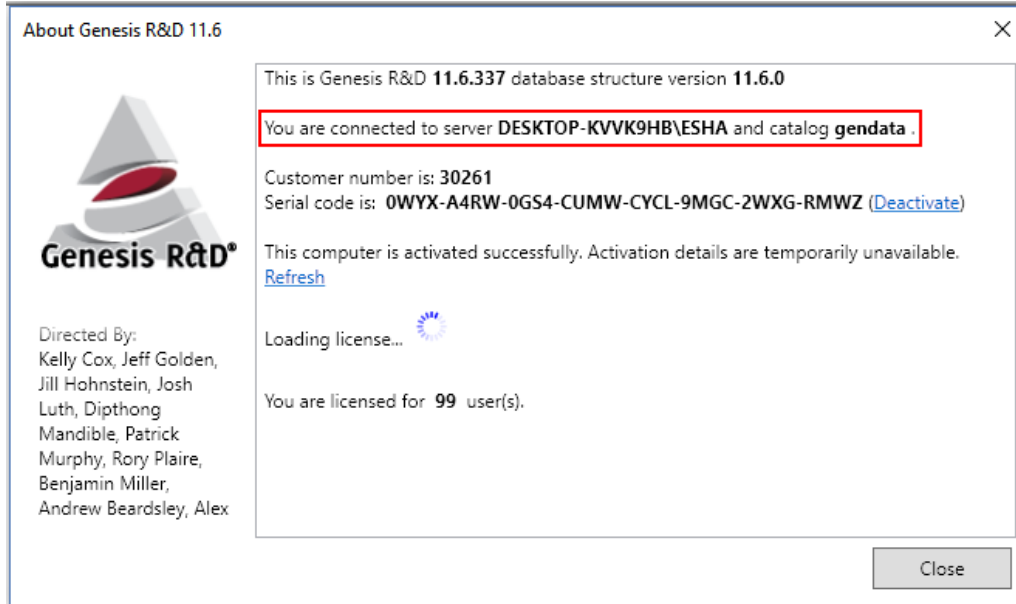


API Installation Procedures

Section 0 – Install Genesis R&D

NOTE: Verify if Genesis has already been installed. If so, verify that the installation is NOT using LocalDB. You can determine what type of database connection any given installation of Genesis is using by launching Genesis and clicking the “About” button from the “Home” tab in the ribbon. You will see a dialog like this:

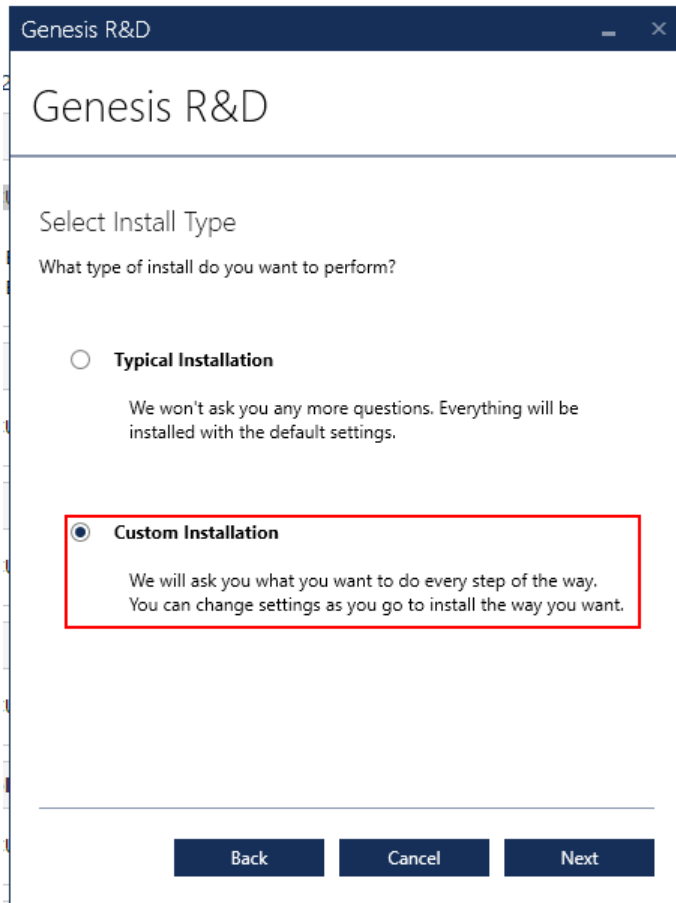


The “You are connected to server...” line would show “(localdb)\v11.0” if the installation was using a LocalDB connection.

If Genesis is configured to use a LocalDB connection, the process for switching from one install-type to another may depend on how many users are configured for the current install and other factors. A call to our Support staff would be the best approach to take for switching the Genesis installation over to a SQL Server instance.

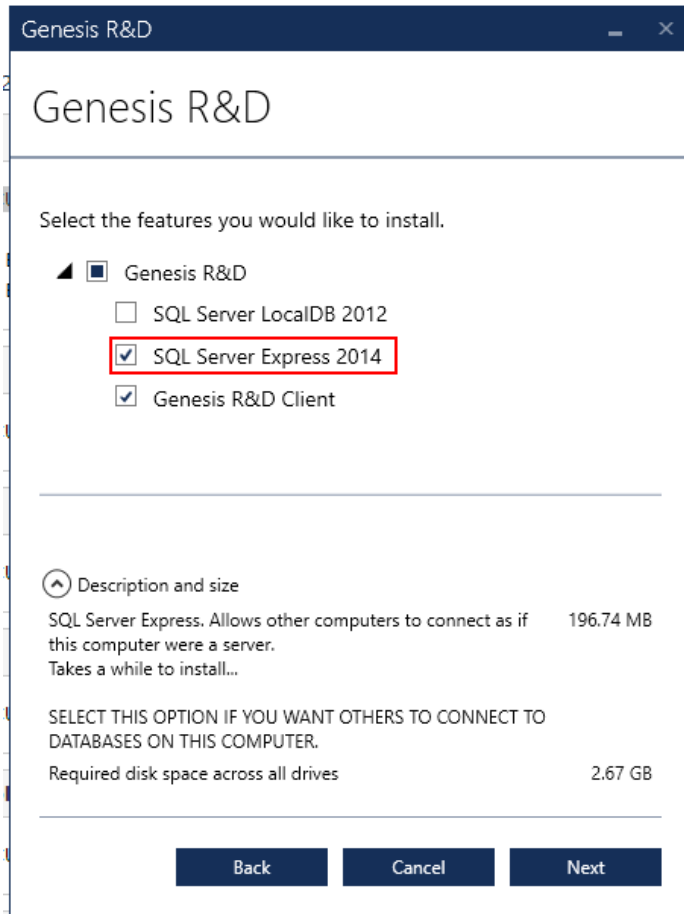
If Genesis has not been installed, proceed with the next steps. If Genesis has been installed and is using a SQL Server instance, proceed to **Section I**.

1. Launch the Genesis R&D installer.
2. On the “Select Install Type” page of the installer, select “Custom Installation”:



3. Click Next.

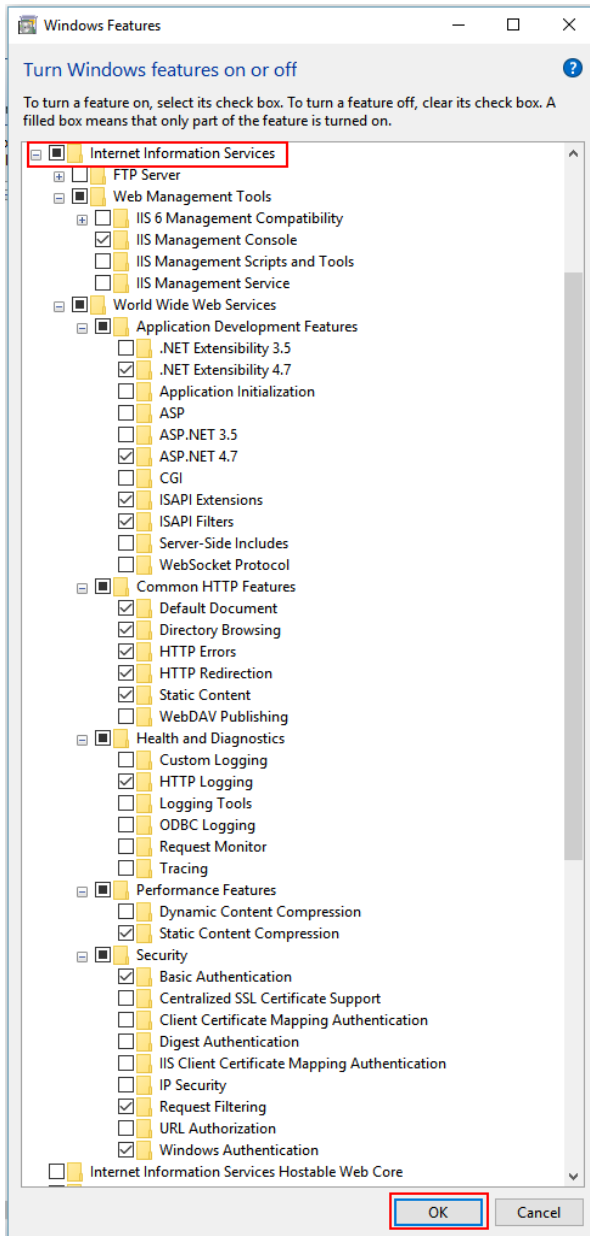
4. On the “Select the features you would like to install” page, uncheck “SQL Server LocalDB 2012” and check “SQL Server Express 2014”:



5. Complete the Wizard and finish the installation.

Section I - Enable IIS on Windows

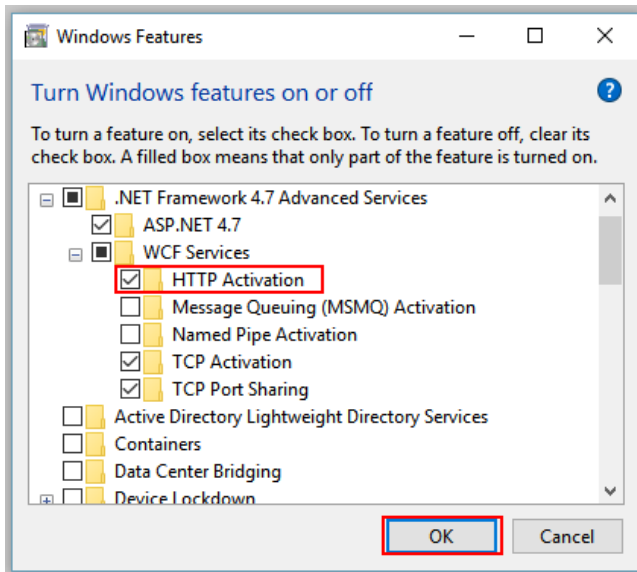
1. Click the Start button.
2. Type "Windows Features".
3. Select "Turn Windows Features on or off" from the search results.
4. Expand the "Internet Information Services Node".
5. Ensure that the features select match the screen shot:



6. Click OK.

Section II – Enable WCF on Windows

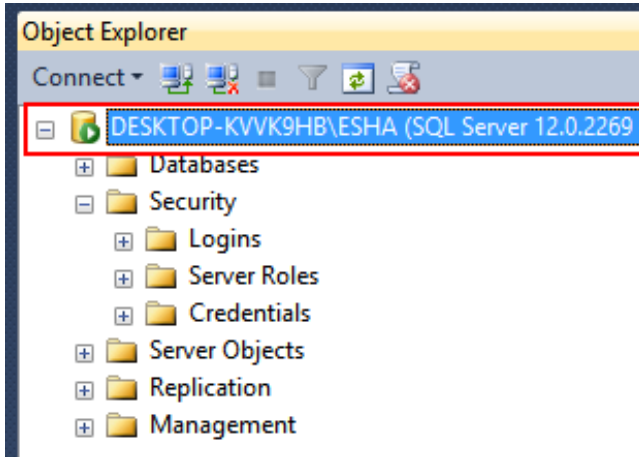
1. Return to the “Windows Features” dialog.
2. Expand the “.NET Framework <version> Advanced Services” node.
(NOTE: If you have a more recent version of .NET installed, the “.NET Framework <version>...” will reflect that version number instead.)
3. Expand the “WCF Services” node.
4. Select “HTTP Activation”:



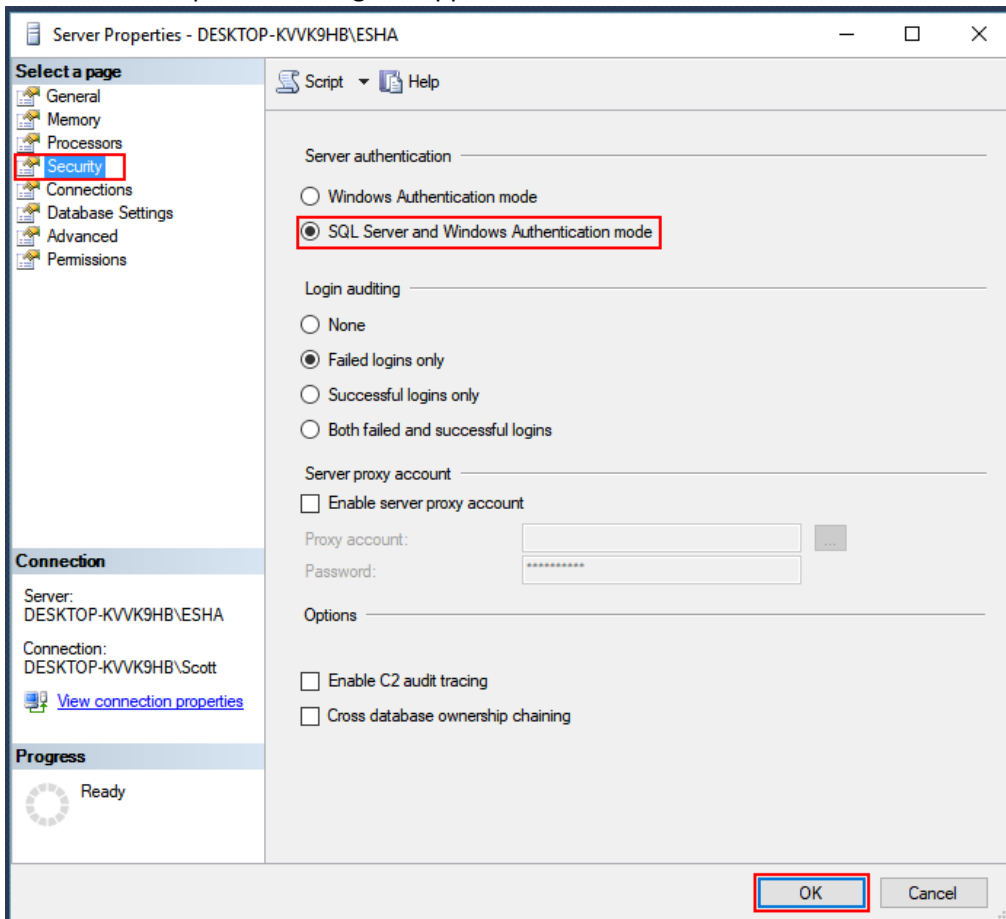
5. Click OK.

Section III – Enable SQL Authentication

1. Launch SQL Server Management Studio (SSMS).
2. Connect to the database server that hosts your GENDATA database.
3. Right-click on the server name:

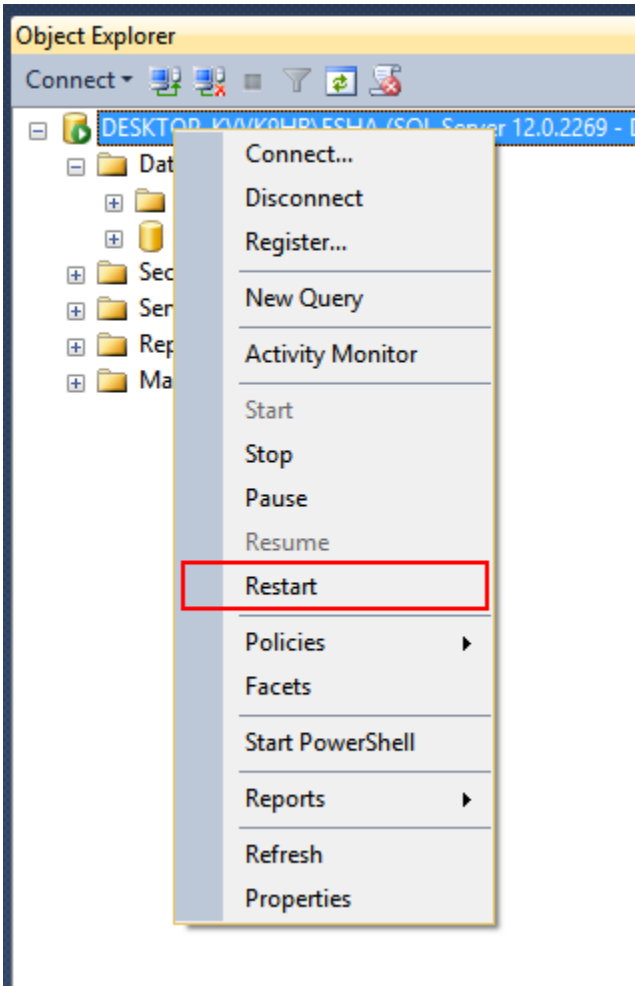


4. Select “Properties”.
5. The “Server Properties” dialog will appear:



6. Select the “Security” node.
7. Select the “SQL Server and Windows Authentication mode” radio button.

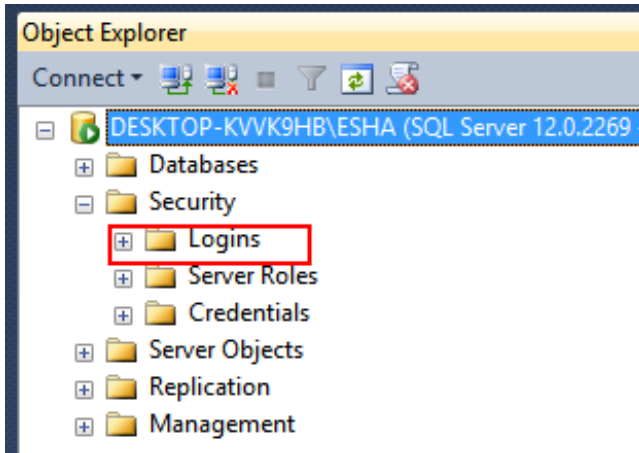
- Click OK.
- Right-click on the sever name again:



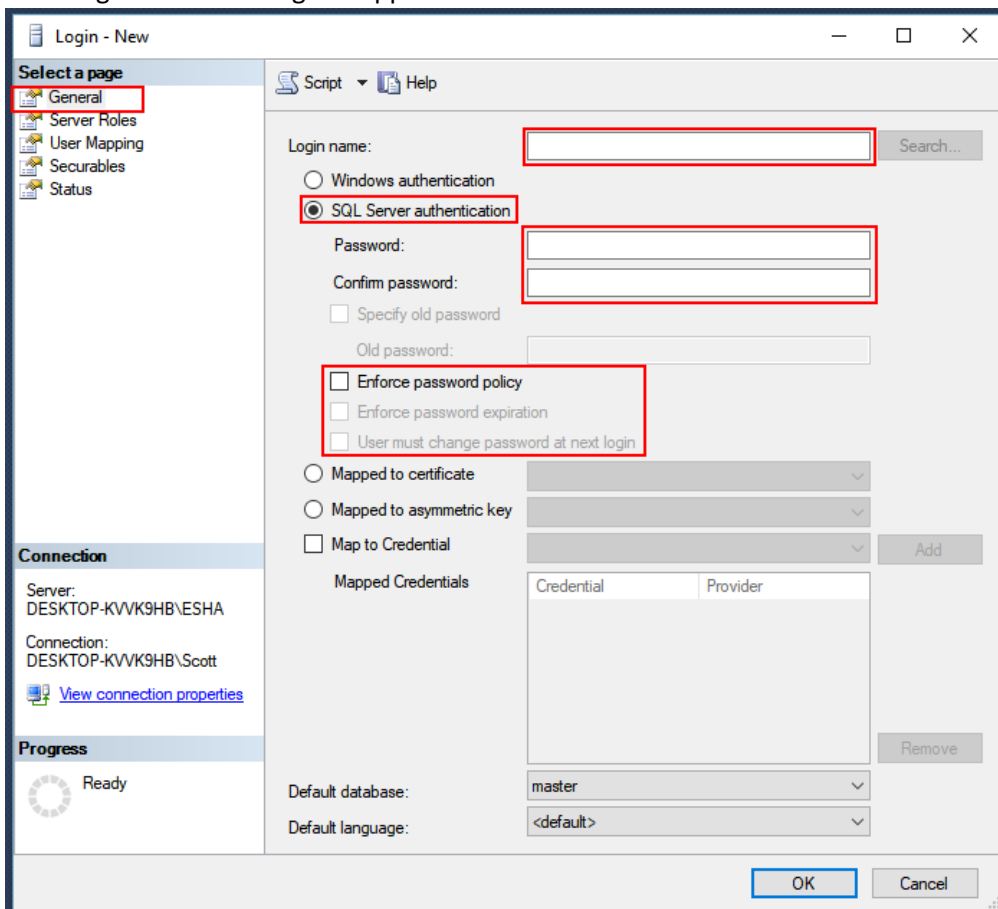
- Select "Restart".
- Answer "Yes" when asked to confirm that you want to restart the server.

Section IV – Create an API User login in GENDATA

1. Launch SSMS.
2. Connect to the database server that hosts your GENDATA database.
3. Expand the “Security” node.
4. Right-click the “Logins” node:

