blueshift

If you run an e-commerce store using Magento, you can use our extension to integrate our platform with it. We will then get data from your store and you can use our platform to run campaigns for your users. Our extension for Magento makes it easy to integrate your store with our platform. Here's how to deploy our extension for Magento into your store:

Note: The following steps correspond to a Magento setup that uses Apache as its web server. Similar/analogous steps can be used for a setup that uses Nginx as its web server.

1. Click here to download the zipped files of the extension and unzip them to a location on your computer. For example: `/Users/TestUsr/Downloads/magento-extension`.
2. Copy the **Blueshift** folder stored under the location described in step 1. For example, in step 1, if you unzipped the files of the extension at: `/Users/TestUsr/Downloads/magento-extension`, the **Blueshift** folder resides under the `magento-extension` folder.
3. Copy the **Blueshift** folder and paste it in the `/app/code` folder of the Magento installation directory (where the Magento installation files are stored). For example, if the Magento installation files are stored at `/var/www/html/magento2`, copy the folder under: `/var/www/html/magento2/`**`app/code`**.
4. After you copy-paste the right set of files at the right location, launch the terminal and change directory to `/var/www/html/magento2`.
5. After you change directory on the terminal to the above location, run the following commands on it:

   ```
   sudo php bin/magento module:enable Blueshift_Blueshiftconnect
   --clear-static-content
   sudo php bin/magento setup:upgrade
   sudo php bin/magento setup:di:compile
   php bin/magento setup:static-content:Deploy -f
   ```

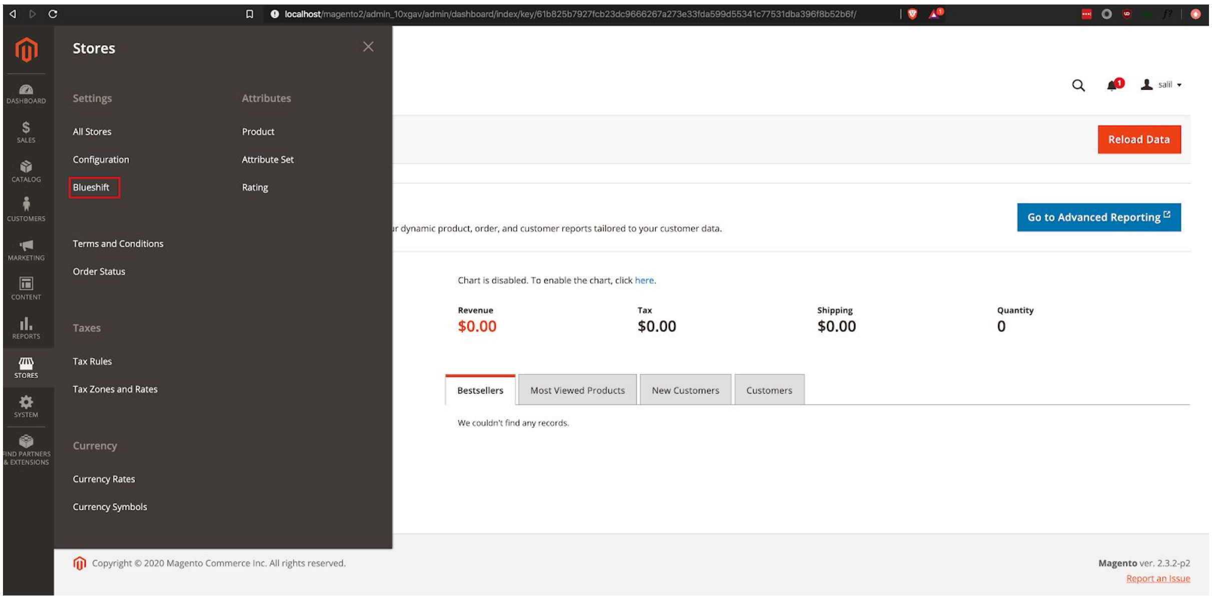
6. Run the following commands to let everyone get read and write access to the following directories:

