

1. Sphinx Search Installation Guides

A) Environment Setup

First, you need to add Sphinx Search repository into your system and update the package list

```
sudo add-apt-repository ppa:builds/sphinxsearch-rel122
sudo apt-get update
```

Next, you can install sphinx search by the following command:

```
sudo apt-get install sphinxsearch
```

To check whether sphinx search has been properly installed, type

```
searchd
```

Into your console, the outcome should look like the below

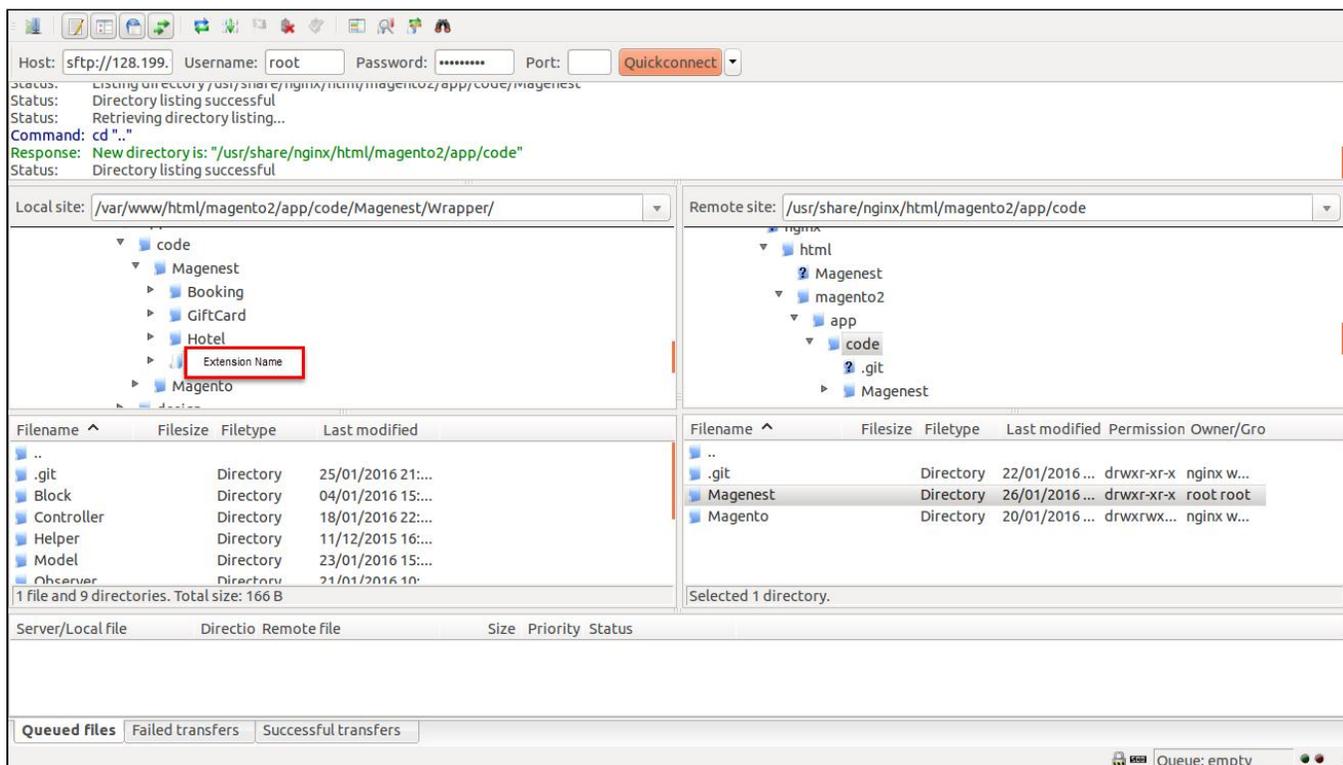
```
Sphinx 2.2.11-id64-release (95ae9a6)
Copyright (c) 2001-2016, Andrew Aksyonoff
Copyright (c) 2008-2016, Sphinx Technologies Inc (http://sphinxsearch.com)
```

Finally, you need to go to Magento's root folder and run the following commands to install additional libraries

```
composer require neutron/sphinxsearch-api
composer require foolz/sphinxql-query-builder
```

B) Extensions Install

Here we will guide you through the installation process of our extension. First off, open FileZilla, log in to your host as follows:



In the above example we switched the local site to the extension directory, and the remote site to the code directory. Now on the remote site window, inside app/code directory, create a folder named **Magenest** and go inside it. Now on your local site window, right click on **Extension Name** and click **Upload**. The extension will be uploaded onto your host in a few minutes.

The second method will require you to compress the extension into .zip or .tar.gz format, then execute the following command in your Command Line /Terminal:

```
scp <path to your extension> <ssh server>:<path to Magenest directory>
```

Then log into your ssh server, go to Magenest directory and execute the following command to decompress the module:

```
unzip <.zip compressed extension>
```

Or

```
tar -xzf <.tar.gz compressed extension>
```

Now go to your **Magento root directory**, say it's named **magento2**, and execute the following commands:

```
php bin/magento setup:upgrade
```

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

After running all these, your Command line/Terminal window might look like this:

```
Module 'Magento_BundleSampleData':
Module 'Magento_Rss':
Module 'Magento_DownloadableSampleData':
Module 'Magento_Captcha':
Module 'Magento_OfflineShippingSampleData':
Module 'Magento_ConfigurableSampleData':
Module 'Magento_SalesSampleData':
Module 'Magento_ThemeSampleData':
Module 'Magento_ProductLinksSampleData':
Module 'Magento_ReviewSampleData':
Module 'Magento_SendFriend':
Module 'Magento_Ui':
Module 'Magento_Sitemap':
Module 'Magento_Authorizenet':
Module 'Magento_Swagger':
Module 'Magento_Swatches':
Module 'Magento_SwatchesSampleData':
Module 'Magento_CatalogRuleSampleData':
Module 'Magento_TaxImportExport':
Module 'Magento_TaxSampleData':
Module 'Magento_CatalogRuleConfigurable':
Module 'Magento_CmsSampleData':
Module 'Magento_Translation':
Module 'Magento_Shipping':
Module 'Magento_Ups':
Module 'Magento_UrlRewrite':
Module 'Magento_GroupedProductSampleData':
Module 'Magento_USps':
Module 'Magento_Variable':
Module 'Magento_Version':
Module 'Magento_Webapi':
Module 'Magento_SalesRuleSampleData':
Module 'Magento_CatalogWidget':
Module 'Magento_WidgetSampleData':
Module 'Magento_Wishlist':
Module 'Magento_WishlistSampleData':
Please re-run Magento compile command
[root@centos-2gb-sgp1-01 magento2]# sudo chmod -R 777 var pub
[root@centos-2gb-sgp1-01 magento2]#
```

Finally, you need to flush the cache by either type this into the Command Line/Terminal:

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

Or go to **System > Cache Management**. Click the following buttons to completely clear the store cache:

1. Flush Magento Cache
2. Flush Cache Storage
3. Flush Catalog Images Cache
4. Flush JavaScript/CSS Cache

System

Tools

Cache Management

Index Management

Flush Cache Storage

Flush Magento Cache

	Description	Tags	Status
<input type="checkbox"/>	Various XML configurations that were collected across modules and merged.	CONFIG	DISABLED
<input type="checkbox"/>	Layout building instructions.	LAYOUT_GENERAL_CACHE_TAG	DISABLED
<input type="checkbox"/>	Page blocks HTML.	BLOCK_HTML	DISABLED
<input type="checkbox"/>	Collection data files.	COLLECTION_DATA	DISABLED
<input type="checkbox"/>	API interfaces reflection data.	REFLECTION	DISABLED
<input type="checkbox"/>	Results of DDL queries, such as describing tables or indexes.	DB_DDL	DISABLED
<input type="checkbox"/>	Entity types declaration cache.	EAV	DISABLED
<input type="checkbox"/>	Integration configuration file.	INTEGRATION	DISABLED
<input type="checkbox"/>	Integrations API Configuration Integrations API configuration file.	INTEGRATION_API_CONFIG	DISABLED
<input type="checkbox"/>	Page Cache Full page caching.	FPC	DISABLED
<input type="checkbox"/>	Translations Translation files.	TRANSLATE	DISABLED

Now you can go to your store and check if the extension is working as expected.