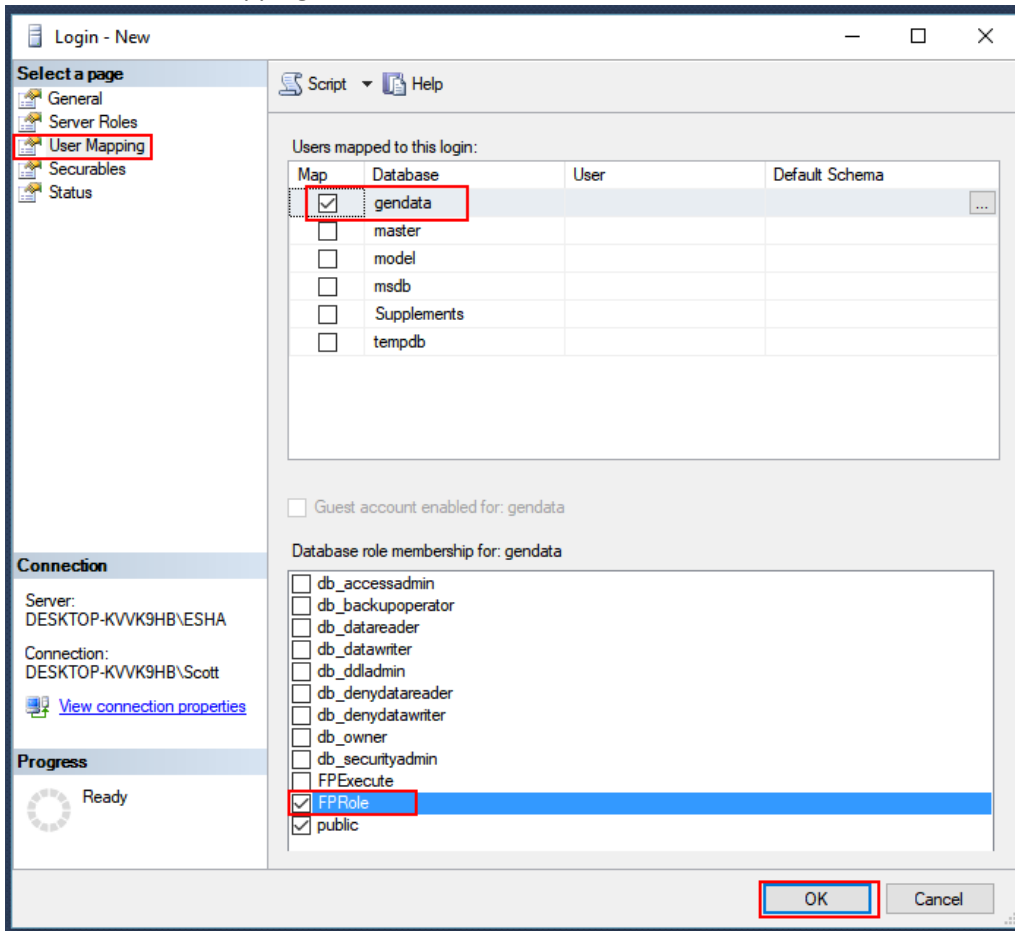


5. Select “New Login”.
6. The “Login – New” dialog will appear:



7. Enter “APIUser” or some other login name for the “Login Name” field.

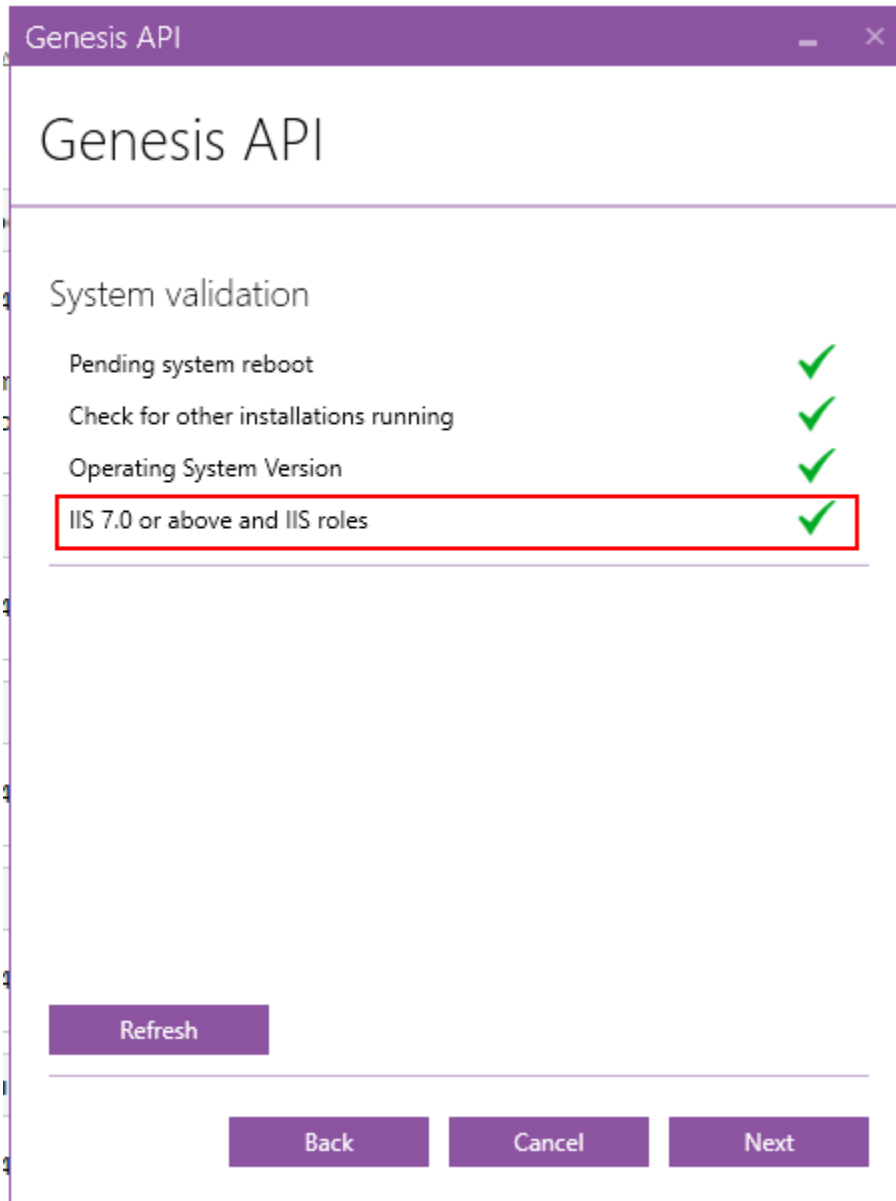
8. Select the “SQL Server Authentication” radio button.
9. Create a password for the new login and make a note of it for later reference.
10. Deselect “Enforce password policy” checkbox.
11. Select the “User Mapping” node:



12. Select the “gendata” database from the “Users mapped to this login” list.
13. Select “FPRole” from the “Database role membership for: gendata” list.
14. Click OK.

