**COMPUTING SUBJECT:** JWT Token

**TYPE:** Assignment

**IDENTIFICATION:** JWT Token Introduction

**COPYRIGHT:** *Michael Claudius*

**LEVEL:** Easy

**TIME CONSUMPTION:** 1-2 hours

**EXTENT:** 1-2 pages answering the questions

**OBJECTIVE:** Basic principles of a token based server

**PRECONDITIONS:** Rest service theory and practice. CORS.

Http concepts

Telerik Fiddler / Postman installed

**COMMANDS: Claim, Signature etc etc.**

**IDENTIFICATION:** JWT RestTokenService/MICL

Purpose

The purpose of this assignment is to be able to provide and consume restful ASP.Net Core web services using JWT.

Mission

You are to utilize JSON Web Token (JWT) in restful web services based on the ASP.Net Core services by setting up a server (provider), test the services by use of Fiddler/Postman and create a client (consumer) using the services provided.

The service supports the classic POST & GET requests. This we shall do in 8 steps:

1. Create a project a token service
2. Test the services using Browser
3. Get all users
4. Authenticate as a user
5. Get all users as authorized user
6. Use the service on another server than localhost
7. Understanding the program and the token generation
8. Publish in Azure if possible

This assignment holds all steps and the first 6 steps is written as a tutorial done by you.

Very imprtamt is to us time to understand the code in step 7. This is done in discussion groups.

*Later in another assignment, JWT RestTokenClient, you will create a console client project.*

Domain description

The service provides two simple operations with routes for:

* /users for a list of users

Accepts GET request if the valid token is given in the Authorization header

Returns all users

* /users/authenticate for login

Accepts POST requests containing username and password in the body.

Returns user details with a JWT token if the user is accepted otherwise error code.

When surfing on the net it is easy to find many descriptions more or less useful, and in more or less updated versions. Here are some of the better:

*Useful links for JWT:*

JSON Web Token structure

<https://en.wikipedia.org/wiki/JSON_Web_Token>

JSON Web Token Introduction

<https://medium.com/ag-grid/a-plain-english-introduction-to-json-web-tokens-jwt-what-it-is-and-what-it-isnt-8076ca679843>

JWT project in Visual Studio

<https://www.c-sharpcorner.com/article/jwt-json-web-token-authentication-in-asp-net-core/>

JWT example with GUI and cookies

<https://code.msdn.microsoft.com/How-to-authentication-web-f58efc25>

Assignment 1: Restful ASP.Net Framework-service provider

You are to make a ASP.NET Core Web Rest Service provider JWTRestService.

Download and install the project JWTRestService from your teacher´s homepage.

Open the project in Visual Studio and get an overview of the program and the various folders.

