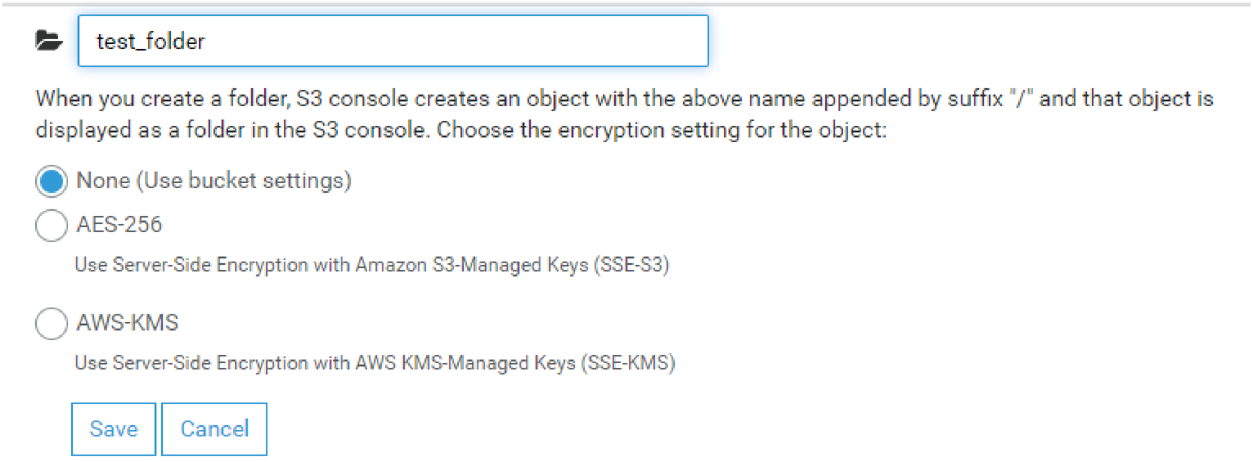
Uploading Data Sets and Files to S3

With Spark EMR clusters, you can use data sets easily by first uploading them to your S3 buckets and then using them in your Jupyter Notebooks. You can upload them directly from the AWS S3 UI, although that would be a slow process since you will have to first download the data sets to your local machines and then upload them.

Instead, you can download the data sets directly to the EC2 instances that you can already access and then simply upload the files to S3. This will save the time and bandwidth on downloading the data sets and then uploading them to S3 through the AWS UI.

To do this, first, create a folder in the S3 bucket. Go to your S3 UI and create a new folder in your S3 bucket. Then click on the 'Save' button.



test_folder

When you create a folder, S3 console creates an object with the above name appended by suffix "/" and that object is displayed as a folder in the S3 console. Choose the encryption setting for the object:

● None (Use bucket settings)

○ AES-256

Use Server-Side Encryption with Amazon S3-Managed Keys (SSE-S3)

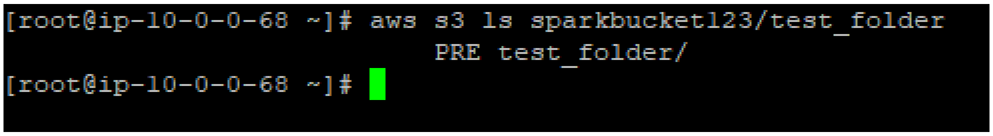
○ AWS-KMS

Use Server-Side Encryption with AWS KMS-Managed Keys (SSE-KMS)

The folder has now been created in the S3 bucket. Now, you will be syncing all the files that you will be downloading to your EC2 instance to this particular folder.

