Zenegy uses OAuth 2.0 for user authorization and API authentication. Applications must be authorized and authenticated before they can fetch and post data from and to Zenegy or get access to data.

Zenegy provides two environments production and test. Environments have separate data and authentication. Authentication server urls are:

Parameter	Description	Sandbox
Production	https://auth.zenegy.com/	No
Test	https://alpha-oauth.zalary.com/	Yes

Configure Your Application

Before starting with authentication you need to sign up for application. You need to supply application and company relevant information.

Parameter	Description	Required
Company GUID	Id of the company account that will be your developer account. If you don't have company account yet you need to sign up first	Yes
App name	Name of the application. This name will show in the authorization window	Yes
App description	Description of your application. If published to the application market this will be show as details	Yes
Install url	Valid URL link that user will be redirected to when he wants to install the app.	No
Redirect url	Valid URL link(s) that will be used as redirect urls in the authentication flows. URL has to be absolute, secure(https), url arguments are ignored and wild card urls are possible ex. https//app.zenegy.com/* . At least one URL is required if you want to use authorization code flow	No

Zenegy will provide you with a unique Client ID and Client Secret. Make note of these values as they have to be integrated into the configuration files or the actual code of your application.

Example:

Parameter	Value
Client ID	8f1f13bc250846109acfd5fc2fe6deec
Client Secret	YzFmZGRlZjZmYTA5NDcxM2FiMjlkZTYyMWY4NjEzYzY=

Important
Your Client Secret protects your application's security so be sure to keep it secure! Do not share your Client Secret value with anyone, including posting it in support forums for help with your application.

Authorization Code Flow

If you need to gain refresh token for long access to the company Authorization code flow is recommended. Authorization code flow returns short lived access_token and refresh token which can be used for acquiring access_token for unlimited period. This flow involves several steps.

Step 1: Request an Authorization Code

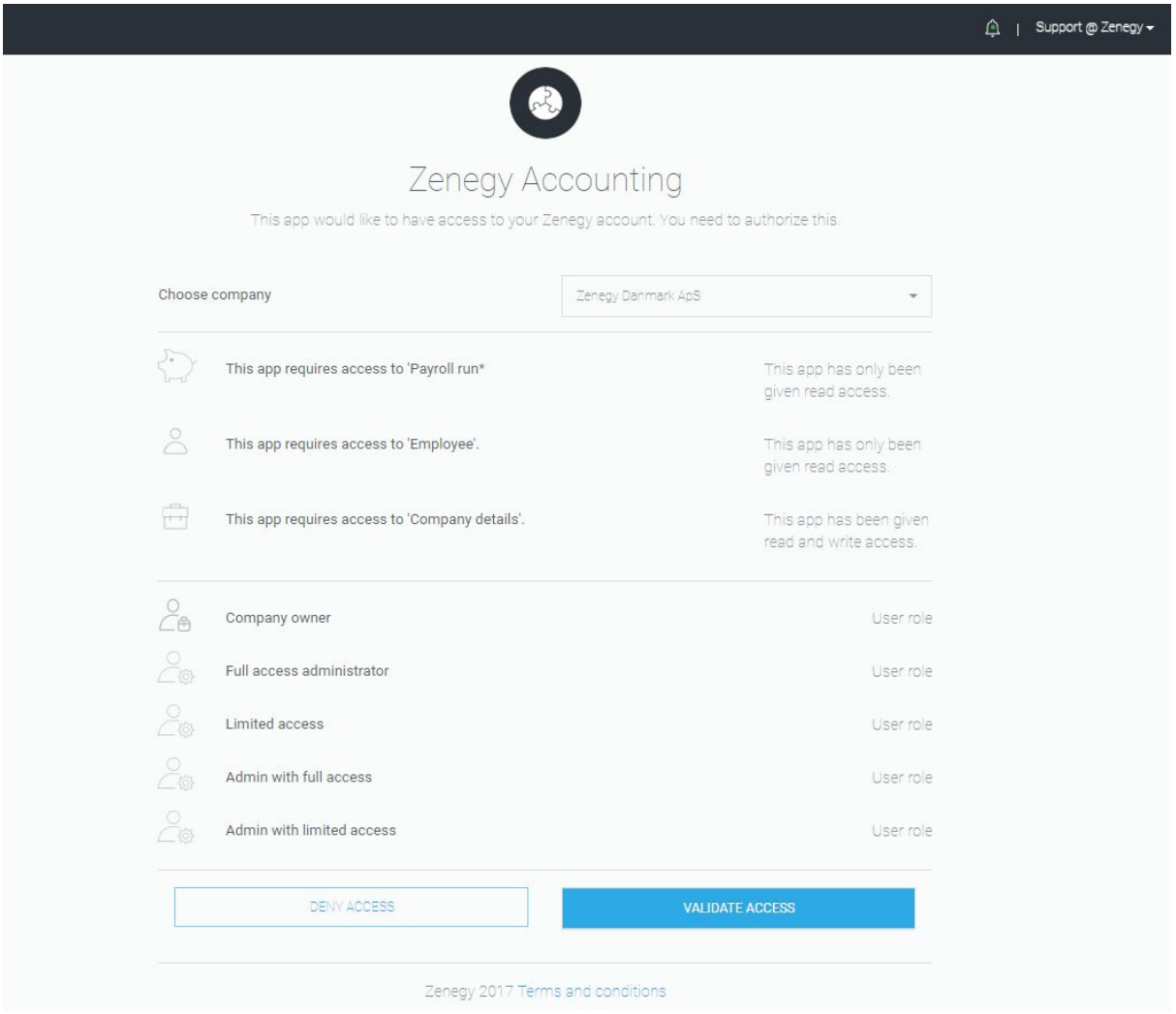
Required parameters for this flow are:

Parameter	Description	Required
response_type	The value of this field should always be: code	Yes
client_id	Unique id of the application, generated by Zenegy	Yes
redirect_uri	The URI your users are sent back to after authorization. This value must match one of the *OAuth 2.0 Authorized Redirect URLs.* Redirect url has to be https	Yes
company_id	Guid of the company for which access is required, if user has access to the company this company will be pre selected	No

Example:

https://auth.zenegy.com/auth/authorize?client_id={{client_id}}redirect_uri={{redirect_uri}}&response_type=code

Once redirected, the user is presented with Zenegy authentication screen. On this screen application name,logo,access scopes and required roles are presented to the user.
Users need to select a company for which access will be granted. Users can validate access(grant access) or Deny access.



Your Application is Approved

By providing valid Zenegy credentials and clicking Validate Access, the user approves your application's request to access the interact with Zenegy on their behalf. This approval instructs Zenegy to redirect the member to the callback URL that you defined in your redirect_uri parameter.

Attached to the redirect_uri is the OAuth 2.0 authorization code. Parameter code is returned as query param.
Example:

{{redirect_uri}}?code=308a4647ab394ea0a4e19c6956f8f067ca8ddded94b04ad3b8a513673d17475c

The code is a value that you exchange with Zenegy for an OAuth 2.0 access token in the next step of the authentication process. For security reasons, the authorization code has a 5 -minute lifespan and must be used immediately. If it expires, you must repeat all of the previous steps to request another authorization code.

Your Application is Rejected

If the member chooses to cancel, or the request fails for any reason, the client is redirected to your redirect_uri callback URL with no code attached.

Step 2: Exchange Authorization Code for an Access Token

The next step is to get an access token for your application using the authorization code from the previous step. To do this, make the following HTTP POST request with a Content-Type header of x-www-form-urlencoded:

Example

```
POST /auth/token HTTP/1.1
Host: auth.zenegy.com
Content-Type: application/x-www-form-urlencoded

grant_type=authorization_code&code={authorization_code_from_step2_respon
se}&redirect_uri={{redirect_uri}}&client_id={your_client_id}&client_secr
et={your_client_secret}
```

Parameter	**Description**
access_token	The access token for the application. Token life is 1 hour.
expires_in	The number of seconds remaining until the token expires. Currently, all access tokens are issued with a 1 hour lifespan.
token_type	Always Bearer
refresh_token	Refresh token, that is used in the refresh token flow for getting new

	access tokens
company_id	Guid of the company for which access is granted

Example:


```
{
    "access_token":
"eyJhbGciOiJodHRwOi8vd3d3LnczLm9yZy8yMDAxLzA0L3htbGRzaWctbW9yZSNobWFjLXNoYTI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6IjEiLCJ1aWQiOiJhNjM4ZDBhOS02NDhkLTQwYmUtYmMwMi1mYmY5MTQ2OWQ2GjZiTCJuYW1lIjoiU3VwcG9ydCBAIFplbmVneSIsIm1vYmlsZV9waG9uZV92ZXJpZmllZCI6IlRydWUiLCJtb2JpbGVfcGhvbmUiOiIrMzg5NzU0NTc0MDUiLCJlbWFpbEFkZHJlc3MiOiJzdXBwb3J0X3RlYW1AemVuZWd5LmNvbSIsImxhbmd1YWdlIjoiZW4iLCJhdXRoX3Byb3ZpZGVyIjoie1wiVHlwZVwiOjEsXCJJZGVudGlmaWNhdGlvblwiOm51bGx9IiwidXNlcl9hcHBfZ3JhbnRfaWQiOiI0MjgyNyIsImFwcGxpY2F0aW9uX25hbWUiOiJaZW5lZ3kgQWNjb3VudGluZyIsInRva2VuX2lkIjoiMko1YXVGWDlPOXAwWjJOOVFBZXJXTlFMTFVuU25YamciLCJuYmYiOjE1Nzg1Nzk4MzksImV4cCI6MTU3ODU4MzQzOSwiaXNzIjoiaHR0cHM6Ly9hdXRoLnphbGFyeS5jb20vIiwiYXVkIjoiY2Q3MWQ1Y2EzNzRjNDliZGFjMTExNTBjZGRjY2VjMTcifQ.rnYvYVwLCyUn7EtwQgF1KLNfdkUxxcMGx0KECFq7DpQ",
    "token_type": "bearer",
    "expires_in": 3599,
    "refresh_token": "OWViZWNhYWE4NDkzNDNkZDhjYjQ3M2Q1NzI0MmM5OTM=",
    "company_id": "ba8d4080-5828-42d1-a702-96615b527c67"
}
```


Step 3: Make Authenticated Requests

Once you've obtained an access token, you can start making authenticated API requests on behalf of the user by including an Authorization header in the HTTP call to Zenegy API.

```
GET /api/companies/{company_id} HTTP/1.1
Host: api.zenegy.com
Authorization: Bearer {access_token}
```

Handling Invalid Tokens

If you make an API call using an invalid token, you'll receive a 401 Unauthorized response from the server, and you'll have to regenerate the token. A token could be invalid due to the following reasons:

- It has expired.
- The member revoked the permission they initially granted to your application.

A predictable expiry time is not the only contributing factor to an invalid token so it's very important that you code your applications to properly handle a 401 Unauthorized error by redirecting the member back to the start of the authorization workflow.

Refresh Tokens

Parameter	Description
grant_type	The value of this field should always be refresh_token.
refresh_token	The number of seconds remaining until the token expires. Currently, all access tokens are issued with a 1 hour lifespan.
client_id	Unique id of the application, generated by Zenegy when you registered your application.
client_secret	The Client Secret value generated by Zenegy when you registered your application.

Example:

```
POST /auth/token HTTP/1.1
Host: auth.zenegy.com
Content-Type: application/x-www-form-urlencoded

grant_type=refresh_token&refresh_token=onN6Fvu9JM9HfOYWY15MSDo2PRqlbx1xSV9d+nO613g=&client_id={{client_id}}&client_secret={{secret}}
```

Result:

```
{
    "access_token":
```

```
"eyJhbGciOiJodHRwOi8vd3d3LnczLm9yZy8yMDAxLzA0L3htbGRzaWctbW9yZSNobWFjLXNoYTI1NiIsInR5cCI6Ik
pXVCJ9.eyZpZCI6IjEiLCJ1aWQiOiJhNjM4ZDBhOS02NDhkLTQwYmUtYmMwMi1mYmY5MTQ2OWQ2MjMiLCJuYW1lIjoi
U3VwcG9ydCBAIFplbmVneSIsIm1vYmlsZV9waG9uZV92ZXJpZmllZCI6IlRydWUiLCJtb2JpbGVfcGhvbmUiOiIrMzg
5NzU0NTc0MDUiLCJlbWFpbEFkZHJlc3MiOiJzdXBwb3J0X3RlYW1AemVuZWd5LmNvbSIsImxhbmd1YWdlIjoiZW4iLC
JhdXRoX3Gyb3ZpZGVyIjoie1wiVHlwZVwiOjEsXCJJZGVudGlmaWNhdGlvblwiOm51bGx9IiwidXNlcl9hcHBfZ3Jhb
nRfaWQiOiI0MjgyNyIsImFwcGxpY2F0aW9uX25hbWUiOiJaZW5lZ3kgQWNjb3VudGluZyIsInRva2VuX2lkIjoiU2tD
ZjhWdXlPYzAxZEVGNTE3VmJkdUZWM1lhVlV2VUQiLCJuYmYiOjE1Nzg1ODM0NDUsImV4cCI6MTU3ODU4NzA0NSwiaXN
zIjoiaHR0cHM6Ly9hdXRoLnphbGFyeS5jb20vIiwiYXVkIjoiY2Q3MWQ1Y2EzNzRjNDliZGFjMTExNTBjZGRjY2VjMT
cifQ.2oemzm1RdhdUQvGIpryQxLsVmlvfTIOAUiY63JAvQsI",
    "token_type": "bearer",
    "expires_in": 3599,
    "refresh_token": "OWViZWNhYWE4NDkzNDNkZDhjYjQ3M2Q1NzI0MmM5OTM=",
    "company_id": "ba8d4080-5828-42d1-a702-96615b527c67"
}
```