   ```
   sudo chmod -R 777 var/ generated/
   sudo chmod -R 777 pub
   ```

7. After you run the commands above, run the following commands to flush the cache:

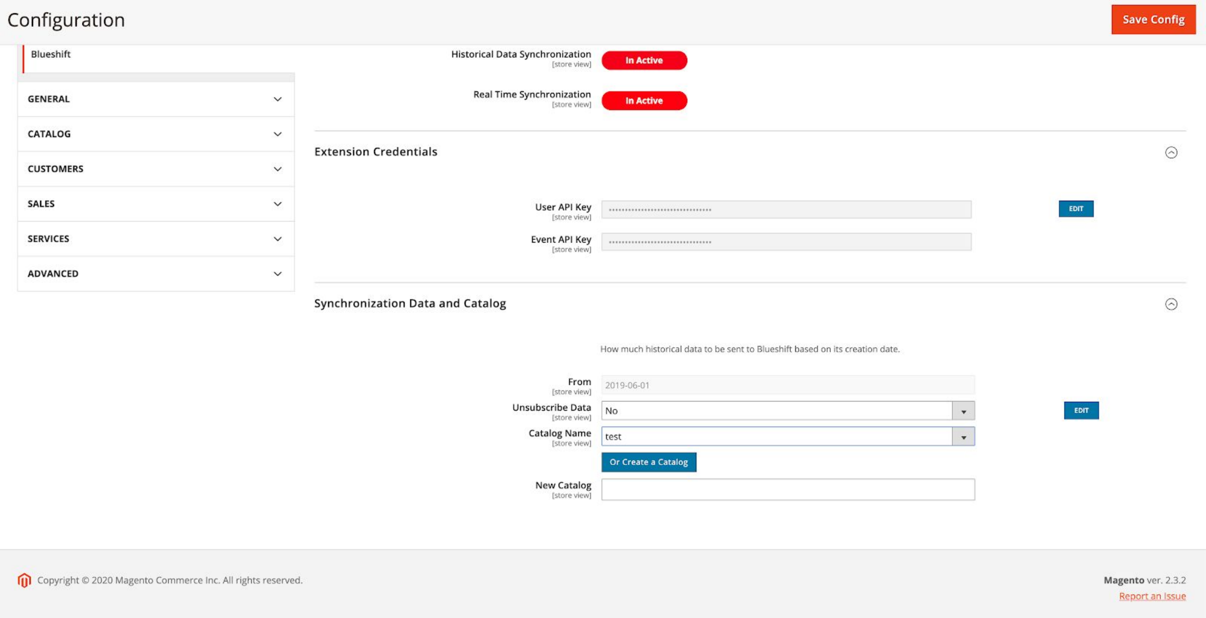
   ```
   php bin/magento cache:clean
   php bin/magento cache:flush
   ```

After you successfully run all the commands above, sign in to Magento and verify if you can see Blueshift listed under stores.



After you launch the extension, perform the following steps to integrate our platform with it:

1. On our Magento extension's page, provide the user API key and the event API key from your Blueshift account and click **Save Config**. You can get those keys from the **API keys** tab on the **Account Profile page**.



Historical Data Synchronization [store view]: In Active

Real Time Synchronization [store view]: In Active

User API Key [store view]: ································

Event API Key [store view]: ································

How much historical data to be sent to Blueshift based on its creation date.

From [store view]: 2019-06-01

Unsubscribe Data [store view]: No

Catalog Name [store view]: test

New Catalog [store view]:

If you run into an error, expand the **Synchronization Data and Catalog** section and choose a date in the **From** field. The fields in this section are described in the next step.

2. After the configuration is saved, provide the following details in the **Synchronization Data and Catalog** section:

From	Select the date from which you want to synchronize your store's data with our platform. You can choose a date that is as early as the date when you created your store.

	The extension can fetch historical data of the users, products, and orders on your store since the date you started your store.
Unsubscribe Data	Select **Yes** if you want Blueshift to add the information that a user has unsubscribed to marketing communications sent from our platform to the user's profile on your Magento store. Otherwise, select **No**.
Catalog Name	Select a catalog from the list of all catalogs that you created on our platform to which you want to add products of your Magento store. You can also create a new catalog.
New Catalog	Click the **Or Create a Catalog** button to: ○ Enable this field ○ Provide a name for the catalog that you create

3. Click **Save Config**.
4. In the **Status** section, click **Start Synchronization** to synchronize data from your store with our platform.

The extension uses JavaScript and Magento events to track events on your Magento store that occur when a user performs an action on your store, and sends them to our platform. Here is the list of events that the Magento extension collects:

- The extension uses a JavaScript (that it inserts at the time of setup) to track the following events:
    - Login activities
        - Login
        - Logout
    - Views
        - Page views
        - Product views
- The extension tracks the following events through the Magento events that it captures
    - Cart activities
        - Cart create
        - Cart update
        - Checkout create
    - Order activities
        - Order create
        - Order cancel
        - Fulfilment create
        - Fulfilment update
        - Refund create
    - Customer related activities
        - Customer create
        - Customer update
        - Customer delete
    - Product related activities

- Product create
- Product delete
- Product update