Mastering Data Management: Unleash the Power of Amazon S3 for Cloud Storage

Organize & Secure Data, Configure Permissions, Data Backup, Versioning & Retrieval

AutOps · Feb 24, 2024 · 7 min read

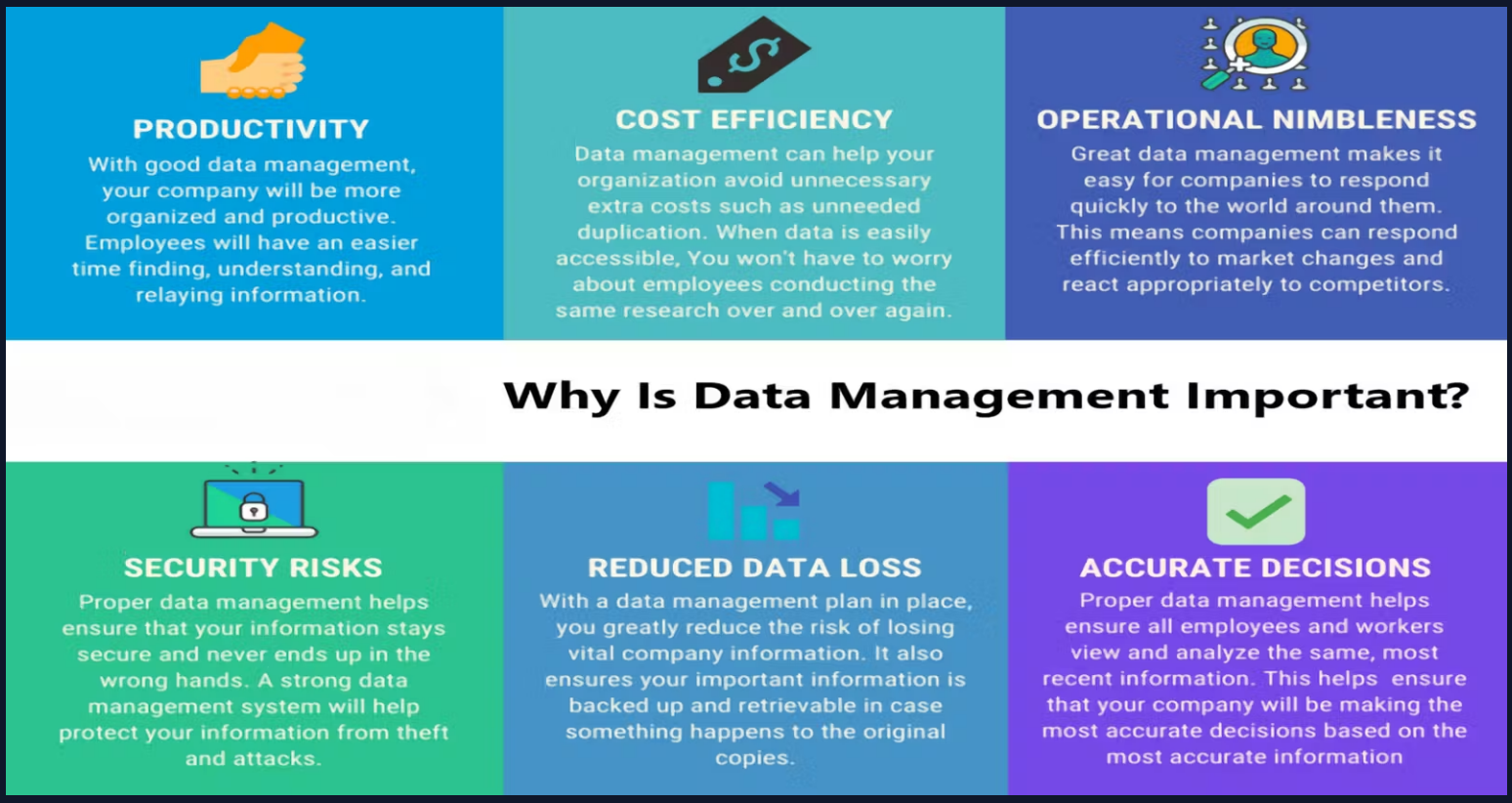


Table of contents

Author: Ujwal Pachghare 🌟

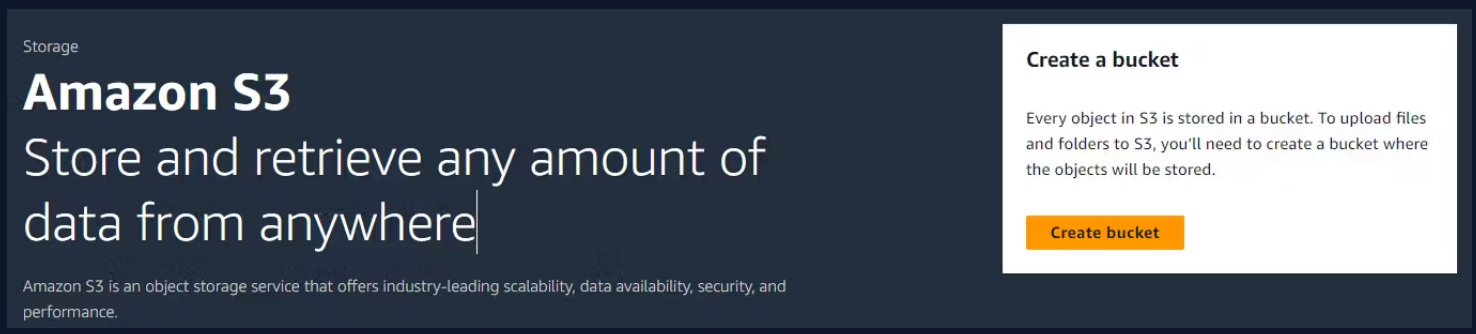
Introductio

*AWS S3, or Amazon Simple Storage Service, functions similarly to a big cloud-based digital filing cabinet. Consider that you require a location to store a large number of assets, including papers, films, images, and more. They