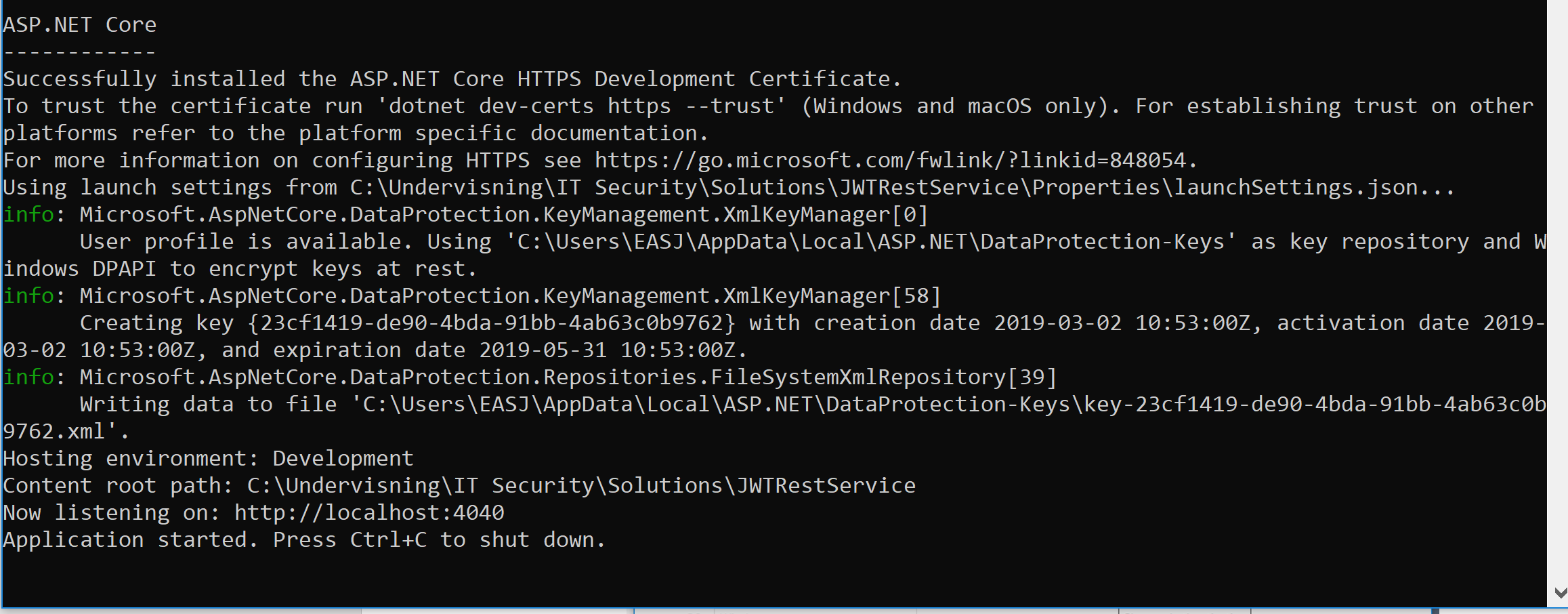
*Notice you cannot just run it from Visual Studio!*

Assignment 2: Execute the web application

Open a command prompt and change directory to the one holding the Webapi.csproj file

Execute the web application by the command : dotnet run.

And you will something like this:



As you can see the application is using http and running on port 4040 on my computer (it will be the same on yours).

Open a browser and give the url:

<http://localhost:4040/users>

Not very successful right ☺ ! Why ?

Check out error Http 401 !

Read on.

Assignment 3: Get all users from Fiddler/Postman

Now we will try to invoke the methods from Fiddler/Postman using the same url as before.

Start Fiddler.

Be aware that you must:

1. Click on Composer,
2. Choose GET
3. Copy and paste the url from Browser (<http://localhost:4040/users>) into the text field
4. Click Execute