Section V – Install the API

1. Install the API via the installer by right-clicking on the installer file and selecting “Run as Administrator”.
2. On the “System Validation” page of the installer, it will verify that all prerequisites have been completed. It will typically look as follows:

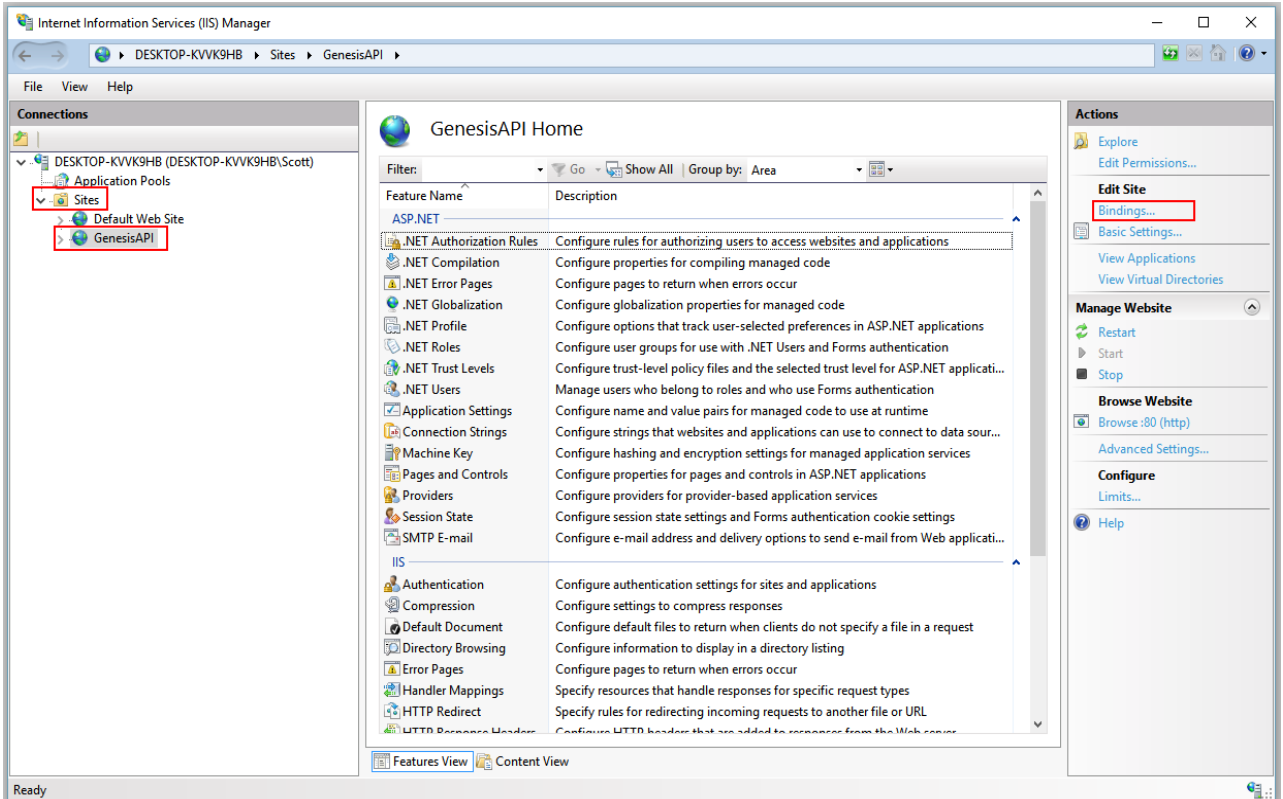


3. If the “IIS 7.0 or above and IIS roles” has a red “X”, click the “Activate required Features and Roles” button that will appear.
4. Click “Next”.
5. On the “SQL Server Connection” page, select the server you connected to in the above steps in the “Server name” drop down. If it’s not in the list, manually enter it.
6. Select the “Use SQL Authentication” radio button.
7. Enter “APIUser” or the name you created in **Section IV – Step 7** in the “User name” field.

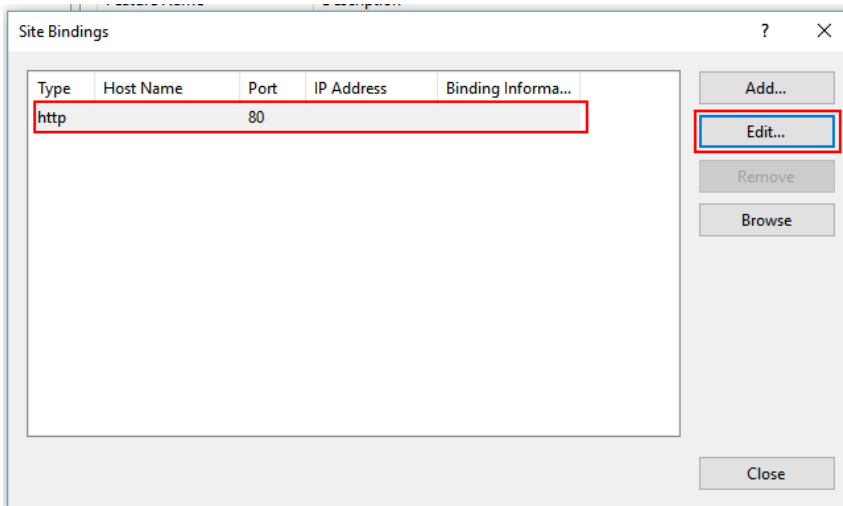
8. Enter the password you created in **Section IV – Step 9** in the “Password” field.
9. Select “gendata” from the “Select or enter a database” drop down.
10. Click the “Test Connection” button. Assuming it succeeds, click the Install button and allow the installation to complete.

