CloudCubes

Follow

All you need to know about encrypting Amazon S3 buckets

Jineshkumar Patel

Sep 10, 2022 · 8 min read

Table of contents

Security is often a deciding factor when choosing a public cloud provider. Encryption of data at rest is increasingly required by industry protocols, government regulations, and internal organizational security standards. Encryption helps you protect your stored data against unauthorized access and other security risks.

That way, even if there are any security breaches or attacks on your company’s system, all of the information will be protected.

> *"Dance like nobody is watching. Encrypt like everyone is."*
> *-Werner Vogels - VP & CTO - Amazon.com*

Is your Data Protected at-rest in S3 Buckets ?
Is your Data Encrypted in-transit to & from S3 Buckets ?

Not sure?
Don't Worry. This Blog will cover almost all aspects of S3 Bucket Data Protection and Encryption with Why and How to do so.

And if you are doing so or know few aspects of it, This blog will help you avoid common but critical pitfalls.

> *Data protection refers to protecting data while*
> *In-transit (as it travels to and from Amazon S3) and*
> *At rest (while it is stored on disks in Amazon S3 data centers)*

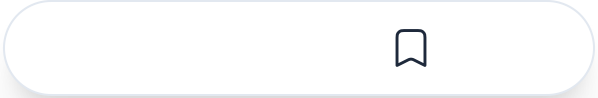
Key Objects to Cover

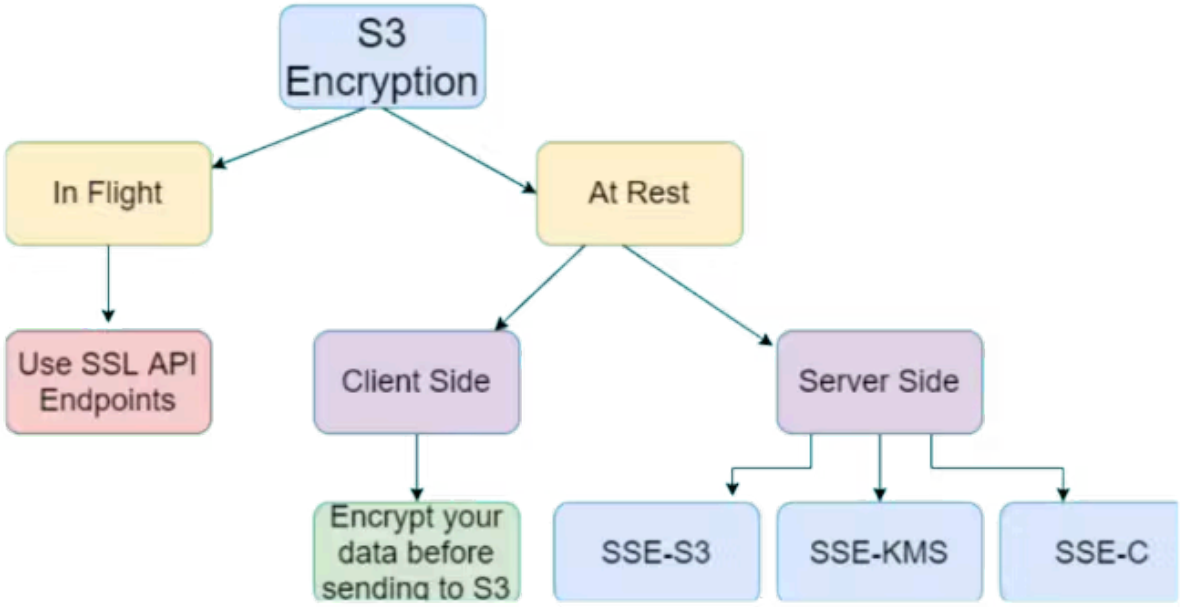
- Options for S3 Encryption: Server-Side(SSE) or Client-Side Encryption
- How to configure S3 Default Encryption ?
- Common Best Practices for Data Protection and Compliance
- Avoid Unnecessary Costs when Enabling SSE.
- Why S3 Bucket Policy is Important to Enforce Encryption
- Cost of Encryption