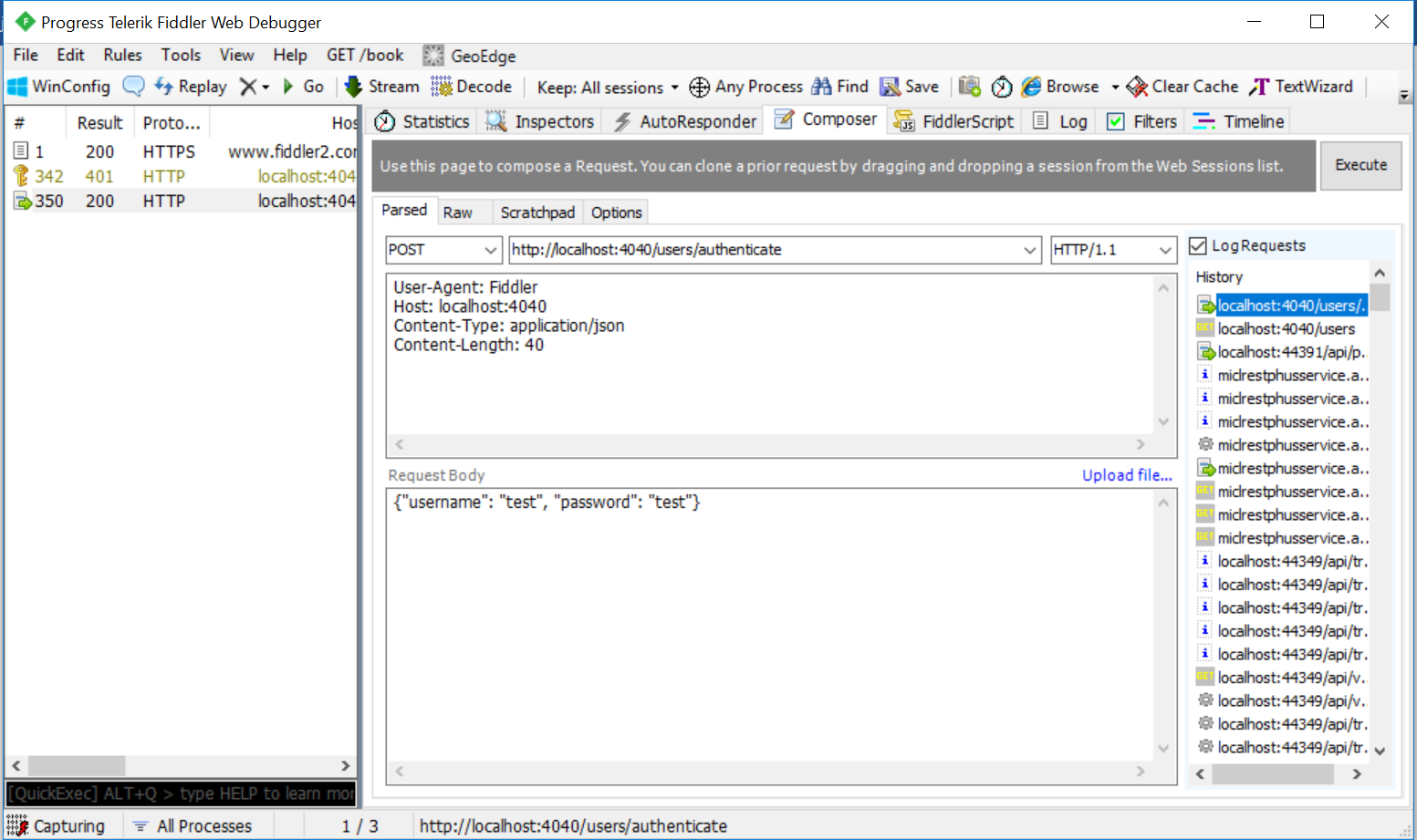
Of course same error, but now we know what to do.

