| **Name:** | Lin Mei An |
| --- | --- |
| **Email:** | lin\_mei\_an@mpa.gov.sg |

**Question 1**

Amazon’s Simple Storage Service (S3) is an object storage service; objects are stored as key/value pairs. Objects can be added or read from S3 either through the web console or through RESTful API.

Read the following documentation of inserting an object into an S3 bucket

<https://docs.aws.amazon.com/AmazonS3/latest/API/API_PutObject.html>

and answer the following questions

1. What content type is supported by S3 in the put object operation?

All Standard MIME types

1. How does S3 ensure payload integrity viz. the object that is uploaded, especially large object, has not been corrupted.

S3 uses the Content-MD5 header that checks the object against the provided MD5 value,

Or, the user can calculate the MD5 and compare the returned ETag to the calculated MD5 value.

* checksum?

1. What other algorithms (wrt b) does S3 support? How are these algorithms specified?

-CRC32

-CRC32C

-SHA1

-SHA256

1. How does S3 ensure content confidentiality?

server side encryption → encryption key can either be user provided or aws managed

1. What strategy does this operation use to support S3 features (eg. encryption, storage classes, etc.) when an object is uploaded?

In your opinion, how are new S3 features supported by this operation?

* key management
* request payer
* storage class

included in the headers

1. How does the put operation support caching?

* cache control

1. How does the operation ensure that all the required parameters (eg. bucket name, encryption key, credentials, etc.) are correct be committing to the put operation?

* 400 error AccessControlistNotSupported →PUT requests that contain other ACLs

1. What are the main differences between this operation (PutObject) and PostObject (<https://docs.aws.amazon.com/AmazonS3/latest/API/RESTObjectPOST.html>)

POST is an alternate form of PUT that enables browser-based uploads as a way of putting objects in buckets. Parameters that are passed to PUT through HTTP Headers are instead passed as form fields to POST in the multipart/form-data encoded message body.

1. S3 charges includes egress, viz. amount of data transferred out from a S3 bucket. If your server is using S3 for data storage, how do you reduce your S3 charges?

store on cache

**Submission**

Copy this Word document to your repository and commit it.

git add .

git commit -m ‘worksheet02’

git push origin master