may be stored on your computer, but what would happen if it broke or ran out of space? S3 fills that role.

S3 can be compared to a massive, safe, and dependable hard drive located high in the sky. It allows you to upload and download files, as well as organize them into various folders (referred to as "buckets" by S3).*

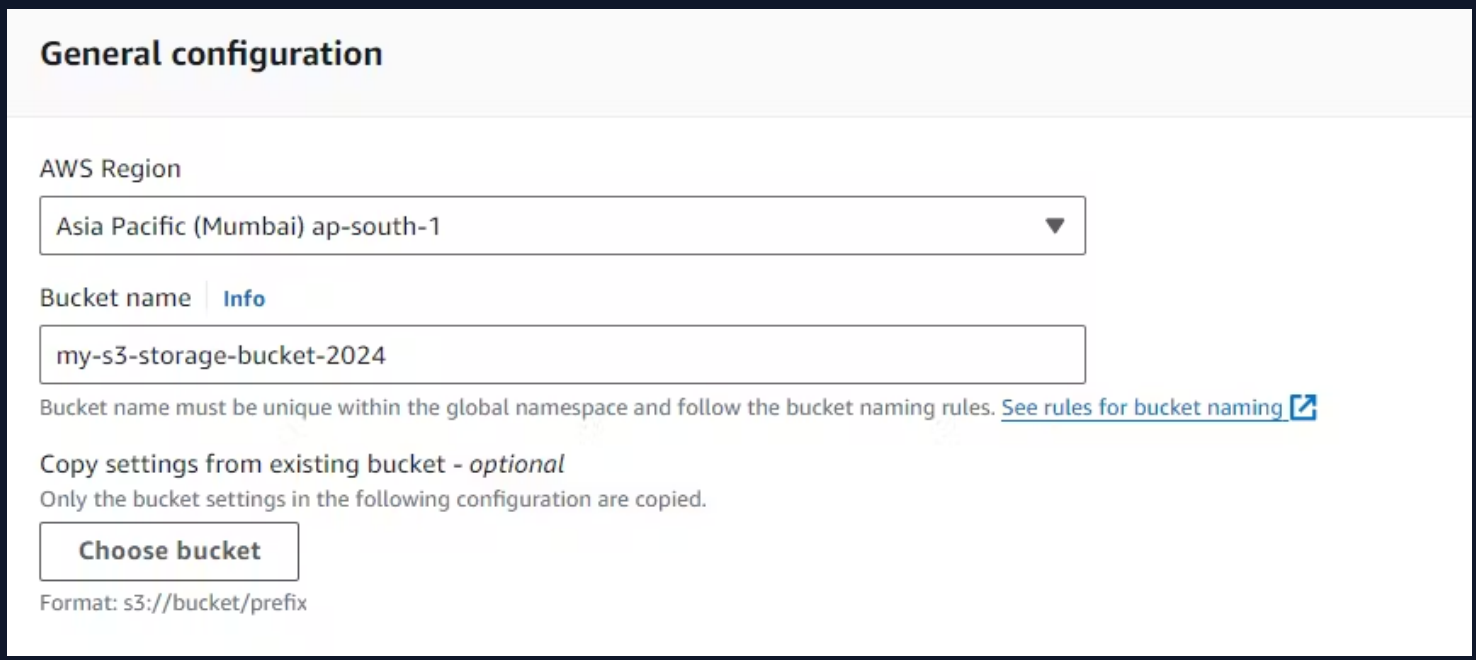
*The fact that S3 is "scalable," or able to expand with you, is one of its interesting features. S3 is capable of handling millions of files, even if you just have ten. You just pay for what you use, too.