Assignment 4: Authenticate the user from Fiddler/Postman

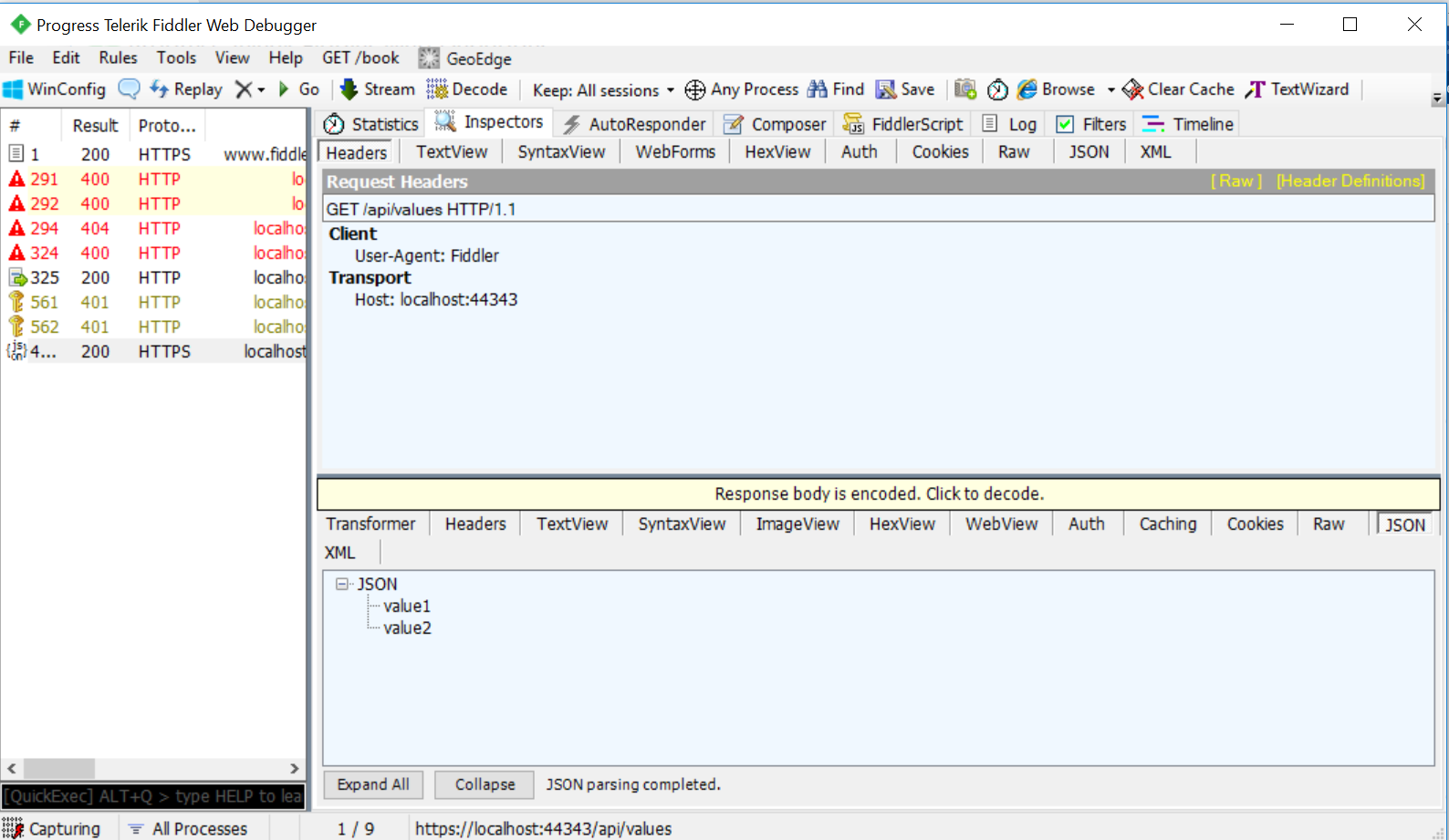
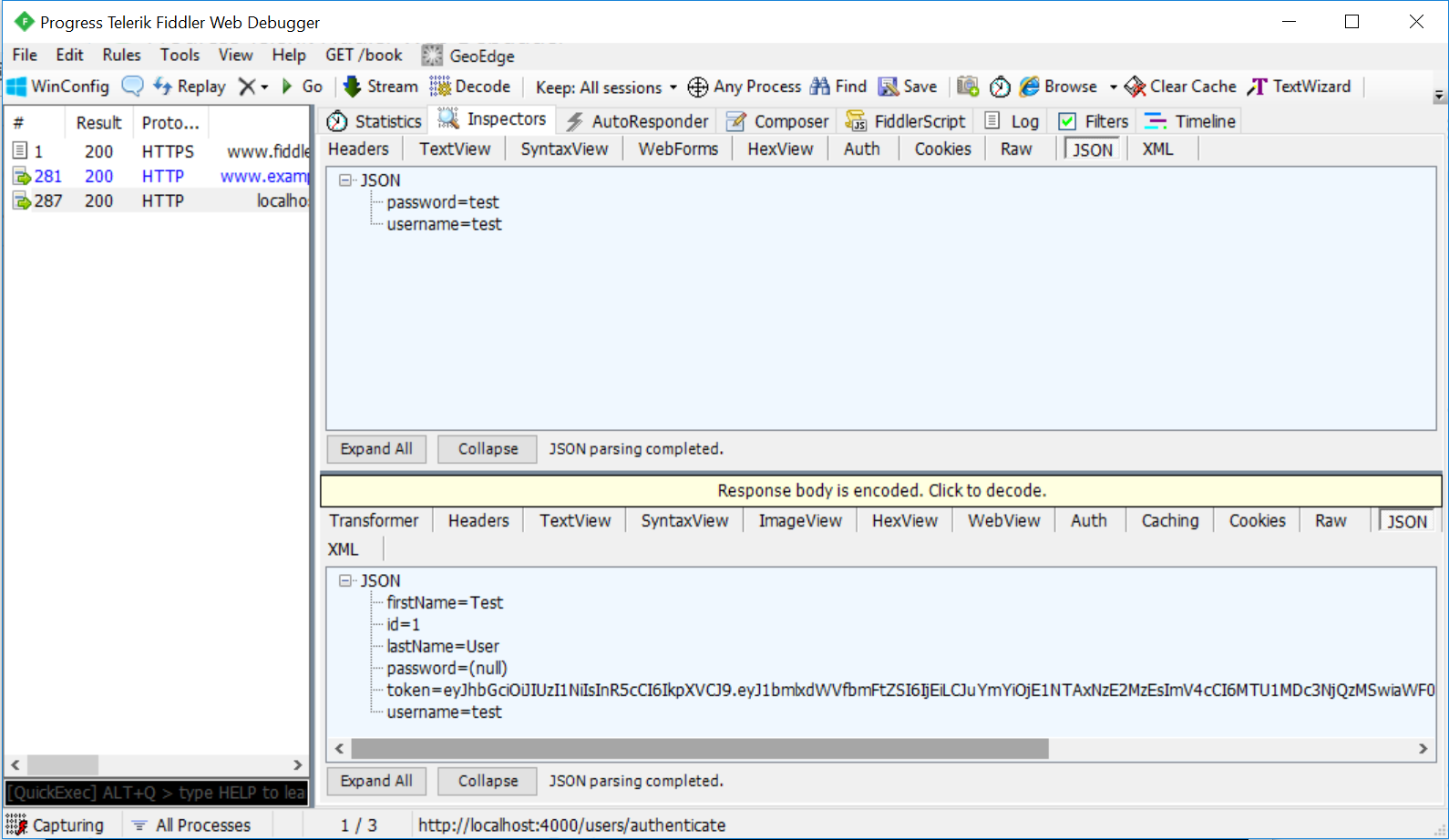
Use POST and the api “users/authenticate” to authenticate a user with correct username and password.

