How To Install a Magento 2 Extension

1. Upload the Magento 2 extension files to your server.

2. Extract the *.zip* extension file.

3. Verify and copy the Magento extension files into the **app/code** folder.

4. Install the Magento 2 extension and check its status.

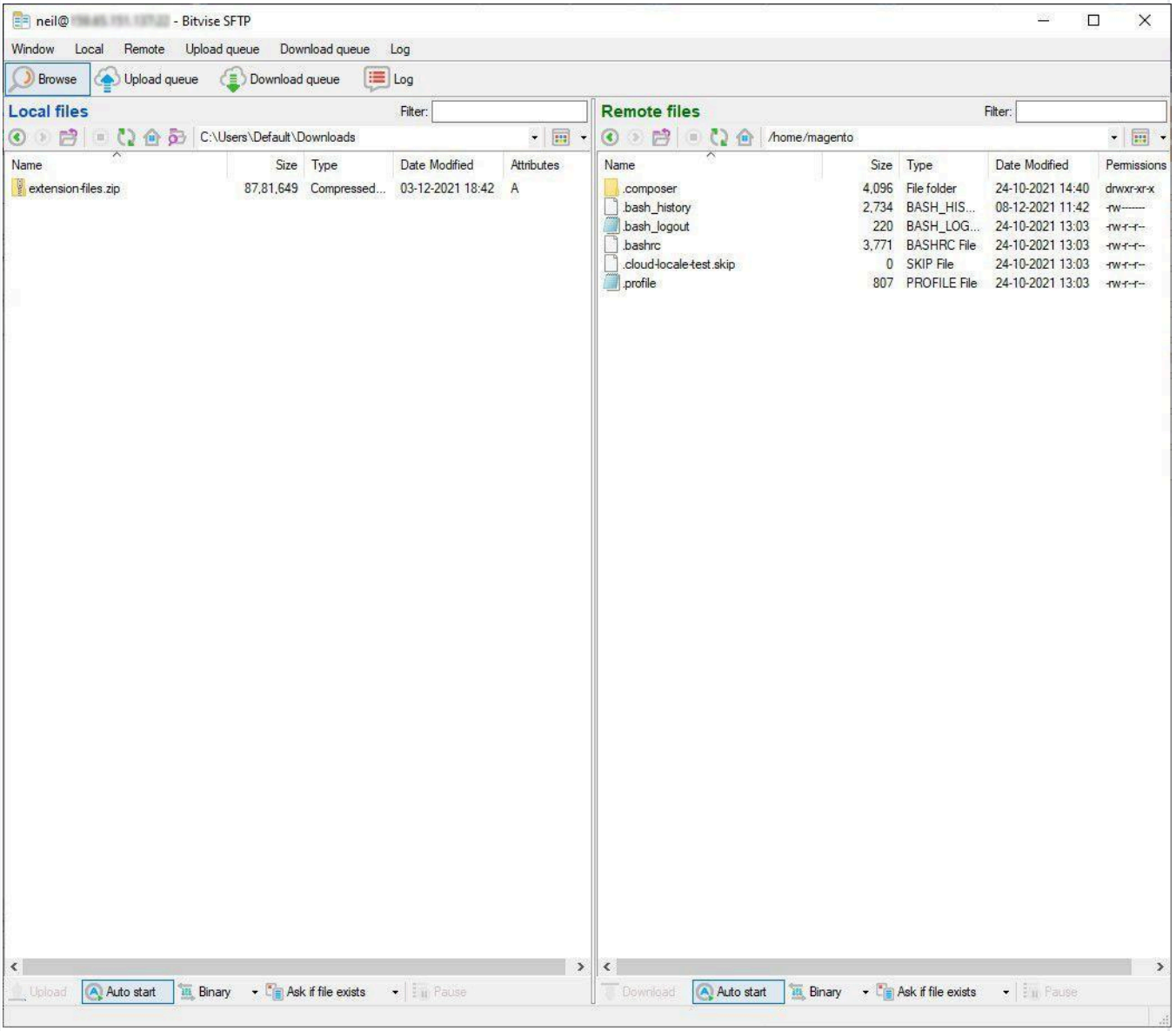
5. Clear Magento cache and disable maintenance mode.