Options for protecting data "at-rest" in Amazon S3:

Server-Side Encryption – Request Amazon S3 to encrypt your object before saving it on disks in its data centers and then decrypt it when the Customer request to download the objects. To configure server-side encryption,
During Creation of a Bucket, Enable server-side encryption with

- AWS Key Management Service key (SSE-KMS)
  or
- Specify Amazon S3-managed keys (SSE-S3) or
- Customer-provided encryption keys (SSE-C)





Client-Side Encryption – Encrypt data client-side and upload the encrypted data to Amazon S3. In this case, Customer manage the encryption process, the encryption keys, and related tools. Client-side encryption is the act of encrypting data locally to ensure its security as it passes to the Amazon S3 service. The Amazon S3 service receives your encrypted data and it does not play a role in encrypting or decrypting it.
To enable client-side encryption, you have the following options:

- Use a key stored in AWS Key Management Service (AWS KMS).
- Use a key that you store within your application. (Not-Recommended) AWS's Note : Your client-side keys and your unencrypted data are never sent to AWS. It's important that you safely manage your encryption keys. If you lose them, you can't decrypt your data.

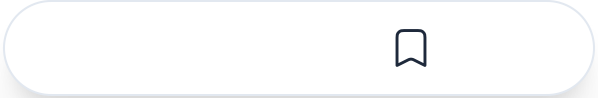
Setting defau n" behavior for Amazon S3

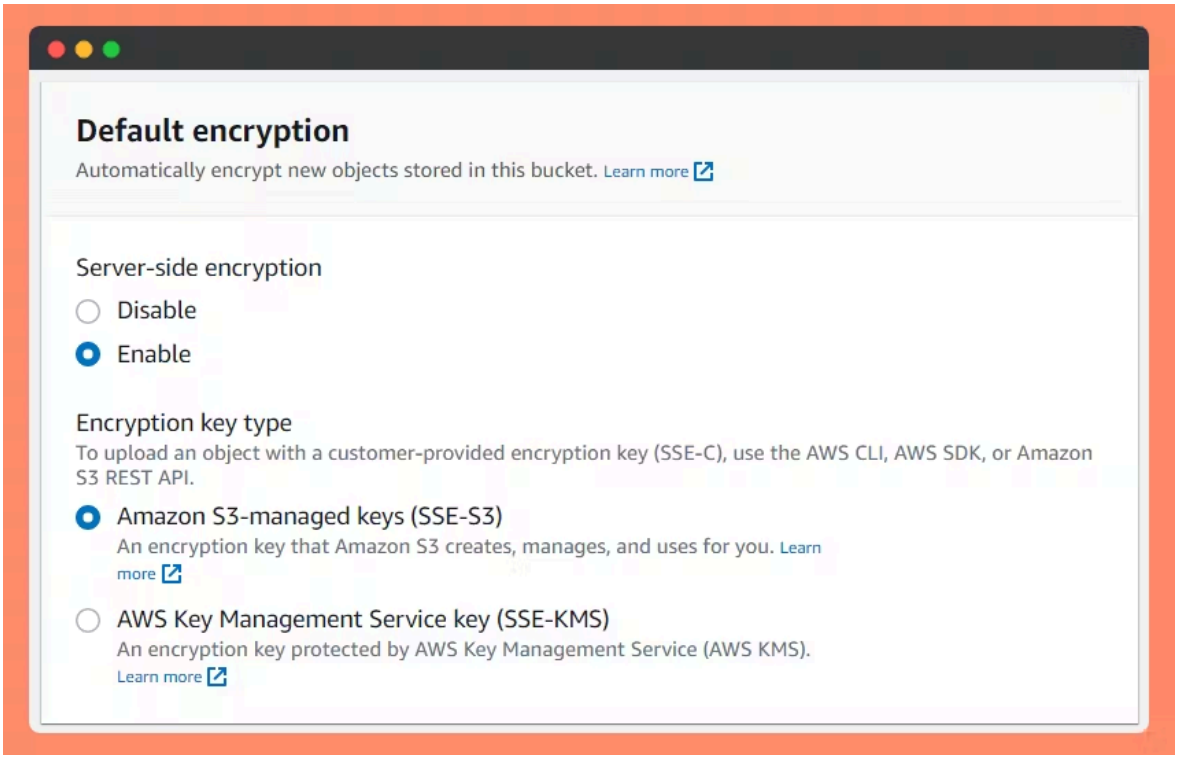
> *With Amazon S3 default encryption, you can set the default encryption behavior for an S3 bucket so that all new objects are encrypted when they are stored in the bucket.*
> *The objects are encrypted using server-side encryption with either Amazon S3-managed keys (SSE-S3) or AWS KMS keys stored in AWS Key Management Service (AWS KMS) (SSE-KMS).*

S3 Default Encryption

When you configure your bucket to use default encryption with SSE-KMS, you can also enable S3 Bucket Keys to decrease request traffic from Amazon S3 to AWS Key Management Service (AWS KMS) and reduce the cost of encryption.

> *When Creating an S3 Bucket, there will be option to Enable SSE*
> *Note: There are no additional fees for using server-side encryption with Amazon S3-managed keys (SSE-S3). However, requests to configure the default encryption feature incur standard Amazon S3 request charges. See S3 Pricing*





Encrypting existing unencrypted objects

To encrypt your existing Amazon S3 objects, you can use Amazon S3 Batch Operations. You provide S3 Batch Operations with a list of objects to operate on, and Batch Operations calls the respective API to perform the specified operation.

> *You can use the Batch Operations Copy operation to copy existing unencrypted objects and write them back to the same bucket as encrypted objects. A single Batch Operations job can perform the specified operation on billions of objects. More on that here*
>
> *I would love to perform this Batch Operation for S3 Encryption as a Hands-On Lab in upcoming Blogs. (Added in To-Do List)*

Encrypting existing Amazon S3 objects with the AWS CLI

Simply, If you must encrypt all objects in your S3 bucket, you can run the following command: (More options on this here)

```
aws s3 cp s3://awsexamplebucket/ s3://awsexamplebucket/ --sse
```

COPY

Is your Data Encrypted in-transit to & from S3 Buckets ?

- Data is more vulnerable when it's in motion. To protect data in transit, companies should implement network security controls like firewalls and network access control. These will help secure the networks used to transmit information against malware attacks or intrusions.
- SSL/TLS uses both asymmetric and symmetric encryption to protect the confidentiality and integrity of data-in-transit. Both the client and server use HTTPS (SSL/TLS + HTTP) for their communication and can be used for File(Data) Transfer.
- TLS is a cryptographic protocol that provides end-to-end security of data sent between applications over the Internet. It should be noted that TLS does not secure data on end systems(When at-rest). It simply ensures the secure delivery of data over the Internet, avoiding possible eavesdropping and/or alteration of the content.

Why S3 Bucket Policy is Important to Enforce Encryption

- Bucket Policy can be set to prevent Users or Applications requests to Put Objects without Encryption or with different encryption method.
- Bucket Policy is Important with Default Encryption for the Bucket to make sure all the objects in the bucket comply with Certain Encryption Standard. (Making the Data Protection Officer Happy 🤵 😎😊🕵 for Compliance !!)
- In order to enforce object encryption on S3 Bucket, create an S3 bucket policy that denies any S3 Put request that does not include the x-amz-server-side-encryption header. There are two possible values for the x-amz-server-side-encryption header: AES256, which tells S3 to use S3-managed keys, and aws:kms, which tells S3 to use AWS KMS–managed keys.

the following bucket policy denies permissions to upload an object unless the request includes the x-amz-server-side-encryption header : AES256 to request server-side encryption:

COPY

```
{
  "Version": "2012-10-17",
  "Id": "PutObjectPolicy",
  "Statement": [
    {
      "Sid": "DenyIncorrectEncryptionHeader",
      "Effect": "Deny",
      "Principal": "*",
      "Action": "s3:PutObject",
      "Resource": "arn:aws:s3:::awsexamplebucket1/*",
      "Conditi
        "StringNotEquals": {
```

```
            "s3:x-amz-server-side-encryption": "AES256"
          }
        }
      }
    ]
}
```

the Above bucket policy denies the upload object (s3:PutObject) permission to everyone if the request does not include the x-amz-server-side-encryption header requesting server-side encryption with SSE-KMS.

OR

COPY

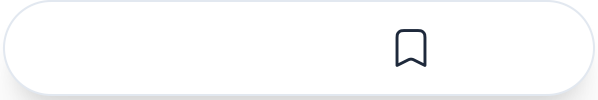
```
{
   "Version":"2012-10-17",
   "Id":"PutObjectPolicy",
   "Statement":[{
         "Sid":"DenyUnEncryptedObjectUploads",
         "Effect":"Deny",
         "Principal":"*",
         "Action":"s3:PutObject",
         "Resource":"arn:aws:s3:::DOC-EXAMPLE-BUCKET1/*",
         "Condition":{
            "StringNotEquals":{
               "s3:x-amz-server-side-encryption":"aws:kms"
            }
         }
      }
   ]
}
```

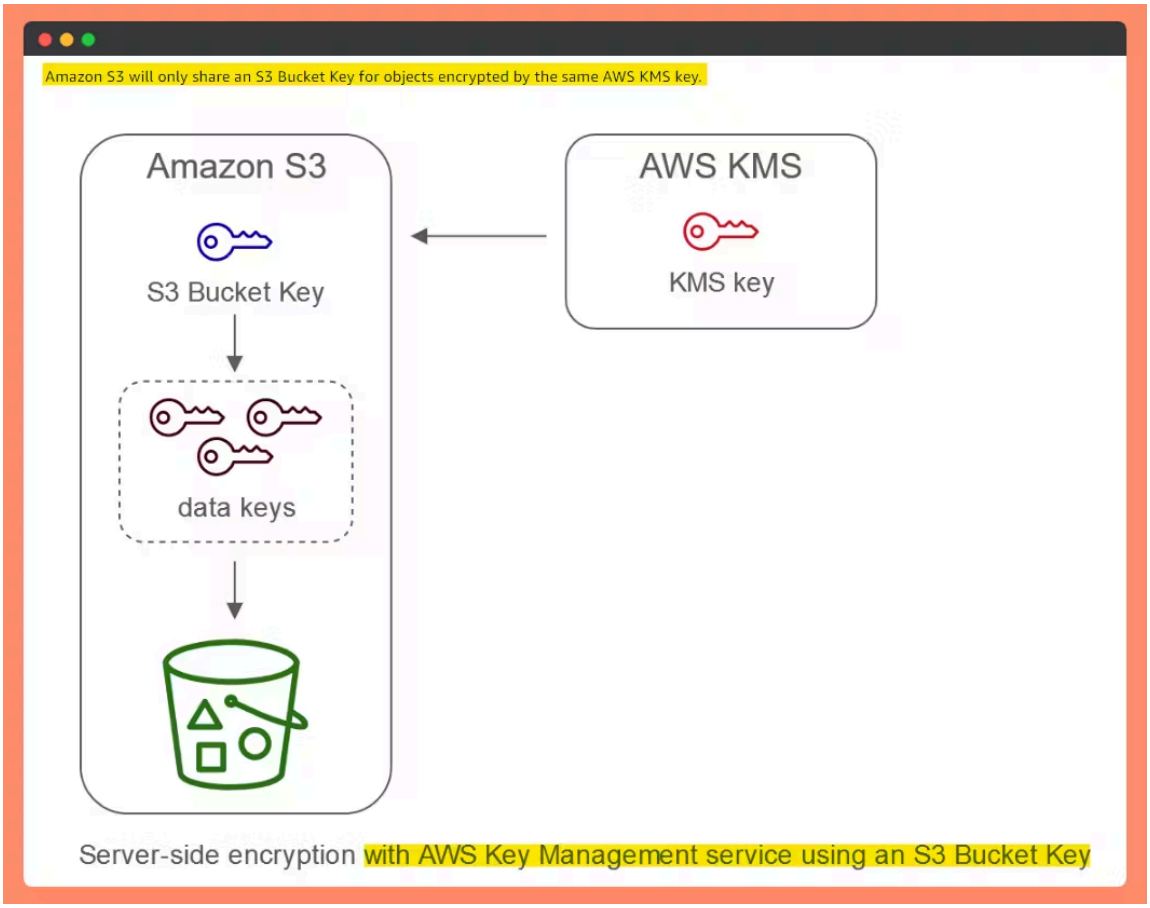
Both above Example Bucket Policy Enforcing Encryption to use "AES256" (SSE-S3) / aws:kms (Respectively) to Allow Put Objects to this Bucket.