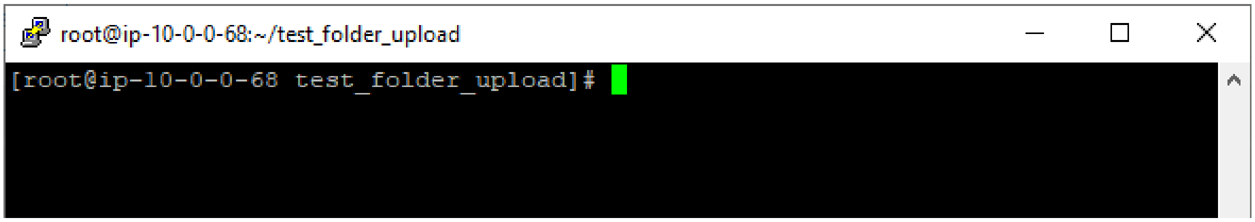
Follow the steps below to sync the files with the folder created in the S3 bucket:

Step 1: First, check the full path of the folder where you will be syncing the files in the S3 bucket. Do this by using the ‘aws s3 ls <bucket name>/<folder name>’ command:



```
[root@ip-10-0-0-68 ~]# aws s3 ls sparkbucket123/test_folder
                           PRE test_folder/
[root@ip-10-0-0-68 ~]#
```

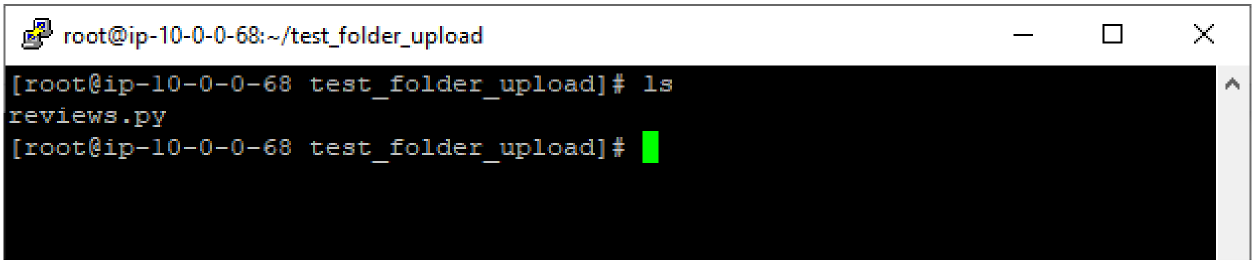
Step 2: Now that this folder has been created, in your EC2 instance, make a new folder, which you will be syncing with the folder in S3. Do this with the help of the mkdir command. After that, go to your newly created folder.



root@ip-10-0-0-68:~/test_folder_upload

```
[root@ip-10-0-0-68 test_folder_upload]#
```

Step 3: Next, you need to download the data set to this folder using the wget command. You will be given the download links to the various data sets in the platform text. For this demonstration, we are using a data set that is present in our S3 bucket itself.



root@ip-10-0-0-68:~/test_folder_upload

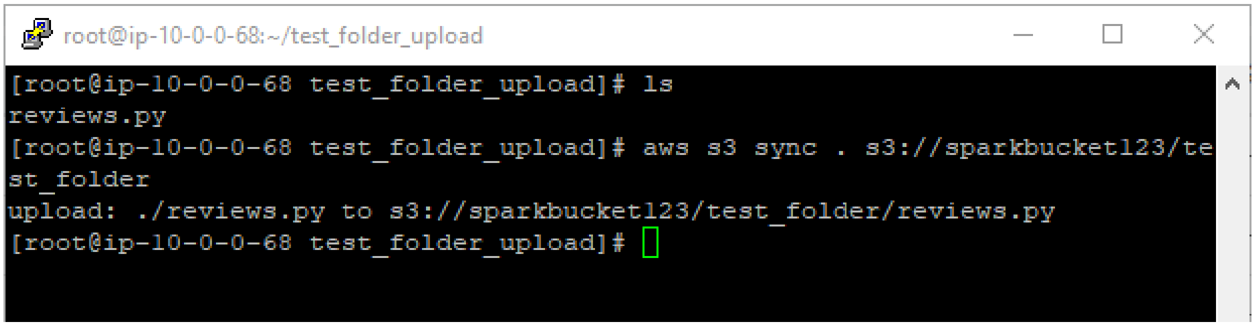
```
[root@ip-10-0-0-68 test_folder_upload]# ls
reviews.py
[root@ip-10-0-0-68 test_folder_upload]#
```

Step 4: Next, we will be using the aws s3 sync command to sync the files present in our folder with the test_folder, which we created in S3. To do this, type the following command and execute it:

```
aws s3 sync <source> <target>
```