Section VI – Final API Configuration

1. Click the “Start” button.
2. Type “IIS”.
3. Select “Internet Information Services (IIS) Manager” from the search results.
4. The “Internet Information Services (IIS) Manager” window will appear:



5. Expand the “Sites” node.
6. Select the “GenesisAPI” node.
7. Click the “Bindings...” link at the far right.
8. The “Site Bindings” dialog will appear:



9. Select the “http” row in the list.
10. Click the “Edit...” button.
11. The “Edit Site Binding” dialog will appear:

The screenshot shows the "Edit Site Binding" dialog box. The "Type" dropdown is set to "http". The "IP address" dropdown is set to "All Unassigned". The "Port" text box contains the value "8080". Below the "Port" field, there is a "Host name" text box which is empty. Below the "Host name" text box, there is an example: "Example: www.contoso.com or marketing.contoso.com". At the bottom right of the dialog, there are two buttons: "OK" and "Cancel". The "OK" button is highlighted with a red box.

12. Change the “Port” value to any available port (e.g. 8080), ensuring it does not conflict with any other sites in IIS, and make a note of that number.
13. Click OK.
14. Click Close from the “Site Bindings” dialog.
15. Click the “Restart” link under the “Manage Website” section at the far left of the “Internet Information Services (IIS) Manager” window.
16. Close the “Internet Information Services (IIS) Manager” window.

Section VII – Verify API Installation

1. Open a browser and navigate to <http://localhost:<portnumber>/ApiInfo.svc>, where <portnumber> is the number you changed the GenesisApi site to in **Section VI – Step 11**. You should receive the “ApiInfo Service” page:



ApiInfo Service

You have created a service.

To test this service, you will need to create a client and use it to call the service. You can do this using the sv

```
svcutil.exe http://localhost/ApiInfo.svc?wsdl
```

You can also access the service description as a single file:

```
http://localhost/ApiInfo.svc?singleWsdl
```

This will generate a configuration file and a code file that contains the client class. Add the two files to your c

C#

```
class Test
{
    static void Main()
    {
        ApiInfoClient client = new ApiInfoClient();

        // Use the 'client' variable to call operations on the service.

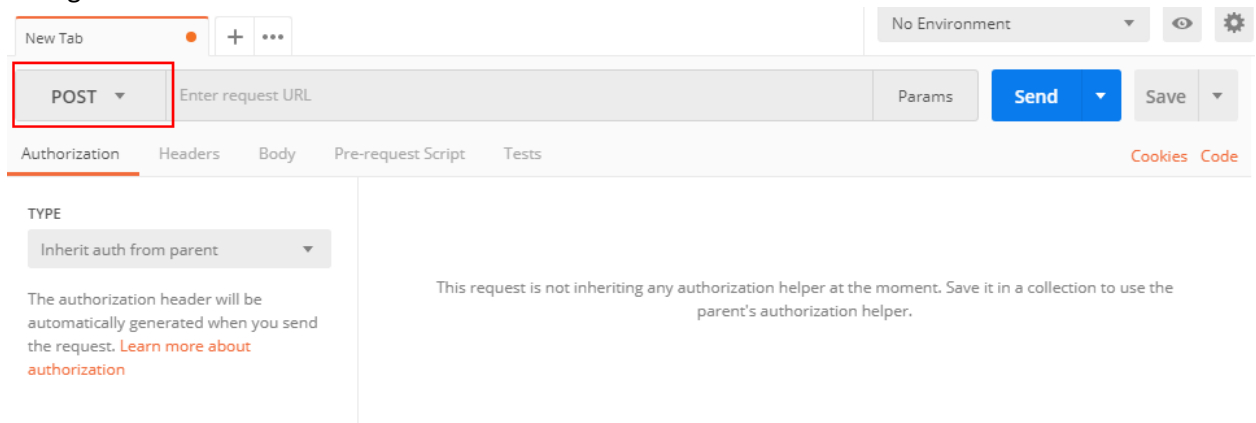
        // Always close the client.
        client.Close();
    }
}
```

Visual Basic

```
Class Test
    Shared Sub Main()
        Dim client As ApiInfoClient = New ApiInfoClient()
        ' Use the 'client' variable to call operations on the service.

        ' Always close the client.
        client.Close()
    End Sub
End Class
```

2. Download and install Postman from <https://www.getpostman.com/apps>.
3. Launch Postman. Use a dummy account to login or use one of your own making.
4. Change “GET” to “POST”:



New Tab + ... No Environment

POST Enter request URL Params Send Save

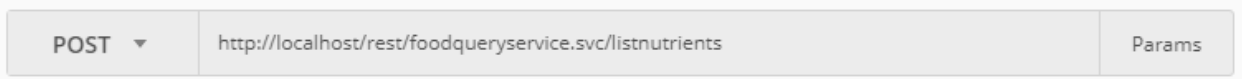
Authorization Headers Body Pre-request Script Tests Cookies Code

TYPE
Inherit auth from parent

The authorization header will be automatically generated when you send the request. [Learn more about authorization](#)

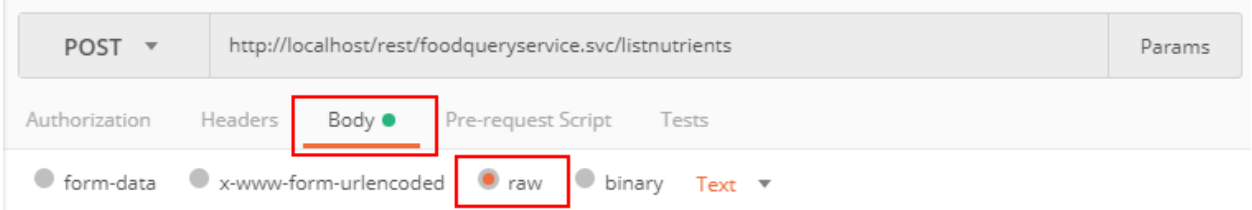
This request is not inheriting any authorization helper at the moment. Save it in a collection to use the parent's authorization helper.

