Client Credentials External

CareerBuilder oAuth 2.0 APIs

Background

In order to facilitate external customers accessing resources(Services) owned by CareerBuilder, we have implemented the OAuth 2.0 Client Credentials flow. To increase the security of this flow, we have augmented it with JWT-Bearer Client Assertions. In this augmented flow, you are given a shared secret and use it to sign a JSON Web Token(JWT) that is used to validate your client. This JWT contains your client id, the token endpoint URL, and an expiration. As a result of sending a valid JWT, you will receive an access token for our OAuth 2.0 protected APIs.

Application setup

Obtaining OAuth2 credentials

Partners need to request client credentials for their application from CareerBuilder. The client credentials include:

- client_id
- shared_secret
- environment: production

Obtaining an Access Token from CareerBuilder's Authorization server

Before your application can access CareerBuilder's API it needs to obtain an access token from the authorization server. The authorization server's URL is: https://api.careerbuilder.com/oauth/token

Create a JWT

Client credentials flow requires a signed JWT with the client_secret, which will be used by the authorization server to verify the validity of the request.

The JWT must contain the following JSON document format:

```
{
  "iss": "<client_id>",
  "sub": "<client_id>",
  "aud": "<token_url>",
  "exp": Time.now + 30s
}
```

Where exp is the expiration date of the JWT. The JWT is short lived and should only last for 30-60 seconds. The value is in epoch time.

For example:

```
{
  "iss": "abc12345",
  "sub": "abc12345",
  "aud": "https://api.careerbuilder.com/oauth/token",
  "exp": 1473378085
}
```

After encoding the JWT. You should get something similar to the following:

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJhYmMxMjMiLCJzdWIiOiJhYmM
xMjMiLCJhdWQiOiJodHRwczovL3d3d3Rlc3QuYXBpLmNhcmVlcmJ1aWxkZXIuY29tL29hdXR
oL3Rva2VuIiwiZXhwIjozMDB9.c1mdrM-PV6SOBcdajRqhQPal5qa_m00Lkw9dy6ZCU2o
```

Note: The JWT must be signed with the client_secret issued by Careerbuilder using a HS512 algorithm.

For example, in Ruby using the jwt gem:

```
JWT.encode(json_document, client_secret, 'HS512')
```

Get Access Token

After you obtained a JWT, your application can issue a POST request to get an access token.

Request format:

```
POST /oauth/token
Host: api.careerbuilder.com
Content-Type: application/x-www-form-urlencoded

grant_type=client_credentials&
client_assertion_type=urn:ietf:params:oauth:client-assertion-type:jwt-be
arer&
client_id=<client_id>&
client_assertion=<jwt_signed_with_client_secret>
```

Response format:

```
{
  "access_token": "<access token>",
  "token_type": "bearer",
  "expires_in": "2100"
}
```

Example request:

```
curl -X POST -H "Cache-Control: no-cache" -H "Content-Type:
application/x-www-form-urlencoded" -d
'client_assertion_type=urn:ietf:params:oauth:client-assertion-type:jwt-b
earer&grant_type=client_credentials&client_id=abc12345&client_assertion=
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJhYmMxMjMiLCJzdWIiOiJhYmM
xMjMiLCJhdWQiOiJodHRwczovL3d3d3Rlc3QuYXBpLmNhcmVlcmJ1aWxkZXIuY29tL29hdXR
oL3Rva2VuIiwiZXhwIjozMDB9.c1mdrM-PV6SOBcdajRqhQPal5qa_m00Lkw9dy6ZCU2o'
"https://wwwtest.api.careerbuilder.com/oauth/token"
```

Example response:

```
{
  "access_token": "tabc1234567xyz",
  "token_type": "bearer",
  "expires_in": "2100"
}
```

Calling CareerBuilder using an Access Token

Once you successfully obtained an access token, your application is able to make API calls to CareerBuilder services. The access token must be added as a value in the Authorization header as follows:

```
Authorization: Bearer <access_token>
```

For example:

```
curl -X GET -H "Authorization:
Bearer tabc1234567xyz" "https://wwwtest.api.careerbuilder.com/[API-URI]"
```

Note: Only request a new bearer token when your current token has expired. Requesting a new token for each request you perform will result in your account being flagged for token generation abuse and being locked out.

More Information

For more information on OAuth 2.0, the Assertion Flow, the JWT-Bearer Assertion Flow, or JWTs, please see the following IETF specifications and drafts:

- OAuth 2.0: http://tools.ietf.org/html/rfc6749
- OAuth 2.0 Bearer Token: http://tools.ietf.org/html/rfc6750
- JWT-Bearer Client Assertion: http://tools.ietf.org/html/draft-ietf-oauth-jwt-bearer-06
- JWT: http://tools.ietf.org/html/draft-ietf-oauth-json-web-token-12
- JWS: http://tools.ietf.org/html/draft-ietf-jose-json-web-signature-17