Essentially, Amazon Web Services' S3 storage solution allows you to store and retrieve enormous volumes of data via the Internet[at any time and from any location. It functions similarly to a cloud-based personal storage unit!



1. General Configurations

👉 Go to Amazon **S3 Console** → Click on **Create Bucket** → Choose your region → Give your **S3 Bucket** a unique **name** → You can directly **copy** settings from **existing S3 bucket**

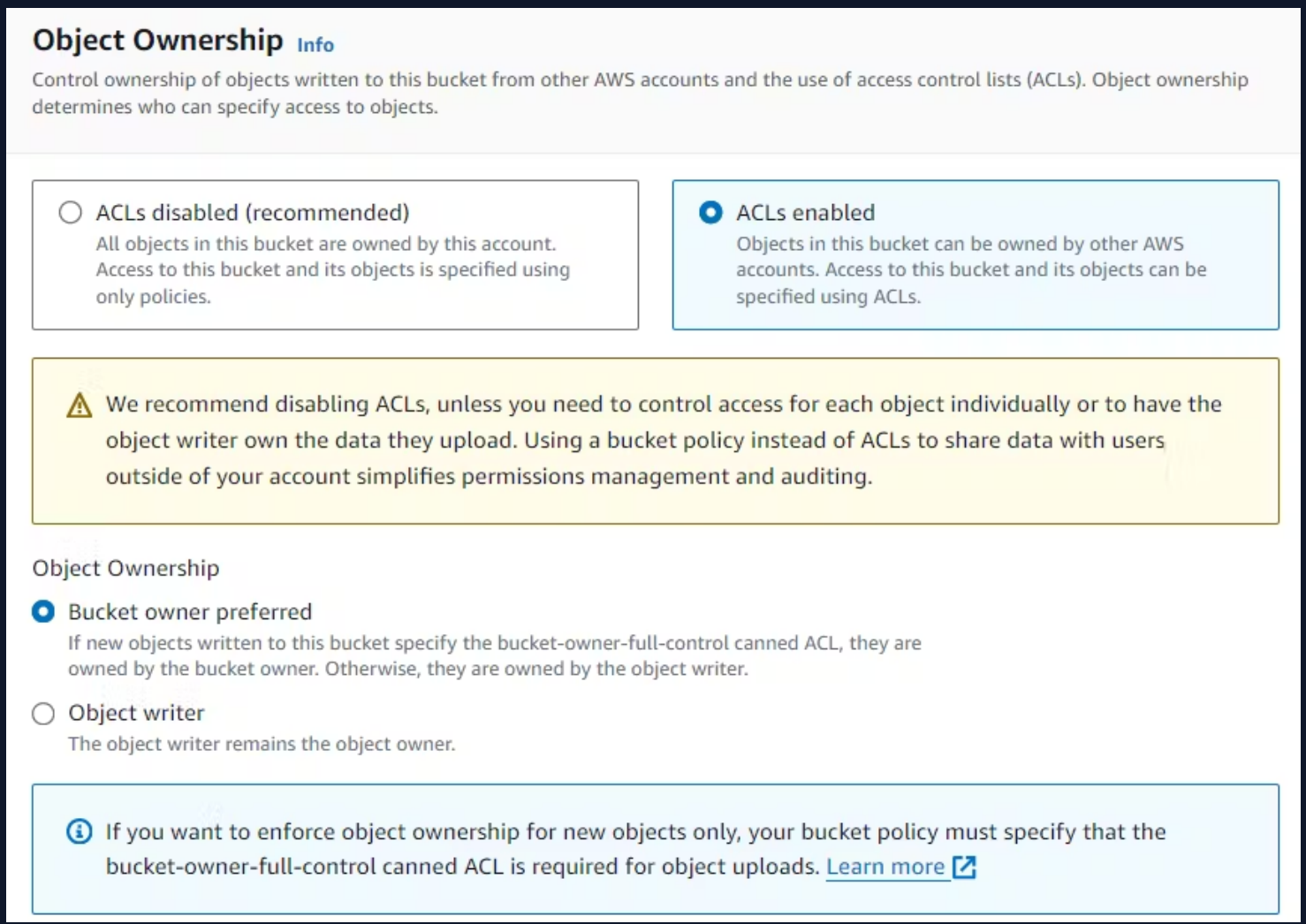


2. Object Ownership

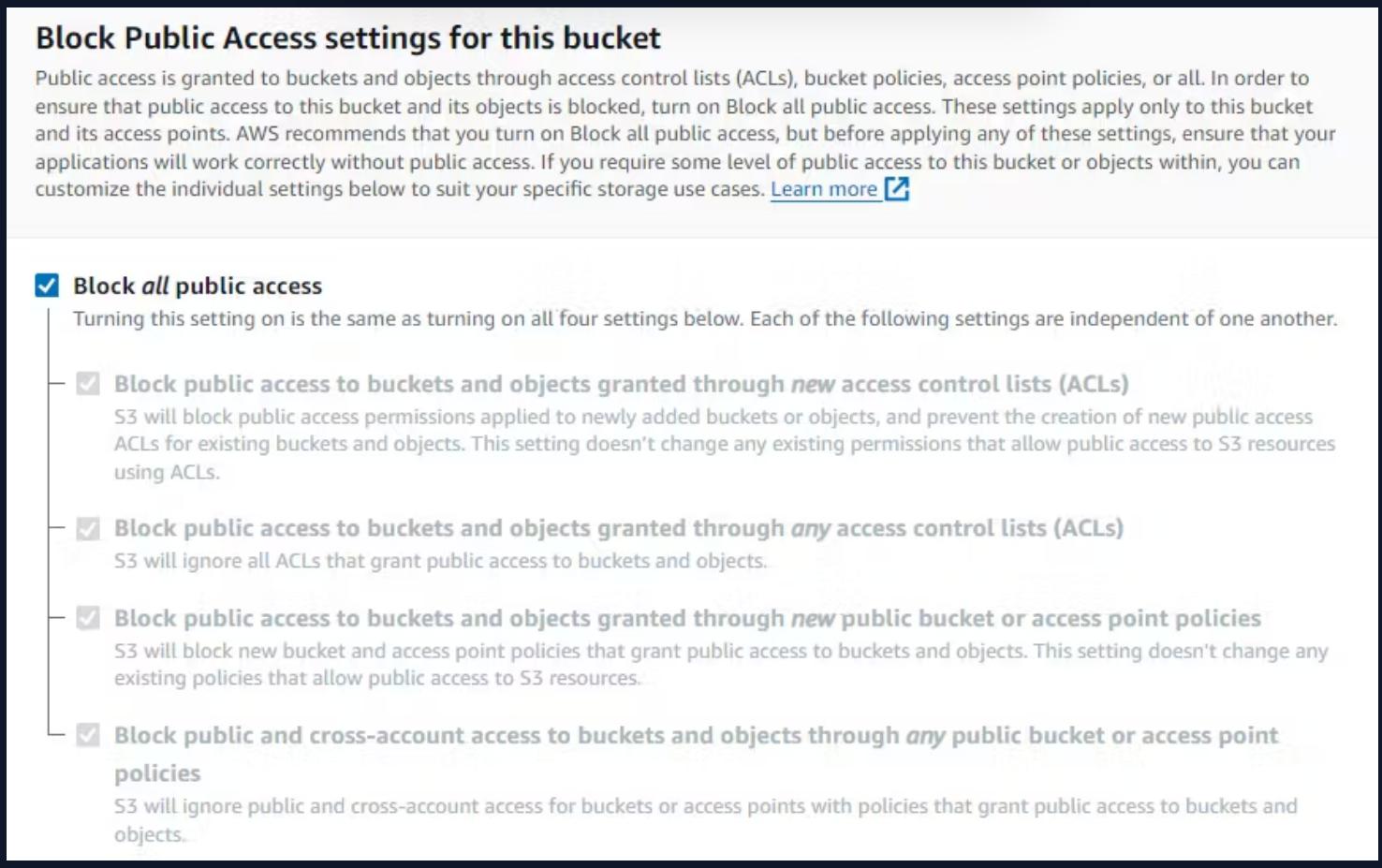
Object ownership de[redacted]with ACLs (Access Control Lists)

