Apr 09, 2021

How to check the integrity of a set of files uploaded to AWS S3

Sonia García-Ruiz[1]

[1]University College London, University of London

1 *Works for me* This protocol is published without a DOI.

Ryten Laboratory

Sonia García-Ruiz

ABSTRACT

This protocol contains the instructions to verify the integrity of any small or large file uploaded to AWS S3.

This protocol has been successfully tested on:

- Ubuntu 16.04.6 LTS (Xenial Xerus)

PROTOCOL CITATION

Sonia García-Ruiz 2021. How to check the integrity of a set of files uploaded to AWS S3. **protocols.io** https://protocols.io/view/how-to-check-the-integrity-of-a-set-of-files-uploa-bt4tnqwn

KEYWORDS

aws-s3, s3-bucket, aws-cli, md5, cloud-computing, etag, md5sum, integrity monitoring

LICENSE

This is an open access protocol distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited

CREATED

Apr 09, 2021

LAST MODIFIED

Apr 09, 2021

PROTOCOL INTEGER ID

49011

MATERIALS TEXT

In this protocol, we are going to make use of the following GitHub repositories:

- s3md5: a bash script to calculate Etag/S3 MD5 sum for very big files uploaded using multipart S3 API.
- aws-s3-integrity-check: a bash script to check the integrity of a set of files uploaded to AWS S3.

BEFORE STARTING

1. Log in to your AWS account using the *'aws configure'* command in your Linux terminal.

IMPORTANT: for the correct operation of this protocol, *'json'* must be chosen as the preferred output format during the *'aws configure'* command execution.

```
> aws configure
AWS Access Key ID [None]: your_AWS_access_key_ID
AWS Secret Access Key [None]: your_AWS_secret_access_key
Default region name [None]: your_default_region_name
Default output format [None]: json
```

This is an open access protocol distributed under the terms of the **Creative Commons Attribution License** (https://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited

2. Upload your local data to your AWS S3 bucket (please, skip this step in case your files are already stored on AWS S3). To synchronize your local folder with your AWS bucket, you can use the AWS CLI *'sync'* command as follows:

```
> aws s3 sync /path-to-your-local-folder/your_data s3://bucket-name
```

1 Clone the s3md5 repo:

```
$ git clone https://github.com/antespi/s3md5.git
```

2 Grant execution access to the s3md5 script file.

```
$ chmod 755 ./s3md5/s3md5
```

3 Clone the aws-s3-integrity-check repo:

```
$ git clone https://github.com/SoniaRuiz/aws-s3-integrity-check.git
```

4 Move the *'s3md5'* folder within the *'aws-s3-integrity-check'* folder:

```
$ mv ./s3md5 ./aws-s3-integrity-check
```

5 The *'aws-s3-integrity-check'* folder should look now similar to the following structure:

```
$ ls ./aws-s3-integrity-check
total 16
-rw-r--r-- 1 your_user your_group 3573 date README.md
-rwxr-xr-x 1 your_user your_group 3301 date aws_check_integrity.sh
drwxr-xr-x 2 your_user your_group 4096 date s3md5
```

6 Execute the *'aws-check-integrity.sh'* bash script following the instructions below:

```
$ aws_check_integrity.sh <local_folder> <bucket_name> <bucket_folder>
```

Usage :

- **<local_folder>:** local path to the folder that contains all files previously uploaded to AWS. For example: */local-path/raw-data-folder/.*
- **<bucket_name>**: name of the S3 bucket containing the files uploaded to AWS that we want to check their integrity. For example: *'bucket-name'.*
- **<bucket_folder>**: folder name on the S3 bucket that contains all files to be checked. The name of this folder should be the same indicated on the *<local_folder>* parameter. For example: *raw-data-folder/*. In case there is not any root folder, this parameter will be substituted by a slash (/), which will indicate the root path.

Example:

```
$ aws_check_integrity.sh /local-path/raw-data-folder/ bucket-name raw-data-folder/
```

This is an open access protocol distributed under the terms of the **Creative Commons Attribution License** (https://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited

7 The execution of this script will:

1. Loop through all files stored on */local-path/raw-data-folder/*.
2. Per each file, the script will check its size. In case the object found is a directory, it will just continue looping through its child directory files.
3. If the file size is smaller than 8MG, the script will generate a simple MD5 digest value.
4. If the file size is bigger than 8MG, it will make a request to the *s3md5* script, which will apply the same algorithm as AWS does: it will split the file into 8MG small parts, generate the MD5 hash corresponding to each small part and generate the final MD5 digest number from the total set of individual MD5 hashes.
5. Retrieve the ETag value from the corresponding file stored on the S3 bucket.
6. Compare the retrieved ETag value with the integrity number generated by the script.
7. If both numbers are identical, the script will confirm the integrity of the file stored on the S3 bucket. Otherwise, the script will generate an error. In any case, the result will be stored within a log file whose name will follow the following pattern: S3_integrity_log.timestamp_bucketname.txt. The log file will be stored within a folder called 'logs'. In case this folder doesn't exist, it will be automatically created by the script in the same path in which it has been executed.

This is an open access protocol distributed under the terms of the **Creative Commons Attribution License** (https://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited