhost:8090/connect/introspect",
  "device_authorization_endpoint": "http://localhost:8090/connect/deviceauthorization",
  "frontchannel_logout_supported": true,
  "frontchannel_logout_session_supported": true,
  "backchannel_logout_supported": true,
  "backchannel_logout_session_supported": true,
  "scopes_supported": [
    "openid",
    "profile",
    "email",
    "phone"
  ],
  "claims_supported": [
    "sub",
    "name",
    "family_name",
    "given_name",
    "middle_name",
    "nickname",
    "preferred_username",
    "profile",
    "picture",
    "website",
    "gender",
    "birthdate",
    "zoneinfo",
    "locale",
    "updated_at",
```

```

    "email",
    "email_verified",
    "phone_number",
    "phone_number_verified"
  ],
  "grant_types_supported": [
    "authorization_code",
    "client_credentials",
    "refresh_token",
    "implicit",
    "urn:ietf:params:oauth:grant-type:device_code"
  ],
  "response_types_supported": [
    "code",
    "token",
    "id_token",
    "id_token token",
    "code id_token",
    "code token",
    "code id_token token"
  ],
  "response_modes_supported": [
    "form_post",
    "query",
    "fragment"
  ],
  "token_endpoint_auth_methods_supported": [
    "client_secret_basic",
    "client_secret_post"
  ],
  "id_token_signing_alg_values_supported": [
    "RS256"
  ],
  "subject_types_supported": [
    "public"
  ],
  "code_challenge_methods_supported": [
    "plain",
    "S256"
  ],
  "request_parameter_supported": true
}

```

2. Retrieve a Bearer token

A `Bearer` token is an auto-generated token required for external requests that have OAuth 2.0 authorization. Each `Bearer` token has a finite lifetime, 3600 seconds by default. The `[AccessTokenLifetime]` column in the

[Clients] database table stores the value of the token lifetime. To **retrieve the value of the token lifetime**, execute the following SQL query.

SQL query

```
select
  "ClientId",
  "AccessTokenLifetime"
from "Clients"
```

To **change the token lifetime**, execute the following SQL query.

SQL query

```
update "Clients"
set "AccessTokenLifetime" = New-Token-Lifetime
where "ClientId" = Some-Client-Id
```

To retrieve a `Bearer` token:

1. **Create a request** to retrieve a `Bearer` token.

- a. Add a request to the collection. Instructions: [Adding requests to the collection](#).
- b. Fill out the request parameters.

Parameter	Parameter description	Parameter value
Request method		POST
Request URL	The URL of the Identity Service website to retrieve a <code>Bearer</code> token.	http://localhost:8090/connect/token

Request

```
POST http://myidentityservice/connect/token
```

- c. Add the request body.
 - a. Open the [*Body*] tab.
 - b. Select `x-www-form-urlencoded` in the [*Body option*] parameter.
 - c. Fill out the body parameters.

Parameter ([Key] column)	Parameter description	Parameter value ([Value] column)
client_id	Use the "Client Id" parameter value that Creatio populates automatically. Learn more: Set up OAuth 2.0 authorization for third-party app.	4*****6
client_secret	Use the "Client secret" parameter value that Creatio populates automatically. Learn more: Set up OAuth 2.0 authorization for third-party app.	5*****C
grant_type		client_credentials

d. Save the request.

2. Execute the request.

As a result, you will retrieve the `Bearer` token.

```

Response

Status: ● 200 OK

{
  "access_token": "eyJhbGciOiJSUzI1NiIsImtpZCI6Iky1MDgxRDYyNUE1QkZFRjJCRjQ5RTBFQTY2NENDMERBOTk",
  "expires_in": 3600,
  "token_type": "Bearer",
  "scope": "ApplicationAccess_7010c58a958946dbaaa4734b0580c741"
}
    
```

You can now integrate third-party apps or web services with Creatio using OAuth 2.0 authorization. Authorize all external requests to Creatio using the `Bearer` token.

3. Retrieve data from Creatio

You can retrieve data from Creatio in multiple ways:

- **OData 4.** Detailed example: [Retrieve data from Creatio using OAuth 2.0 authorization and OData 4.](#)

- **OData 3.** Detailed example: [Retrieve data from Creatio using OAuth 2.0 authorization and OData 3.](#)
- **DataService.** Detailed example: [Retrieve data from Creatio using OAuth 2.0 authorization and DataService.](#)
- **Web service.** Detailed example: [Retrieve data from Creatio using OAuth 2.0 authorization and web service.](#)

Regardless of the chosen method, to retrieve data from Creatio:

1. **Create a request** to retrieve data.

- Add a request to the collection. Instructions: [Adding requests to the collection.](#)
- Fill out the request parameters:
 - [*Request method*] **required**
 - [*Request URL*] **required**
 - Request body parameters. For example, [*Body option*], [*Body type*], [*Body value*].
- Save the request.

2. **Set up the authorization.**

- Open the [*Authorization*] tab.
- Fill out the authorization parameters.

Parameter	Parameter value
Type	OAuth 2.0
Token	e*****w

- Save the request.

As a result, the auto-generated header will be added to the [*Headers*] tab. The header has the following parameters.

Parameter	Parameter value
Key	Authorization
Value	Bearer e*****w

KEY	VALUE	DESCRIPTION	...	Bulk Edit	Presets
<input checked="" type="checkbox"/> Authorization ⓘ	Bearer eyJhbGciOiJIUzI1NiIsImtpZCI6Iky1MDgxRDY...				
<input checked="" type="checkbox"/> Cache-Control ⓘ	no-cache				
<input checked="" type="checkbox"/> Postman-Token ⓘ	<calculated when request is sent>				
<input checked="" type="checkbox"/> Host ⓘ	<calculated when request is sent>				
<input checked="" type="checkbox"/> User-Agent ⓘ	PostmanRuntime/7.25.0				
<input checked="" type="checkbox"/> Accept ⓘ	*/*				
<input checked="" type="checkbox"/> Accept-Encoding ⓘ	gzip, deflate, br				
<input checked="" type="checkbox"/> Connection ⓘ	keep-alive				

3. Execute the request.

If the **token lifetime has expired**, i. e. you got the **401 Unauthorized** response, retrieve a new `Bearer` token.

Retrieve data from Creatio using OAuth 2.0 authorization and OData 4

 Medium

To implement the example:

Step 1: Set up the integration of a third-party app with Creatio using Identity Service. [Read more >>>](#)

Step 2: Retrieve a `Bearer` token. [Read more >>>](#)

Step 3: Retrieve data from Creatio. [Read more >>>](#)

Example. Retrieve the list of all accounts from the [*Accounts*] section. Use OAuth 2.0 authorization and OData 4 protocol to integrate the app with Creatio.

Response

Status: ● 200 OK

```
{
  "@odata.context": "http://mycreatio.com/0/odata/$metadata#Account",
  "value": [
    {
      "Id": "405947d0-2ffb-4ded-8675-0475f19f5a81",
      "Name": "Accom (sample)",
      "OwnerId": "410006e1-ca4e-4502-a9ec-e54d922d2c00",
      "CreatedOn": "2022-12-02T08:52:59Z",
      "CreatedById": "410006e1-ca4e-4502-a9ec-e54d922d2c00",
      "ModifiedOn": "2023-08-07T18:36:28.92Z",
    }
  ]
}
```

```

"ModifiedById": "410006e1-ca4e-4502-a9ec-e54d922d2c00",
"ProcessListeners": 0,
"OwnershipId": "f7af2e24-f46b-1410-fb98-00155d043204",
"PrimaryContactId": "c4ed336c-3e9b-40fe-8b82-5632476472b4",
"ParentId": "00000000-0000-0000-0000-000000000000",
"IndustryId": "fbbf0e52-f36b-1410-c493-00155d043205",
"Code": "1",
"TypeId": "03a75490-53e6-df11-971b-001d60e938c6",
"Phone": "+1 617 440 2498",
"AdditionalPhone": "",
"Fax": "",
"Web": "ac.com",
"AddressTypeId": "780bf68c-4b6e-df11-b988-001d60e938c6",
"Address": "31 Union Street",
"CityId": "74144617-a288-4984-bcac-e78733277a61",
"RegionId": "658acfe7-09ae-4747-b1b3-541e2dfa1b9e",
"Zip": "02111",
"CountryId": "e0be1264-f36b-1410-fa98-00155d043204",
"AccountCategoryId": "38ea507c-55e6-df11-971b-001d60e938c6",
"EmployeesNumberId": "ef7f0eb0-f36b-1410-849f-0026185bfc3",
"AnnualRevenueId": "bc30bdaa-55e6-df11-971b-001d60e938c6",
"Notes": "",
"Logo@odata.mediaEditLink": "Account(405947d0-2ffb-4ded-8675-0475f19f5a81)/Logo",
"Logo@odata.mediaReadLink": "Account(405947d0-2ffb-4ded-8675-0475f19f5a81)/Logo",
"Logo@odata.mediaContentType": "application/octet-stream",
"AlternativeName": "Accom-Westhouse Company",
"GPSN": "",
"GPSE": "",
"PriceListId": "00000000-0000-0000-0000-000000000000",
"Completeness": 95,
"AccountLogoId": "00000000-0000-0000-0000-000000000000",
"AUM": ""
},
{
  "Id": "e308b781-3c5b-4ecb-89ef-5c1ed4da488e",
  "Name": "Our company",
  "OwnerId": "00000000-0000-0000-0000-000000000000",
  "CreatedOn": "2011-01-30T15:22:43.771Z",
  "CreatedById": "22c5540c-d9b1-49ef-8eb7-72419b78e57c",
  "ModifiedOn": "2023-08-07T18:36:28.92Z",
  "ModifiedById": "410006e1-ca4e-4502-a9ec-e54d922d2c00",
  "ProcessListeners": 0,
  "OwnershipId": "00000000-0000-0000-0000-000000000000",
  "PrimaryContactId": "00000000-0000-0000-0000-000000000000",
  "ParentId": "00000000-0000-0000-0000-000000000000",
  "IndustryId": "00000000-0000-0000-0000-000000000000",
  "Code": "43",
  "TypeId": "57412fad-53e6-df11-971b-001d60e938c6",
  "Phone": ""
}

```

```

    "AdditionalPhone": "",
    "Fax": "",
    "Web": "",
    "AddressTypeId": "00000000-0000-0000-0000-000000000000",
    "Address": "",
    "CityId": "00000000-0000-0000-0000-000000000000",
    "RegionId": "00000000-0000-0000-0000-000000000000",
    "Zip": "",
    "CountryId": "00000000-0000-0000-0000-000000000000",
    "AccountCategoryId": "00000000-0000-0000-0000-000000000000",
    "EmployeesNumberId": "00000000-0000-0000-0000-000000000000",
    "AnnualRevenueId": "00000000-0000-0000-0000-000000000000",
    "Notes": "",
    "Logo@odata.mediaEditLink": "Account(e308b781-3c5b-4ecb-89ef-5c1ed4da488e)/Log",
    "Logo@odata.mediaReadLink": "Account(e308b781-3c5b-4ecb-89ef-5c1ed4da488e)/Log",
    "Logo@odata.mediaContentType": "application/octet-stream",
    "AlternativeName": "",
    "GPSN": "",
    "GPSE": "",
    "PriceListId": "00000000-0000-0000-0000-000000000000",
    "Completeness": 0,
    "AccountLogoId": "00000000-0000-0000-0000-000000000000",
    "AUM": ""
  }
]
}

```

This example uses Postman to test requests.

1. Set up the integration of a third-party app with Creatio using Identity Service

1. **Set up the Identity Service.** Instructions: [Set up the Identity Service](#).
2. **Set up the OAuth 2.0 authorization.** Instructions: [Set up the OAuth 2.0 authorization](#).
3. **Make sure the Identity Service is running.** Instructions: [Set up the integration of a third-party app with Creatio using Identity Service](#) (step 3).

2. Retrieve a Bearer token

Instructions: [Retrieve a Bearer token](#).

You can now integrate third-party apps or web services with Creatio using OAuth 2.0 authorization. Authorize all external requests to Creatio using the `Bearer` token.

3. Retrieve data from Creatio

1. **Create a request** to retrieve data using OData 4.

- Add a request to the collection. Instructions: [Adding requests to the collection](#).
- Fill out the request parameters.

Parameter	Parameter description	Parameter value
Request method		GET
Request URL	The URL of the Creatio instance to retrieve data.	http://mycreatio.com/0/odata/Account

Request

```
GET http://mycreatio.com/0/odata/Account
```

c. Save the request.

2. **Set up the authorization.** Instructions: [Retrieve data from Creatio](#) (step 2).

3. **Execute the request.**

As a result, you will retrieve the list of all accounts. [View result >>>](#)

Retrieve data from Creatio using OAuth 2.0 authorization and OData 3

 Medium

To implement the example:

Step 1: Set up the integration of a third-party app with Creatio using Identity Service. [Read more >>>](#)

Step 2: Retrieve a `Bearer` token. [Read more >>>](#)

Step 3: Retrieve data from Creatio. [Read more >>>](#)

Example. Retrieve the names of all accounts from the [*Accounts*] section. Use OAuth 2.0 authorization and OData 3 protocol to integrate the app with Creatio.

Response

Status: ● 200 OK

```
{
```

```

"d": {
  "results": [
    {
      "__metadata": {
        "id": "http://mycreatio.com/0/ServiceModel/EntityDataService.svc/Accour
        "uri": "http://mycreatio.com/0/ServiceModel/EntityDataService.svc/Accoi
        "type": "Terrasoft.Configuration.Account"
      },
      "Name": "Accom (sample)"
    },
    {
      "__metadata": {
        "id": "http://mycreatio.com/0/ServiceModel/EntityDataService.svc/Accour
        "uri": "http://mycreatio.com/0/ServiceModel/EntityDataService.svc/Accoi
        "type": "Terrasoft.Configuration.Account"
      },
      "Name": "Our company"
    }
  ]
}

```

This example uses Postman to test requests.

1. Set up the integration of a third-party app with Creatio using Identity Service

1. **Set up the Identity Service.** Instructions: [Set up the Identity Service](#).
2. **Set up the OAuth 2.0 authorization.** Instructions: [Set up the OAuth 2.0 authorization](#).
3. **Make sure the Identity Service is running.** Instructions: [Set up the integration of a third-party app with Creatio using Identity Service](#) (step 3).

2. Retrieve a Bearer token

Instructions: [Retrieve a Bearer token](#).

You can now integrate third-party apps or web services with Creatio using OAuth 2.0 authorization. Authorize all external requests to Creatio using the `Bearer` token.

3. Retrieve data from Creatio

1. **Create a request** to retrieve data using OData 3.
 - a. Add a request to the collection. Instructions: [Adding requests to the collection](#).

b. Fill out the request parameters.

Parameter	Parameter description	Parameter value
Request method		GET
Request URL	The URL of the Creatio instance to retrieve data.	http://mycreatio.com/0/ServiceModel/EntityDataService.svc/AccountCollection/?\$select=Name

Request

```
GET http://mycreatio.com/0/ServiceModel/EntityDataService.svc/AccountCollection/?$select=Name
```

c. Add the request headers.

- a. Open the [*Headers*] tab.
- b. Fill out the body parameters.

Parameter ([<i>Key</i>] column)	Parameter value ([<i>Value</i>] column)
Accept	application/json;odata=verbose

d. Save the request.

2. **Set up the authorization.** Instructions: [Retrieve data from Creatio](#) (step 2).
3. **Execute the request.**

As a result, you will retrieve the names of all accounts. [View result >>>](#)

Retrieve data from Creatio using OAuth 2.0 authorization and DataService



To implement the example:

- Step 1:** Set up the integration of a third-party app with Creatio using Identity Service. [Read more >>>](#)
- Step 2:** Retrieve a `Bearer` token. [Read more >>>](#)
- Step 3:** Retrieve data from Creatio. [Read more >>>](#)

Example. Retrieve the names of all accounts from the [*Accounts*] section. Use OAuth 2.0 authorization

and DataService to integrate the app with Creatio.

Response

Status: ● 200 OK

