



Deltek. ProPricer >

BOE Pro Web API Programming Guide

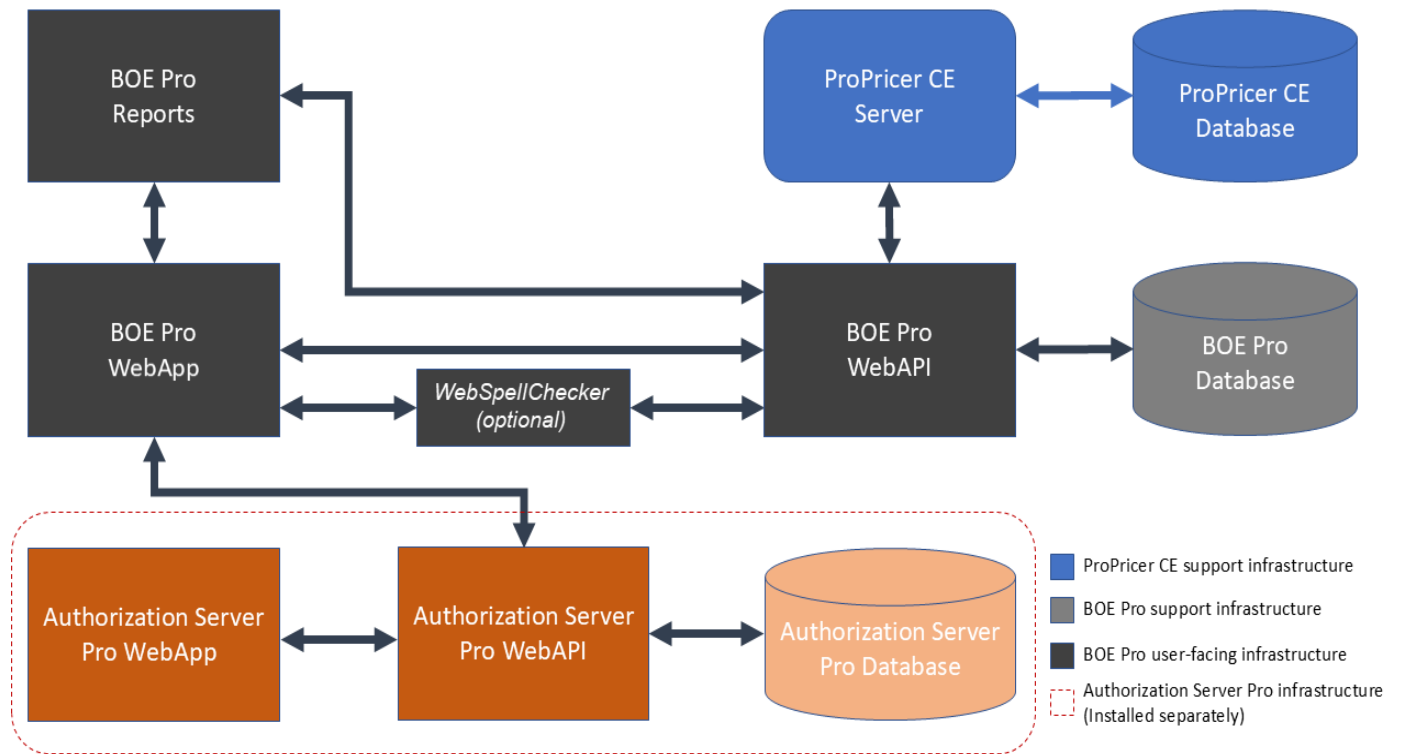
Contents

Introduction	1
Architecture	1
Requirements	2
Minimum requirements	2
Known limitations.....	2
Enable Resource Password Owner.....	3
Option 1: BOE Pro deployed in Windows (IIS)	3
Option 2: BOE Pro deployed in Azure	5
Obtain access token.....	6
WebAPI requests.....	9
References	16

Introduction

BOE Pro, an n-tier web application developed with ASP.NET Core, supports various deployments options. This guide focuses on how to use BOE Pro WebAPI programmatically.

Architecture



Requirements

Minimum requirements

- BOE Pro installed.
- Programming language or tool capable of HTTP requests.

Known limitations

- This guide uses Resource Owner Password (ROP) Flow (defined in [OAuth 2.0 RFC 6749, section 4.3](#)) which requires credentials (username and password). Because credentials are sent to the back end and can be stored for future use before being exchanged for an access token, it is imperative that the application is absolutely trusted with this information.

Because the ROP Flow involves the application handling the user's password, it must not be used by third-party clients.

Enable Resource Password Owner

By default, BOE Pro Authorization Server does not allow Resource Password Owner Flow. Depending on your deployment option, use one of the following two options to enable it.

Option 1: BOE Pro deployed in Windows (IIS)

1. Navigate to the folder where Authorization Server is installed (typically `C:\inetpub\BOEPro\AuthorizationServer`).
2. Using a text editor like Notepad, open the configuration file `appsettings.json`.
3. Add a new entry "ResourceOwnerPassword" in the `AllowedGrantTypes` array inside the `Clients` array. See the yellow highlighted portion of the following example.

Example

```
"Clients": [  
  {  
    "Id": "e29c00e2-15c2-4e8e-99c7-ec392d665670",  
    "Name": "BOE Pro",  
    "Secrets": [  
      "secret" ],  
    "AllowedScopes": [  
      "openid",  
      "profile",  
      "email",  
      "boeproapiScope",  
      "authorizationserverlivedataScope"  
    ],  
    "AllowedRedirectUrls": [  
      "https://boepro.mycompany.com/callback",  
      "https://boepro.mycompany.com/silent"  
    ],  
    "AllowedPostLogoutRedirectUrls": [  
      "https://boepro.mycompany.com/logout"  
    ],  
    "AllowedCorsOrigins": [  
      "https://boepro.mycompany.com"  
    ],  
    "AllowedGrantTypes": [  
      "Implicit",  
      "ResourceOwnerPassword"  
    ],  
    "RequireConsent": false,  
    "AllowAccessTokensViaBrowser": true,  
    "AccessTokenLifetime": 720  
  }  
]
```

Option 2: BOE Pro deployed in Azure

1. Log in to Azure portal.
2. Navigate to the BOE Pro Authorization Server WebApp.
3. Go to the Configuration section.
4. Add a new application setting:
 - Name: Clients:0:AllowedGrantTypes:1
 - Value: ResourceOwnerPassword

Example

Add/Edit application setting

Name

Value

Deployment slot setting

Obtain access token

To obtain the required access token to interact with WebAPI, send a POST request to your Authorization Server. Include an existing BOE Pro user's credentials (username and password) with the ProPricer login type.

Request

- Method: POST
- URL: `https://your-auth-boepro.mycompany.com/connect/token`
- Body parameters (url encoded):
 - `client_id=e29c00e2-15c2-4e8e-99c7-ec392d665670`
 - `client_secret=secret`
 - `grant_type=password`
 - `username=apiuser@mycompany.com`
 - `password=MyApiUserStrongPassword`
- Header:
 - `Content-Type: application/x-www-form-urlencoded`

Example using curl

```
curl -H "Content-Type: application/x-www-form-urlencoded" --data-urlencode "client_id=e29c00e2-15c2-4e8e-99c7-ec392d665670" --data-urlencode "client_secret=secret" --data-urlencode "grant_type=password" --data-urlencode "username=apiuser@mycompany.com" --data-urlencode "password=MyApiUserStrongPassword" -X POST https://your-auth-boepro.mycompany.com/connect/token
```

Sample response

```
{
  "access_token": "U3Sjeml\u002BvudgSIde2GaKr41H/\u002XBRqI6z6Ibw8D/ejm0=",
  "expires_in": 14400,
  "token_type": "Bearer",
  "scope": "authorizationserverlivedataScope boeproapiScope email openid profile"
}
```

Example using PowerShell

```
$params = @{username="apiuser@mycompany.com";client_id="e29c00e2-15c2-4e8e-99c7-ec392d665670";client_secret="secret";password="MyApiUserStrongPassword";grant_type="password"}

$response = Invoke-RestMethod -Uri https://your-auth-boepro.mycompany.com/connect/token -Method Post
-Body $params -ContentType 'application/x-www-form-urlencoded'

Write-Host
"Response:" Write-
Host $response


$token =
$response.access_token Write-
Host "Token:"
Write-Host $token
```




