



# Token Based Authentication

ASP.NET Zero Angular UI consumes the [host](#) via token based authentication. Any application can authenticate and use any functionality in the application as API. For instance, you can create a mobile application consumes the same API. In this section, we'll demonstrate usage of the API from [Postman](#) (a Google Chrome extension).

## Authentication

We suggest you to disable two factor authentication for the user which will be used for remote authentication. Otherwise, two factor authentication flow should be implemented by the client. You can check account module source code to understand the flow. We assume that you have disabled two factor authentication for the **admin** user of **default** tenant since we will use it in this sample.

Following headers should be configured for all requests ( [Abp.TenantId](#) is Id of the default tenant. This is not required for single tenant applications or if you want to work with host users):

Authorization	Headers (2)	Body ●	Pre-request Script	Tests
<input checked="" type="checkbox"/>	Content-Type	application/json		
<input checked="" type="checkbox"/>	Abp.TenantId	1		

Then we can send username and password as a **POST** request to <https://localhost:44301/api/TokenAuth/Authenticate>

# ASP.NET CORE ANGULAR

## DOCUMENTS

http://localhost:22742

No Environment

POST

http://localhost:22742/api/TokenAuth/Authenticate

Params

Send

Save

Authorization

Headers (2)

Body

Pre-request Script

Tests

Code

☐ form-data

☐ x-www-form-urlencoded

☒ raw

☐ binary

JSON (application/json)

```
1 {
2   "userNameOrEmailAddress": "admin",
3   "password": "123qwe"
4 }
```

Body

Cookies

Headers (6)

Tests

Status: 200 OK

Time: 169 ms

Pretty

Raw

Preview

JSON

```
1 {
2   "result": {
3     "accessToken": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9
      .eyJodHRwOi8vc2NoZW1hcy54bWxzbnZlLnM9Zy93cy8yMDA1LzA1L2lkZW50aXR5L2NsYW1tcy9uYW1lbnR1bnRpdZm
      1lcilI6IjIiLCJodHRwOi8vc2NoZW1hcy54bWxzbnZlLnM9Zy93cy8yMDA1LzA1L2lkZW50aXR5L2NsYW1tcy9uYW1lbnR1bnRpdZm
      joiYWRtaW4iLCJodHRwOi8vc2NoZW1hcy5taWw3b3NvZnQuY29tL2FjY2Vzc2NvbnRyb2xzZXJ2aW50aWwMTAvMDcv
      Y2xhaW1zL2lkZW50aXR5cHJvdmlkZXIiOiJBU1AuTkVUIE1kZW50aXR5IiwiaXNwTmV0Lk1kZW50aXR5L1N1Y3VyaXR
      5U3RhbXAiOiJjY2I4MTNiY01YmZkLWFiODgtNmQ2Yy0zOWRlZjBjMmVhMDMiLCJodHRwOi8vc2NoZW1hcy5taWw3b3NvZnQuY29tL2FjY2Vzc2NvbnRyb2xzZXJ2aW50aWwMTAvMDcv
      NvZnQuY29tL2FjY2Vzc2NvbnRyb2xzZXJ2aW50aWwMTAvMDcvY2xhaW1zL2lkZW50aXR5cHJvdmlkZXIiOiJBU1AuTkVUIE1kZW50aXR5IiwiaXNwTmV0Lk1kZW50aXR5L1N1Y3VyaXR
      m9pbGVycGxhdGUuY29tL2lkZW50aXR5L2NsYW1tcy90ZW50aWw3b3NvZnQuY29tL2FjY2Vzc2NvbnRyb2xzZXJ2aW50aWwMTAvMDcvY2xhaW1zL2lkZW50aXR5cHJvdmlkZXIiOiJBU1AuTkVUIE1kZW50aXR5IiwiaXNwTmV0Lk1kZW50aXR5L1N1Y3VyaXR
      ZjgtNmY4MS00ZmIyLTkxOGMtMDUxMmQyNGJlYzIiIiwiaWF0IjoxNDgzNjE5NjEzLCJyYmYiOiJlODM2MTk2MTMsImV
      4cCI6MTQ4MzcwNjAxMywianXNzIjoiQWw3b3NvZnQuY29tL2FjY2Vzc2NvbnRyb2xzZXJ2aW50aWwMTAvMDcvY2xhaW1zL2lkZW50aXR5cHJvdmlkZXIiOiJBU1AuTkVUIE1kZW50aXR5IiwiaXNwTmV0Lk1kZW50aXR5L1N1Y3VyaXR
      .qOXVZwSrpnx286hniIeyE9EAmTRjwUwFNmV7K8-MpNw",
4     "expireInSeconds": 86400,
5     "shouldResetPassword": false,
6     "passwordResetCode": null,
7     "userId": 0,
8     "requiresTwoFactorVerification": false,
9     "twoFactorAuthProviders": null,
10    "twoFactorRememberClientToken": null
11  },
12  "targetUrl": null,
13  "success": true,
14  "error": null,
15  "unAuthorizedRequest": false,
16  "__abp": true
17 }
```

Feedback



# Using API

After authenticate and get the access token, we can use it to call any **authorized** actions. All **services** are available to be used remotely. For example, we can use the **User service** to get a **list of users**:



# ASP.NET CORE ANGULAR

## DOCUMENTS

http://localhost:22742

No Environment

GET

http://localhost:22742/api/services/app/User/GetUsers

Params

Send

Save

Authorization

Headers (2)

Body

Pre-request Script

Tests

Code

☒ Abp.TenantId 1

☒ Authorization Bearer eyJhbGciOiJIUzI1NiI:

key value

Bulk Edit Presets

Body

Cookies

Headers (6)

Tests

Status: 200 OK Time: 60 ms

Pretty

Raw

Preview

JSON

```
1 {
2   "result": {
3     "totalCount": 3,
4     "items": [
5       {
6         "name": "admin",
7         "surname": "admin",
8         "userName": "admin",
9         "emailAddress": "admin@defaulttenant.com",
10        "phoneNumber": "4445522",
11        "profilePictureId": "ffef8dd6-23f4-421b-b9cb-23e9e82a1639",
12        "isEmailConfirmed": true,
13        "roles": [
14          {
15            "roleId": 2,
16            "roleName": "Admin"
17          }
18        ],
19        "lastLoginTime": "2017-01-05T15:33:33.65",
20        "isActive": true,
21        "creationTime": "2016-12-08T16:18:46.787",
22        "id": 2
23      },
24      {
25        "name": "Douglas",
26        "surname": "Adams",
27        "userName": "adams",
28        "emailAddress": "adams@douglas.com",
29        "phoneNumber": "123123123",
30        "profilePictureId": null,
31        "isEmailConfirmed": true,
32        "roles": [
33          {
34            "roleId": 3,
35            "roleName": "User"
36          }
37        ],
38        "lastLoginTime": "2016-12-22T14:11:06.45",
39        "isActive": true,
40        "creationTime": "2016-12-22T14:11:05.87",
41        "id": 11
42      },
43      {
44        "name": "John"
```



# ASP.NET CORE ANGULAR

DOCUMENTS



## Next

- [Sign Up](#)

