

How to integrate with Eloqua

Last Modified on 07/06/2026 2:44 pm EDT

Eloqua is a software-as-a-service (SaaS) platform for marketing automation offered by Oracle that aims to **help B2B marketers and organizations manage marketing campaigns and sales lead generation**.

Digioh's Eloqua integration allows you to push a lead as a contact to Eloqua with pop-up forms, pop-ups, surveys, and quizzes. We will integrate Eloqua Rest API with Digioh to pass the leads as contacts.

Eloqua Integration Requirements

Before setting up your integration, you will need the following parameters from Eloqua :

- **Base URL**
- **Company Name**
- **Username**
- **Password**

For Rest API basic authentication [see this](#)

Base URLs

Eloqua supports multiple data centers, each with a unique domain name. Eloqua refers to these as "pods", there are currently seven in total - p01, p02, p03, p04, p06, p07, and p08. The <https://login.eloqua.com/id> endpoint allows you to programmatically discover a given Eloqua instance's pod, associated domain name, and base URL, which is required for making API calls to the instance.

Note: We will get the Base URL through Eloqua Login API

Digioh

If you are setting up a Digioh integration by yourself, we first recommend reading about [Digioh Integration Basics](#).

Follow these instructions to integrate with Eloqua.

1. Add a New Integration.

2. Set a name in the Friendly Name field.

Add New Integration

Friendly Name:

Integration:

Form Post URL:

Post Type:

Box (optional):

3. Set API Form Post in Integrations Dropdown.

Add New Integration

Friendly Name:

Integration:

Form Post URL:

Post Type:

Box (optional):

4. Enter Form Post URL

<https://www05.secure.eloqua.com/api/REST/1.0/data/contact>

Note: <https://www05.secure.eloqua.com> The Host url will be different for your Eloqua account

Add New Integration

Friendly Name:

Integration:

Form Post URL:

Post Type:

Box (optional):

Create Integration

Cancel

5. Select JSON Raw in Post Type Dropdown

Post Type:

6. Select Campaign

Add New Integration

Friendly Name:

Integration:

Form Post URL:

Post Type:

Box (optional):

Create Integration

Cancel

7. Click Create Integration.

After clicking **Create Integration**, you will see the advanced settings page.

8. Configure the Advanced Integration Settings.

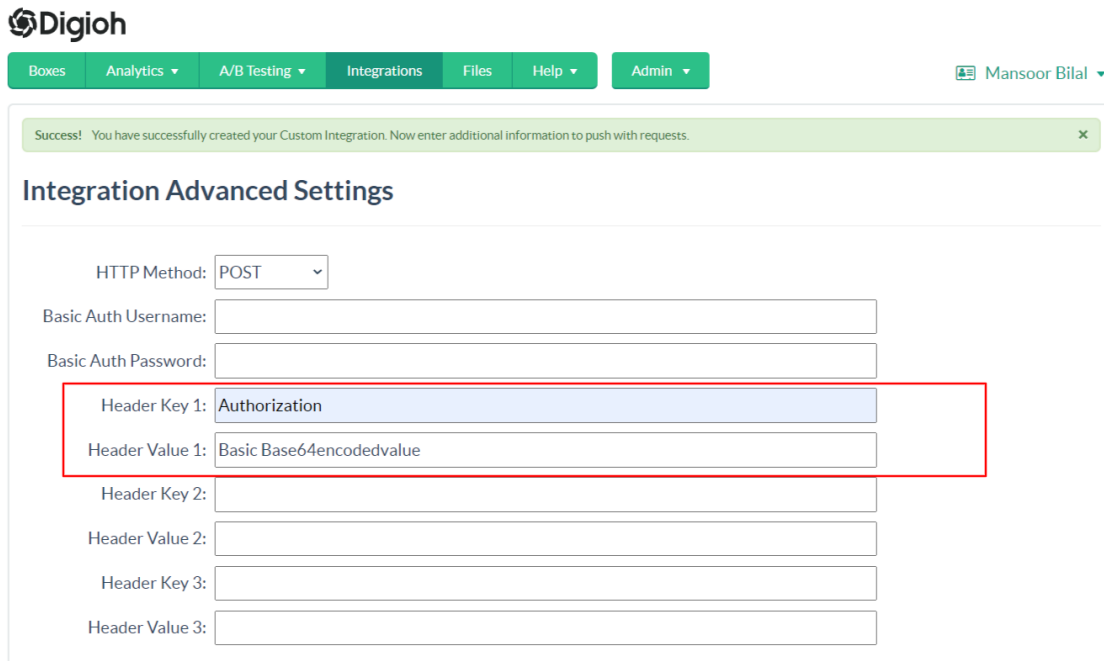
- Set the HTTP method to **POST**

Steps to create base64 encoded Authorization Header value:

- Combine your Eloqua company name with username and password in this way => `siteName + '\ ' + username + ':' + password`
Where siteName is your company name and Username and password are your Eloqua account login details.
- Go to this [website](#) And enter the combined string it will generate the base64 encoded value that will be used in the authorization header.

- **Header 1:** Enter the **Authorization** in Header 1 Key, and enter the Base64 Encoded value like this.

```
Authorization: Basic Q09NUEFOWVhcdXNlcjE6cGFzc3dvcmQxMjM=
```



The screenshot shows the Digioh user interface. At the top, there is a navigation bar with buttons for 'Boxes', 'Analytics', 'A/B Testing', 'Integrations', 'Files', 'Help', and 'Admin'. The user's name 'Mansoor Bilal' is visible in the top right. A green success message states: 'Success! You have successfully created your Custom Integration. Now enter additional information to push with requests.' Below this is the 'Integration Advanced Settings' section. It contains several input fields: 'HTTP Method' (set to POST), 'Basic Auth Username', 'Basic Auth Password', 'Header Key 1' (set to 'Authorization'), 'Header Value 1' (set to 'Basic Base64encodedvalue'), 'Header Key 2', 'Header Value 2', 'Header Key 3', and 'Header Value 3'. A red rectangular box highlights the 'Header Key 1' and 'Header Value 1' fields.

- Add the following JSON in Raw Data to Send (merge variables will be replaced) field

```
{
  "address1": "[ADDRESS]",
  "businessPhone": "[PHONE]",
  "city": "[CITY]",
  "country": "[COUNTRY]",
  "emailAddress": "[EMAIL]",
  "firstName": "[FIRST_NAME]",
  "lastName": "[LAST_NAME]",
  "postalCode": "[POSTAL_CODE]",
  "title": "Actor",
  "fieldValues": [
    {
      "id": "100017",
      "value": "Mr."
    },
    {
      "id": "100044",
      "value": "source-integration"
    }
  ]
}
```

NOTE: Update the payload according to your requirement.

That's all, Your Eloqua integration is ready to go!