Before you install a Magento 2 extension, make sure you backup your server, set pre-install file permissions, enable developer mode, and put Magento in the maintenance mode.

Step 1: Upload the Magento 2 Extension Files to Your Server

Log in to your Magento server as the file system owner.

Upload the *.zip* archive file to a new folder inside your user's home directory using SSH or SFTP.



Step 2: Extract the .zip Extension File

Extract the *.zip* extension file into a new folder called **extension** using:

```
$ unzip redtrack-magento.zip -d /home/magento/extension
```

Once you finish the extension installation process, you can delete this folder using:

```
$ rm -R /home/magento/extension
```

Step 3: Verify and Copy the Magento Extension Files Into the app/code Folder

Verify the extracted component file structure meets Magento's required "vendor/module-name" format.

Navigate to the Magento project root directory and copy the extension files to the **app/code** folder using:

```
$ cp -R /home/magento/extension/* app/code
```

The **app/code** directory doesn't exist in the default Magento filesystem. If this is your first attempt to install a Magento extension manually, create the directory using:

```
$ mkdir app/code
```

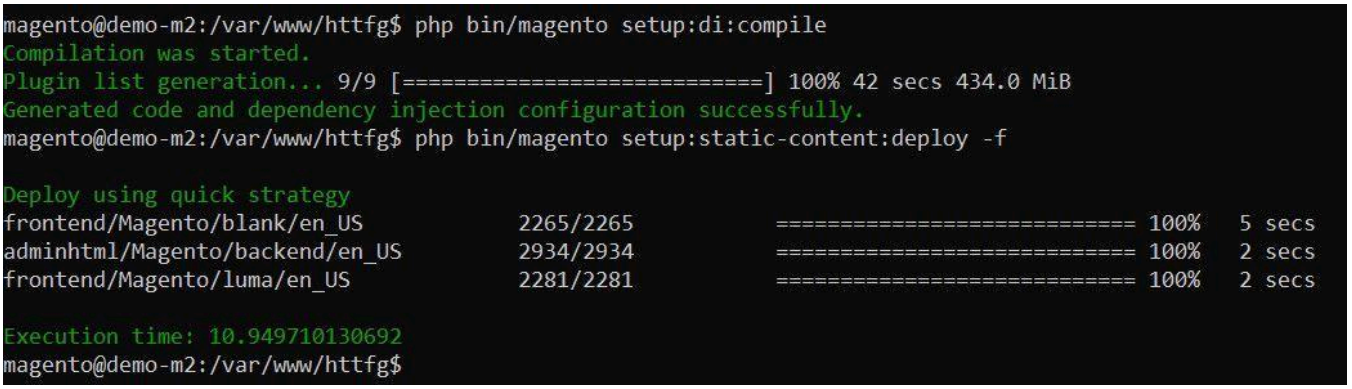
Step 4: Register the Magento 2 Extension and Check Its Status

Complete the Magento 2 extension installation process by executing:

```
$ php bin/magento setup:upgrade

$ php bin/magento setup:di:compile

$ php bin/magento setup:static-content:deploy -f
```



The "setup:upgrade" command registers the Magento 2 extension and updates the Magento database schema and data. The subsequent commands compile Magento code and deploy static view files.

Note: You don't need to deploy static view files in developer mode unless your extension developer's documentation mandates it.