```
{
  "rowConfig": {
    "Id": {
      "dataValueType": 0
    },
    "Name": {
      "dataValueType": 1
    },
    "AccountLogo": {
      "dataValueType": 16,
      "isLookup": true,
      "referenceSchemaName": "SysImage"
    }
  },
  "rows": [
    {
      "Name": "Accom (sample)",
      "Id": "405947d0-2ffb-4ded-8675-0475f19f5a81",
      "AccountLogo": ""
    },
    {
      "Name": "Our company",
      "Id": "e308b781-3c5b-4ecb-89ef-5c1ed4da488e",
      "AccountLogo": ""
    }
  ],
  "notFoundColumns": [],
  "rowsAffected": 2,
  "nextPrceElReady": false,
  "success": true
}
```

This example uses Postman to test requests.

1. Set up the integration of a third-party app with Creatio using Identity Service

1. **Set up the Identity Service.** Instructions: [Set up the Identity Service](#).
2. **Set up the OAuth 2.0 authorization.** Instructions: [Set up the OAuth 2.0 authorization](#).

3. **Make sure the Identity Service is running.** Instructions: [Set up the integration of a third-party app with Creatio using Identity Service](#) (step 3).

2. Retrieve a Bearer token

Instructions: [Retrieve a Bearer token](#).

You can now integrate third-party apps or web services with Creatio using OAuth 2.0 authorization. Authorize all external requests to Creatio using the `Bearer` token.

3. Retrieve data from Creatio

1. **Create a request** to retrieve data using DataService.
 - a. Add a request to the collection. Instructions: [Adding requests to the collection](#).
 - b. Fill out the request parameters.

Parameter	Parameter description	Parameter value
Request method		POST
Request URL	The URL of the Creatio instance to retrieve data.	<code>http://mycreatio.com/0/DataService/json/SyncReply/SelectQuery</code>

- c. Add the request body.
 - a. Open the [*Body*] tab.
 - b. Fill out the body parameters.

Parameter ([Key] column)	Parameter value ([Value] column)
Body option	raw
Body type	JSON
Body value	<pre> { "RootSchemaName": "Account", "columns": { "items": { "Name": { "expression": { "columnPath": "Name" } } } } } </pre>

Request

POST <http://mycreatio.com/0/DataService/json/SyncReply/SelectQuery>

```

{
  "RootSchemaName": "Account",
  "columns": {
    "items": {
      "Name": {
        "expression": {
          "columnPath": "Name"
        }
      }
    }
  }
}

```

2. **Set up the authorization.** Instructions: [Retrieve data from Creatio](#) (step 2).
3. **Execute the request.**

As a result, you will retrieve the names of all accounts. [View result >>>](#)

Retrieve data from Creatio using OAuth 2.0

authorization and web service



To implement the example:

Step 1: Set up the integration of a third-party app with Creatio using Identity Service. [Read more >>>](#)

Step 2: Retrieve a `Bearer` token. [Read more >>>](#)

Step 3: Retrieve data from Creatio. [Read more >>>](#)

Example. Retrieve the converted `test` value. To convert the value, use the `GetConvertedPasswordValue` endpoint in the `CryptographicService` Creatio REST web service. Use OAuth 2.0 authorization to integrate the app with Creatio.

Response

Status: ● 200 OK

```
{
  "GetConvertedPasswordValueResult": "iOTfMVCYCjk="
}
```

This example uses Postman to test requests.

1. Set up the integration of a third-party app with Creatio using Identity Service

1. **Set up the Identity Service.** Instructions: [Set up the Identity Service](#).
2. **Set up the OAuth 2.0 authorization.** Instructions: [Set up the OAuth 2.0 authorization](#).
3. **Make sure the Identity Service is running.** Instructions: [Set up the integration of a third-party app with Creatio using Identity Service](#) (step 3).

2. Retrieve a Bearer token

Instructions: [Retrieve a `Bearer` token](#).

You can now integrate third-party apps or web services with Creatio using OAuth 2.0 authorization. Authorize all external requests to Creatio using the `Bearer` token.

3. Retrieve data from Creatio

1. **Create a request** to retrieve data.

- a. Add a request to the collection. Instructions: [Adding requests to the collection](#).
- b. Fill out the request parameters.

Parameter	Parameter description	Parameter value
Request method		POST
Request URL	The URL of the Creatio instance to retrieve data.	http://mycreatio.com/0/rest/CryptographicService/GetConvertedPasswordValue

- c. Add the request body.
 - a. Open the [*Body*] tab.
 - b. Fill out the body parameters.

Parameter ([<i>Key</i>] column)	Parameter value ([<i>Value</i>] column)
Body option	raw
Body type	JSON
Body value	<pre>{ "password": "test" }</pre>

Request

POST http://mycreatio.com/0/rest/CryptographicService/GetConvertedPasswordValue

```
{
  "password": "test"
}
```

2. **Set up the authorization.** Instructions: [Retrieve data from Creatio](#) (step 2).
3. **Execute the request.**

As a result, you will retrieve the `test` value converted by the `CryptographicService` web service.

[View result >>>](#)