Look in Userservices, UsersController and User classes to find and see the properties and the information.

Then Execute a POST request.



Hopefully you will get something like this:



Notice the “Token=”

Finally click on Headers and in Raw you can see the HTTP response.

HTTP/1.1 200 OK

Date: Thu, 14 Feb 2019 19:13:50 GMT

Content-Type: application/json; charset=utf-8

Server: Kestrel

Transfer-Encoding: chunked

109

{"id":1,"firstName":"Test","lastName":"User","username":"test","password":null,"token":"eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1bmlxdWVfbmFtZSI6IjEiLCJuYmYiOjE1NTAxNzE2MzEsImV4cCI6MTU1MDc3NjQzMSwiaWF0IjoxNTUwMTcxNjMxfQ.-TQRMeH\_K15ZWgrPqlfasLETBGlBuBTnR980CdqRuF4"}

0

Copy the token to be used later.

Assignment 5: Get all users as authorized user

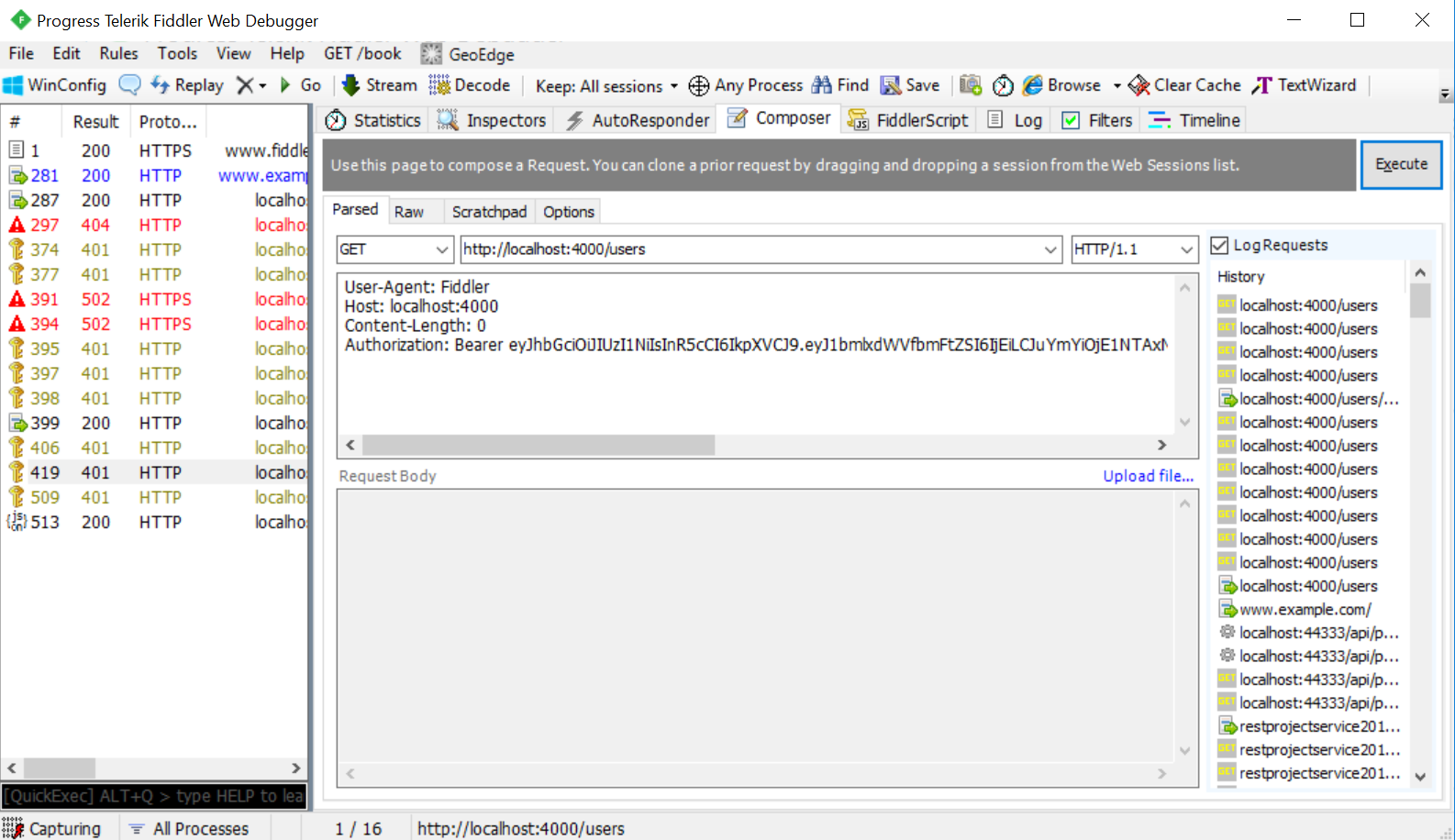
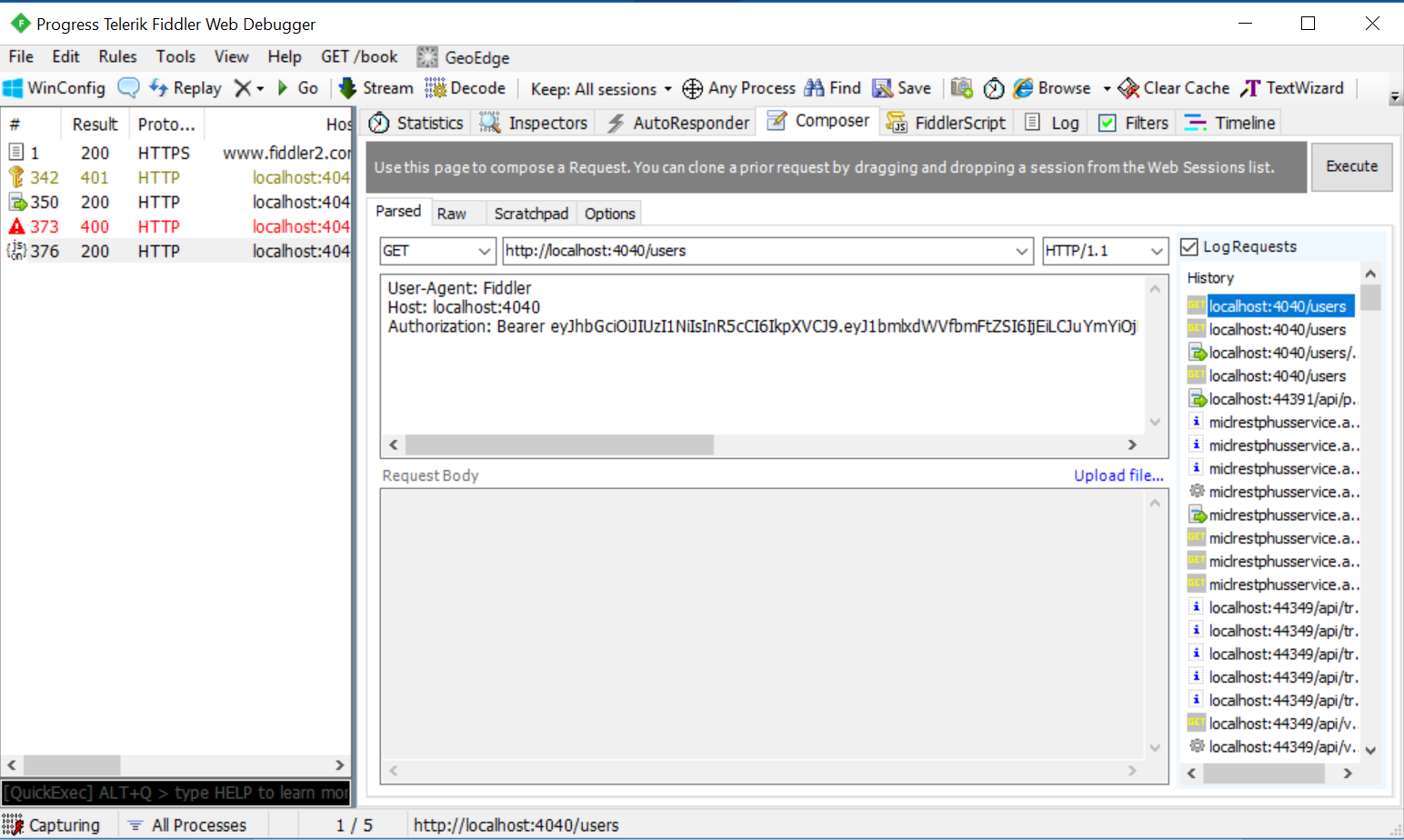
Run GET from Fiddler again like in assignment 3, BUT remember to send over the token.

Can you find out yourself how to do that ?

Otherwise look at next page.

In the header you need to add the Authorization field with a Bearer <token>.

Looks like this:



And the list of users is on the screen

Assignment 6 Two different hosts one consumer one provider

From your client try to consume the service on the other server.

Exchange tokens with another student. Can you use his token ?

*Tip: Remember in the server program to use the ip-address of the server; i.e. not localhost.*

Assignment 7 Understanding JWT generating and the program

In a group of 3-5 students discuss the following questions and write down the answers:

1. What are the main purpose of each class in the program
2. Program.cs: Explain especially
3. AddAuthenticateScheme
4. AddJwtBearer
5. What is Bearer
6. What are the properties of User ?
7. UsersController: What is the purpose of [Authorize] [AllowAnonymous]
8. User: What are the properties of User ?
9. Userservice: Explain
   1. Tokenhandler
   2. SecurityTokenDescriptor
   3. The key
   4. Claim
   5. SymmetricSecurityKey
   6. List the token content
   7. Use of HMac and SHA256

And finally an interesting issue:

1. Where is the token saved on the server ?

Assignment 8 Publish in Azure

Is it possible to publish the service to Azure and use it?

If not why ?