Sample response



```
Response:
@{access_token=U3Sjeml\u002BvudgSIde2GaKr41H/\u002XBRqI6z6Ibw8D/ejm0=; expires_in=14400;
token_type=Bearer; scope=authorizationserverlivedataScope boeproapiScope email openid profile}


Token: U3Sjeml\u002BvudgSIde2GaKr41H/\u002XBRqI6z6Ibw8D/ejm0=
```

Example using Postman

BOE Pro WebAPI / Get Token 



Save   





POST  https://boepro-stage-auth.██████████connect/token Send 

Params Authorization Headers (10) **Body**  Pre-request Script Tests Settings Cookies

none
 form-data
 x-www-form-urlencoded
 raw
 binary
 GraphQL

	Key	Value	Description	...	Bulk Edit
<input checked="" type="checkbox"/>	client_id	e29c00e2-15c2-4e8e-99c7-ec392d665670			
<input checked="" type="checkbox"/>	client_secret	secret			
<input checked="" type="checkbox"/>	grant_type	password			
<input checked="" type="checkbox"/>	username	██████████@mycompany.com			
<input checked="" type="checkbox"/>	password	████████████████████			
	Key	Value	Description		

Body Cookies (1) Headers (10) Test Results  200 OK 1082 ms 554 B Save Response 

Pretty Raw Preview Visualize JSON  
 

```

1  {
2    "access_token": "U3Sjeml+vudgSIde2GaKr41H/+BRqI6z6Ibw8D/ejm0=",
3    "expires_in": 14400,
4    "token_type": "Bearer",
5    "scope": "authorizationserverlivedataScope boeproapiScope email openid profile"
6  }
    
```

WebAPI requests

Use the access token obtained from Authorization Server as the Bearer token in the Authorization header when sending requests to WebAPI.

Request

- Method: GET or POST
- URL: `https://your-webapi-boepro.mycompany.com/api/controller/method`
- Header:
 - Authorization: Bearer YOUR-ACCESS-TOKEN

Example obtaining all proposals using curl

```
curl -H "Authorization: Bearer U3Sjem1\u002BvudgSIde2GaKr41H/\u002XBRqI6z6Ibw8D/ejm0=" -X  
GET https://your- webapi-boepro.mycompany.com/api/proposal/getProposals
```