- ▶ **ACLs Disabled**
- ▶ **ACLs Enabled**

- **Bucket owner preferred**: The bucket owner owns and has full control over new objects that other accounts write to the bucket with the bucket-owner-full-control canned ACLs.
- **Object writer**: The AWS account that uploads an object owns the object, has full control over it and can grant other users access to it through ACLs



3. Block Pu bucket

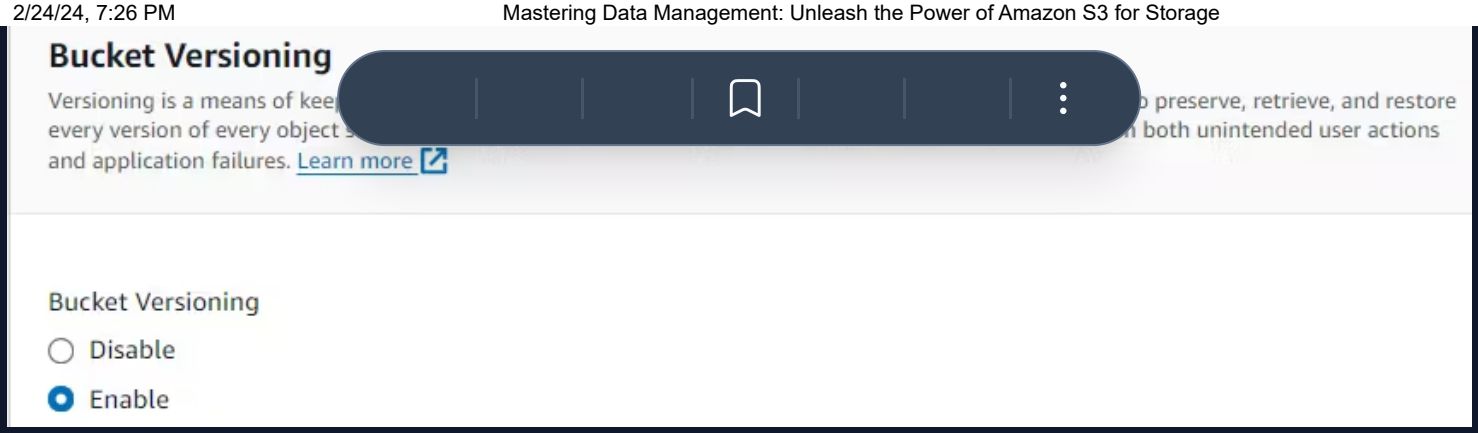


---

4. Bucket Versioning

- ▶ **Disable**
- ▶ **Enable**

- **Advantages Of Versioning**: Amazon S3 bucket versioning is a feature that allows you to keep multiple versions of an object in the same bucket. Versioning can help you recover objects from accidental deletions or overwrites.
- **Keep In mind**: Once you Enable it, you can't Disable it, but you can suspend it and, if you have three versions of an object stored, you are charged for three objects.



---

5. Tags and Default encryption

Tags

> 💡 *Tags in Amazon S3 are key-value pairs that allow you to categorize and classify your S3 buckets based on specific attributes such as project name, environment, or ownership. You can also use bucket tags to track storage costs and organize buckets.*

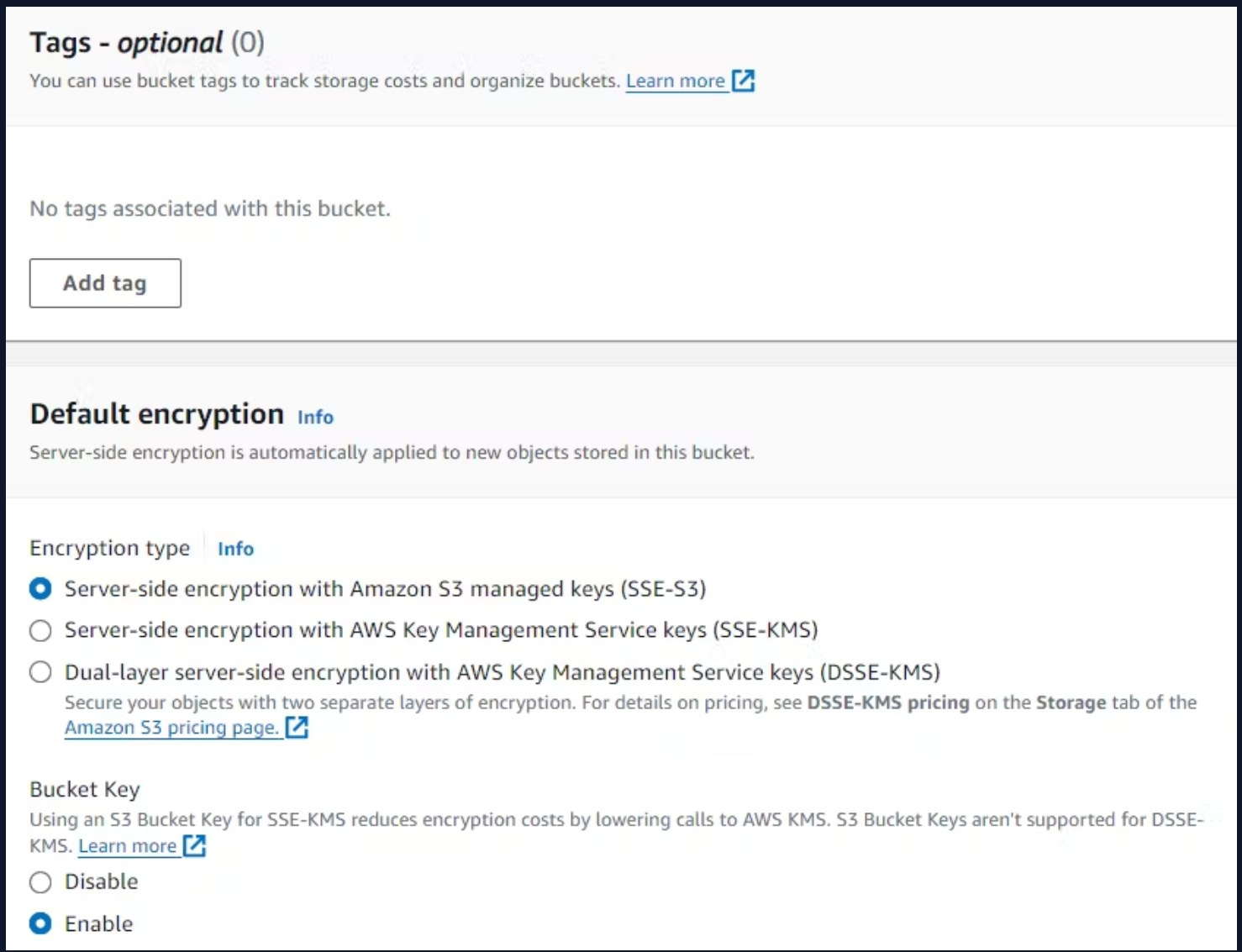
Encryption type

> *Server-side encryption with Amazon S3 managed keys (SSE-S3)*
>
> - *This encrypts data with S3-managed keys, which is the default option for S3.*
>
> *Server-side encryption with Amazon KMS Keys (SSE-KMS)*
>
> - *We can encrypt data with Amazon KMS Keys by giving permissions that we want.*
>
> *Dual-layer Server-side encryption with Amazon KMS Keys (DSSE-KMS)*

- *We can apply [illegible] layer encryption with Amazon KMS*

Bucket Key

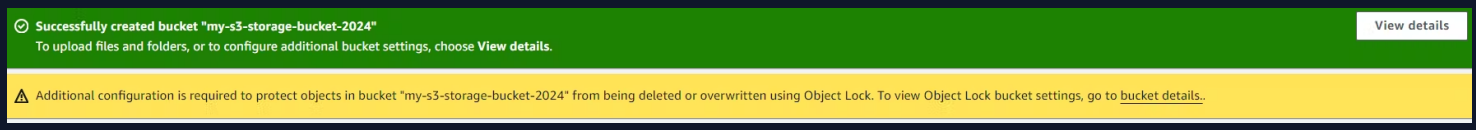
1. The S3 Bucket Key feature is like a master key for your locker. Instead of having a separate key for each document, you have one master key that can open any document in the S3 bucket.
2. Every time you use it, it creates a temporary copy of itself (a “data key”) to open a document. This means that even if someone else gets hold of this temporary key, they can only access that one document, not the whole S3 bucket.
3. S3 Bucket Keys aren't supported for DSSE-KMS