Cost comes after Security

- Cost is also extremely important when dealing with Huge amount of Objects and their Buckets. And Enforcing S3 Security and Encryption comes with a Cost.
- Amazon S3 Bucket Keys reduce the request costs of Amazon S3 server-side encryption (SSE) with AWS Key Management Service (KMS) by up to 99% by decreasing the request traffic from S3 to KMS. With a few clicks in AWS Management Console and no changes to your client applications, you can configure your buckets to use an S3 Bucket Key for KMS-based encryption on new objects.
- ## Reduce the cost of S3 encryption

  When you configure default encryption to your bucket **with SSE-KMS**, you can also **enable S3 Bucket Keys** to decrease request traffic from Amazon S3 to AWS Key Management Service (AWS KMS) and reduce the cost of encryption. For more information, see Reducing the cost of SSE-KMS with Amazon S3 Bucket Keys.





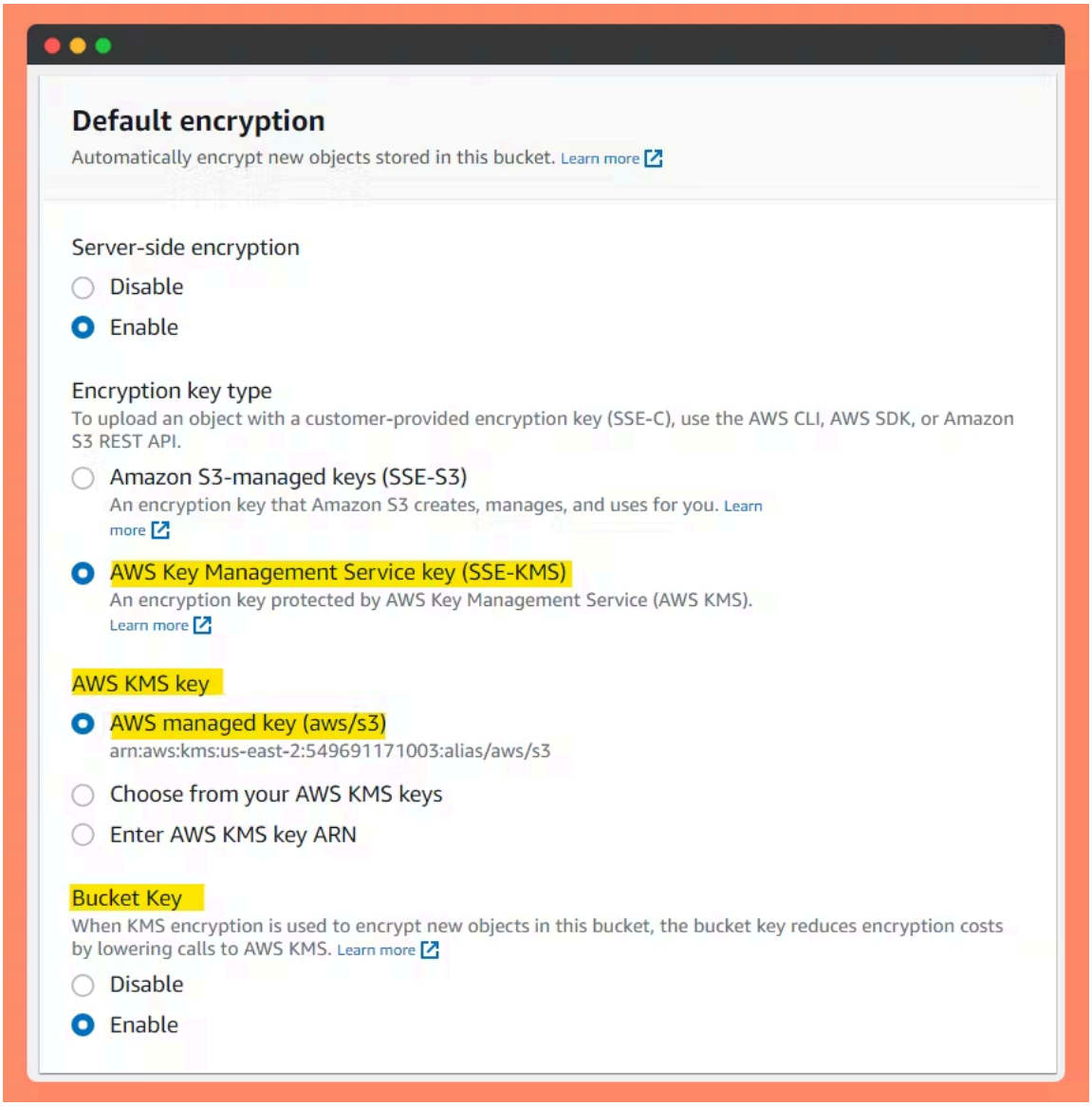
> *Workloads that access millions or billions of objects encrypted with SSE-KMS can generate large volumes of requests to AWS KMS.*