Sample response

```
[
  {
    "id": "955d17f1-c655-4e5f-ae2c-ac4e1595af7d",
    "activityId": "80a96036-94e9-4a68-bf9e-
a5e9ecda67c7", "name": "Advanced Tank",
    "description": "Advanced Tank",
    "location": "test-folder\\",
    "locationId": "92cc7d92-8210-42fd-9c17-241a6bca56ac",
    "version": "0",
    "start": "2020-01-01T00:00:00",
    "end": "2022-03-31T00:00:00",
    "fiscalStart": "2020-01-01T00:00:00",
    "fiscalEnd": "2022-03-31T00:00:00",
    "duration": 27,
    "manager": "C.
Layton",
    "status": "New",
    "targetPrice":
0.0,
    "totalPriceDisplayName": "Total
Price", "dataServer": "ProPricer_SQL",
    "reportFooter": "Proprietary
Information", "headLine1": "[Company
Name]", "headLine2": "[Report Name]
Report", "timescale": "Month",
    "fiscalStartMonth": "JanThruDec",
    "fiscalYearDefinition": "LastMonth",
    "due": "2018-12-31T00:00:00",
    "isLocked": false,
    "dateType":
    "Month",
    "creator": "User
abc",
    "creationDate": "2023-03-08T15:39:53.2285733+02:00",
```

```
"lastUpdater": "User abc",
"lastUpdate": "2023-05-29T11:47:29.5400468+03:00",
"number": "2020 001",
"drtName": "Tank Directs",
"tags": [],
"owners": [
  "73cc51a9-b72e-4a34-85f1-5ed35f4efc6b"
],
"rfq": "AA-2020",
"precision": 0,
"unit": "DC Tank Works (DTW)",
"pbsLabel": "WBS",
"pbsSort": "Numeric",
"taskLabel": "TASK
ID", "taskSort":
"Alphabetic",
"textLink": "Any",
"accountingCalendar": "c2696764-0538-4d78-adf1-56c012167476",
"conflictedAccountingCalendar": null,
"accountingCalendarError": null,
"defaultAccountHours": 160,
"equivalentHeadDecimalPrecision": 3,
"profitFee": null,
"feeBase": "Cost",
"contractType": "CPIF",
"defaultBOETemplate": null,
"sendEmailNotifications":
false,
"enableDefaultBOEName":
false, "requireApproval":
true,
"drtDescription": "Advanced Tank Direct
Rates", "drtIsEmbedded": false,
"drtError": null,
```

```
"brtName": "Tank Indirects",
"brtDescription": "Advanced Tank Burden
Rates", "brtIsEmbedded": false,
"brtError": null,
"rbtName": null,
"rbtDescription":
null, "rbtIsEmbedded":
false, "rbtError":
null, "trtName": "Tank
Travel",
"trtDescription": "Advanced Tank Travel
Rates", "trtEnableWeeklyCarRate": true,
"trtEnableWeeklyCarRateQuantity": false,
"trtDaysPerWeek": 5.0,
"trtDaysPerMonth":
28.0, "trtIsEmbedded":
false, "trtError":
null, "peoplePerCar":
3,
"miscQtyFormula":
"PeopleMultiplyDays", "oneDayTrip":
1.0,
"firstDay": 1.0,
"lastDay": 1.0,
"hasTravelEstimates":
false, "type": "BOE",
"cftName": null,
"cftDescription":
null, "cftIsEmbedded":
false, "cftError":
null, "certName":
null,
"certDescription":
null,
"certIsEmbedded":
```

```
    false, "certError":  
    null,  
    "fontName": "Arial,Helvetica,sans-  
    serif", "fontSize": 14,  
    "lockFontSettings": false  
  }  
]
```

Example obtaining all proposals using PowerShell

```
$auth = @{Authorization="Bearer $token"}

$proposals = Invoke-RestMethod -Method Get -Headers $auth
-Uri https://your-webapi-boepro.mycompany.com/api/proposal/getProposals

$countProposals = $proposals.Count
Write-Host "Number of proposals: ${countProposals}"
$name = $proposals[0].name
$ver = $proposals[0].version
Write-Host "First proposal: '$name, Ver. $ver'"
```

Sample response

```
Number of proposals:
200
First proposal: 'Advanced Tank, Ver. 0'
```

Example using Postman

The screenshot shows the Postman interface for an API call. The URL is `https://boepro-stage-webapi-██████████/api/proposal/getProposals`. The request method is GET. The Authorization tab is selected, showing a Bearer Token: `U3Sjemi+vudgSlde2GaKr41H/+BRql6z6lbv...`. The response status is 200 OK, with a response time of 618 ms and a size of 328.36 KB. The response body is displayed in JSON format:

```
1  [
2    {
3      "id": "955d17f1-c655-4e5f-ae2c-ac4e1595af7d",
4      "activityId": "80a96036-94e9-4a68-bf9e-a5e9ecda67c7",
5      "name": "██████████ Advanced Tank CERs 3.5.100.1",
6      "description": "Advanced Tank",
7      "location": "██████████",
8      "locationId": "92cc7d92-8210-42fd-9c17-241a6bca56ac",
9      "version": "0",
10     "start": "2020-01-01T00:00:00",
11     "end": "2022-03-31T00:00:00",
12     "fiscalStart": "2020-01-01T00:00:00",
13     "fiscalEnd": "2022-03-31T00:00:00",
14     "duration": 27,
15     "manager": "C. Layton",
16     "..."
17   }
18 ]
```

References

- The OAuth 2.0 Authorization Framework > Resource Owner Password
<https://datatracker.ietf.org/doc/html/rfc6749 - section-4.3>