Verify the status of the extension:

```
$ php bin/magento module:status RedTrack_UniversalScript
```

If it's disabled, you can enable it using:

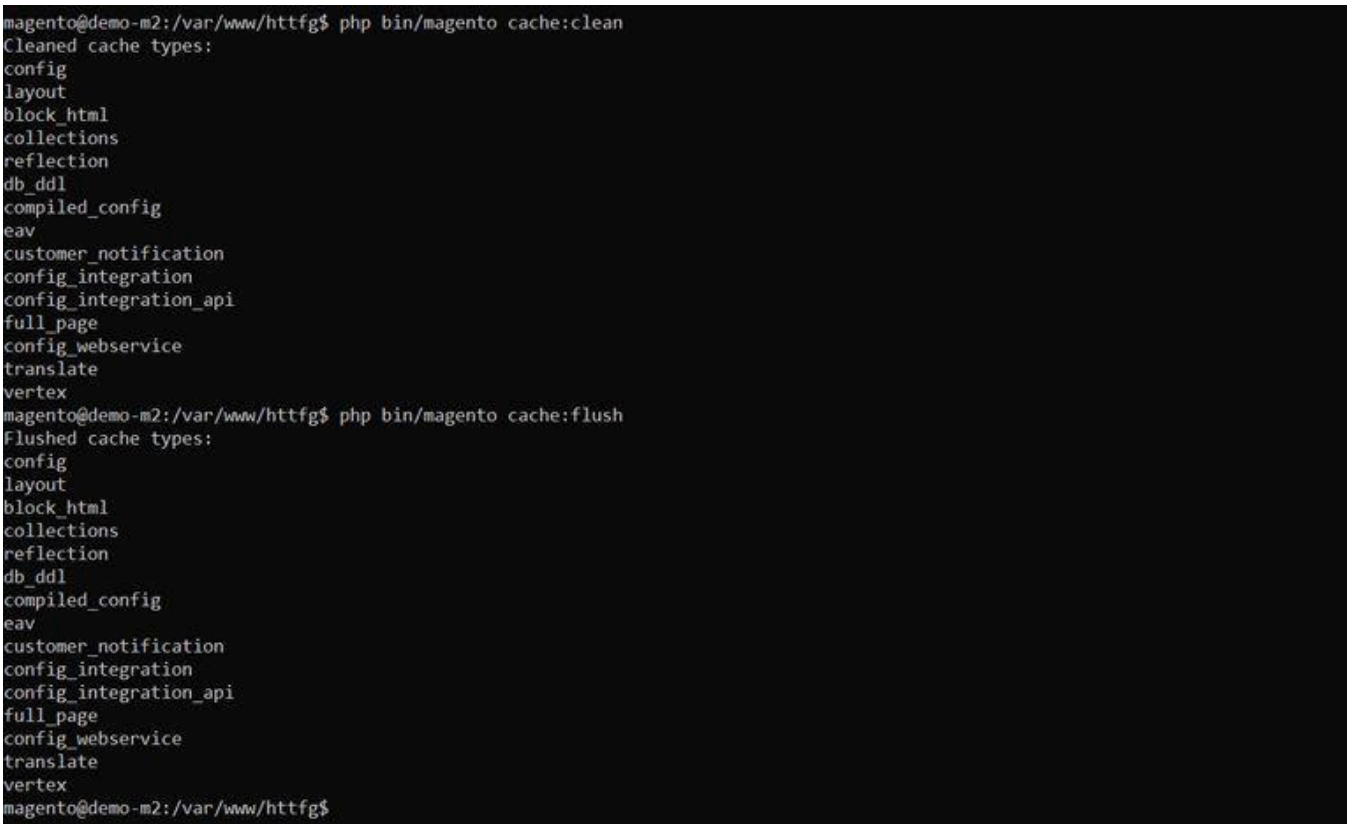
```
$ php bin/magento module:enable RedTrack_UniversalScript
--clear-static-content
```

Step 5: Clear Magento Cache and Disable Maintenance Mode

Clear the Magento cache using:

```
$ php bin/magento cache:clean

$ php bin/magento cache:flush
```



Follow Adobe's guide to set permissions for a production file system. Then, disable maintenance mode using:

```
$ php bin/magento maintenance:disable
```