- When you use SSE-KMS to protect your data without an S3 Bucket Key, Amazon S3 uses an individual AWS KMS data key for every object. It makes a call to AWS KMS every time a request is made against a KMS-encrypted object.
- When you configure your bucket to use an S3 Bucket Key for SSE-KMS, AWS KMS generates a bucket-level key that is used to create unique data keys for new objects that you add to the bucket. This S3 Bucket Key is used for a time-limited period within Amazon S3, reducing the need for Amazon S3 to make requests to AWS KMS to complete encryption operations.

  This reduces t g to access AWS

KMS-encrypted objects in S3 at a fraction of Cost compared to the previous approach(without an S3 Bucket Key).

One Magic Trick that can save 99.8% on AWS S3 KMS charges



Let me Summarise what I explained in this Blog regarding S3 Encryption.

- Server-Side Encryption for S3 Buckets is Must for Data Protection.
- Enable Default Encryption

- Enforce Bucket Policy to Use S3 Encryption for Put Object.
- Encrypt Unencrypted Objects in Bucket using Batch Job / AWS CLI
- Use S3 Bucket Keys for SSE-KMS to Reduce Cost for Encryption.
- Use Bucket Level Keys instead of Object Level Keys to reduce Cost on KMS requests.

Hope you have Enjoyed the Blog. Thank you for Reading.

Feel free to ask questions about how to encrypt data at rest on S3.

Happy Learning.

Like and Follow for more Azure and AWS Content.

Thank you,
Jineshkumar Patel

Amazon S3 S3 encryption Security AWS

Written by

Jineshkumar Patel

Cloud Enthusiast working as Cloud Infrastructure Consultant. My Hobby is to build and destroy Cloud Projects for Blogs. Love to share my learning journey about DevOps, AWS and Azure.

Subscribe and Follow up with "CloudCubes".
Thank you and Happy Learning !!

Follow

MORE ARTICLES

Jineshkumar Patel

S3 Bucket is Mounted using Mountpoint for Amazon S3

S3 Mountpoint Introduction Mountpoint for S3 translates local file system API Calls to S3 Object API...

Jineshkumar Patel

Enabling or Disabling Detailed Monitoring for Amazon EC2 Instances

Monitoring your Amazon Elastic Compute Cloud (Amazon EC2) instances is crucial for ensuring optimal ...

Jineshkumar Patel

Optimizing S… Comprehen…

In today's digital landscape, efficient storage management is crucial for businesses of all sizes. W...