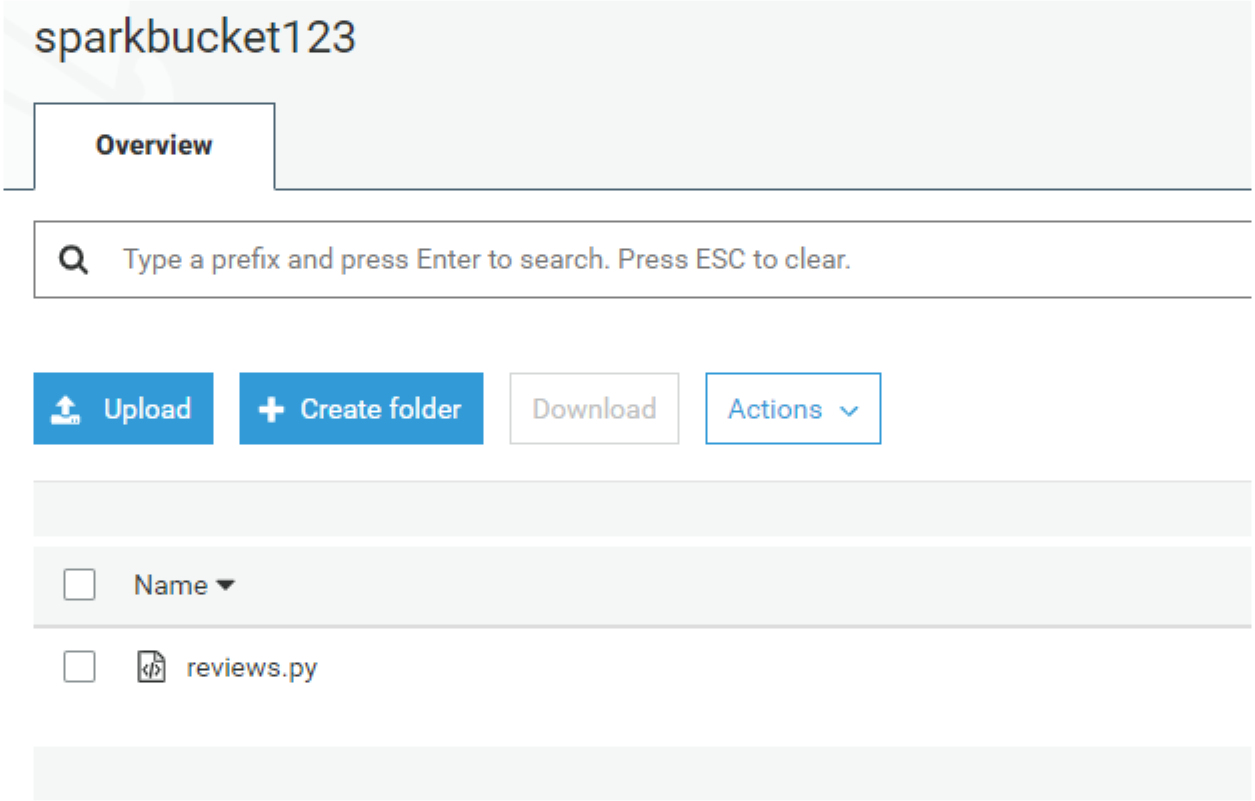
Here, we have to replace the target with ‘**s3://sparkbucket123/test_folder**’.

After executing the command, the following will be shown in the EC2 instance:



```
[root@ip-10-0-0-68 test_folder_upload]# ls
reviews.py
[root@ip-10-0-0-68 test_folder_upload]# aws s3 sync . s3://sparkbucket123/te
st_folder
upload: ./reviews.py to s3://sparkbucket123/test_folder/reviews.py
[root@ip-10-0-0-68 test_folder_upload]#
```

Now, you can again check the files in the S3 bucket using the AWS S3 UI.


sparkbucket123
Overview
Type a prefix and press Enter to search. Press ESC to clear.
Upload
Create folder
Download
Actions
Name
reviews.py