5. Enter the <http://localhost:<portnumber>/rest/FoodQueryService.svc/ListNutrients> in the “Enter request URL”:



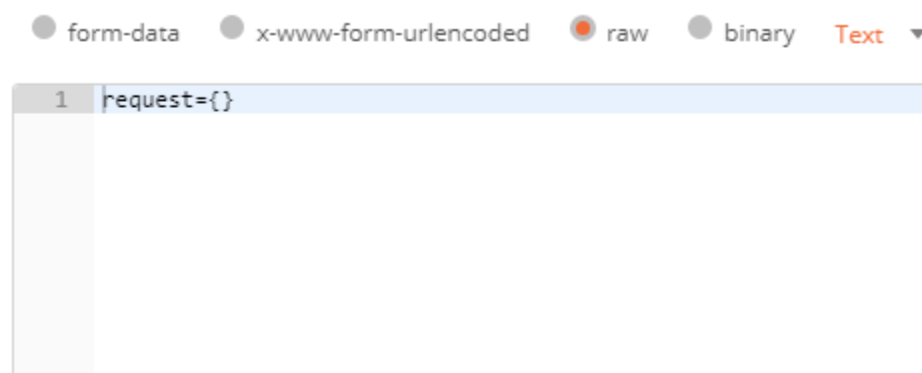
A screenshot of a REST client interface. At the top, there is a dropdown menu set to "POST" and a text input field containing the URL "http://localhost/rest/foodqueryservice.svc/listnutrients". To the right of the URL field is a "Params" button.

6. Select the “Body” tab and then click the “Raw” radio button:



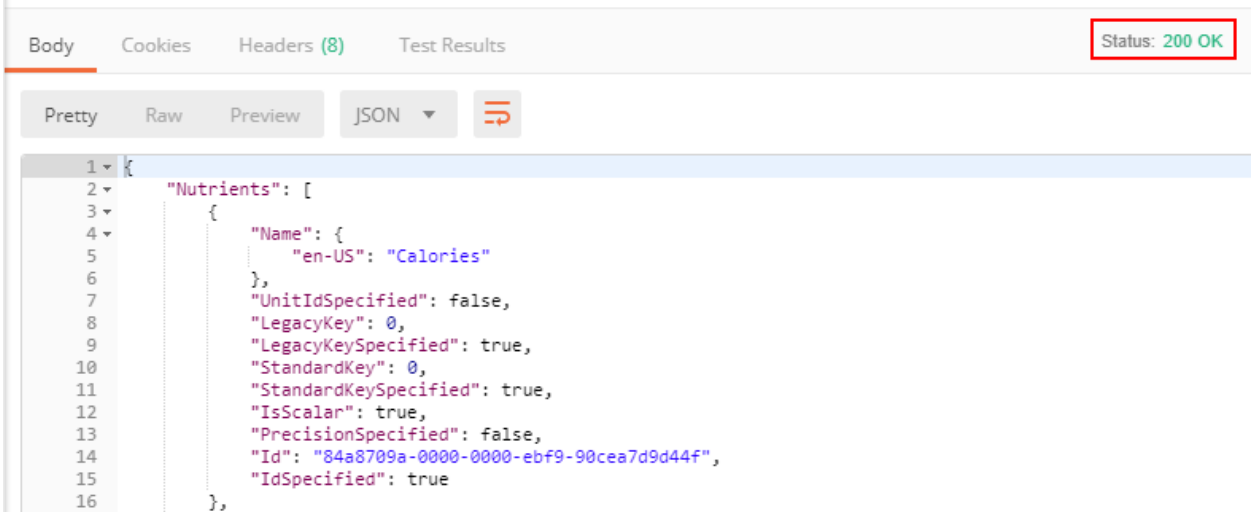
A screenshot of the REST client interface. The "Body" tab is selected and highlighted with a red box. Below the tabs, there are radio buttons for "form-data", "x-www-form-urlencoded", "raw", and "binary". The "raw" radio button is selected and highlighted with a red box. To the right of the radio buttons is a "Text" dropdown menu.

7. In the empty text box below, enter “request={}”:



A screenshot of the REST client interface. The "raw" radio button is selected. Below the radio buttons is a text input field containing the text "request={}" on line 1. The text input field is highlighted with a blue background.

8. Click the blue “Send” button.
9. You should get a response in the “Response” section below the request text box with a Status of 200:



A screenshot of the REST client interface showing the response. The "Status" is "200 OK" and is highlighted with a red box. The response body is displayed in JSON format, showing a list of nutrients. The "raw" view is selected, and the JSON is displayed in a code editor. The response is as follows:

```
1 {
2   "Nutrients": [
3     {
4       "Name": {
5         "en-US": "Calories"
6       },
7       "UnitIdSpecified": false,
8       "LegacyKey": 0,
9       "LegacyKeySpecified": true,
10      "StandardKey": 0,
11      "StandardKeySpecified": true,
12      "IsScalar": true,
13      "PrecisionSpecified": false,
14      "Id": "84a8709a-0000-0000-ebf9-90cea7d9d44f",
15      "IdSpecified": true
16    },
17  ]
18 }
```

10. If you have gotten the ApilInfo.svc page and a valid response the API has been successfully configured.