6. Advanced

Object Lock

- ▶ **Disable**
- ▶ **Enable**

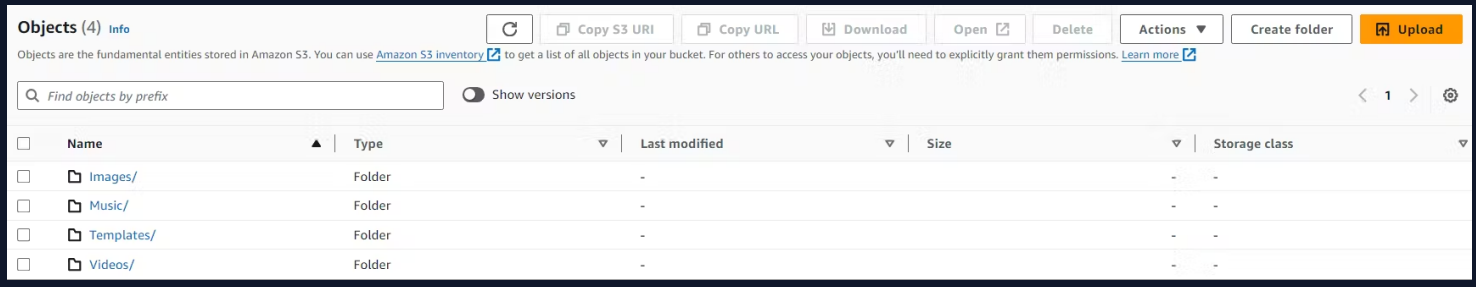


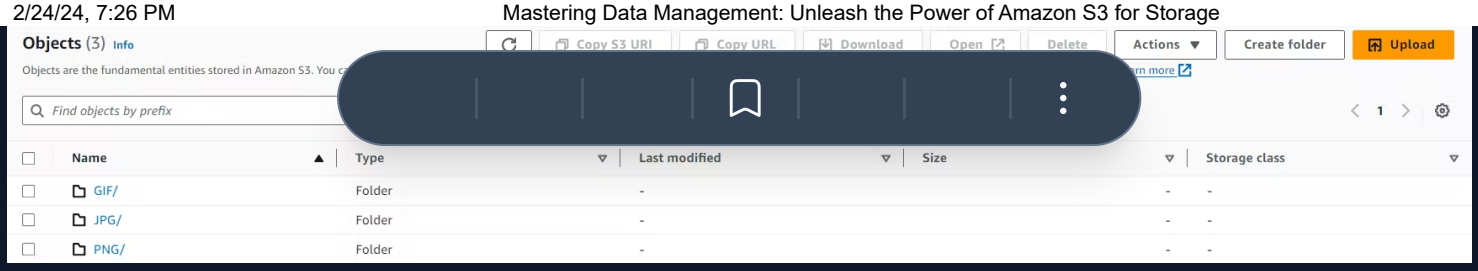
---

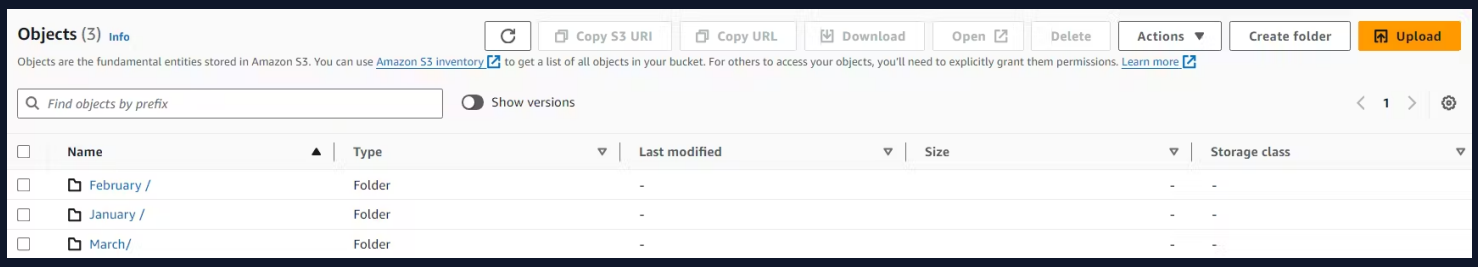
7. Organizing Data In S3 Bucket

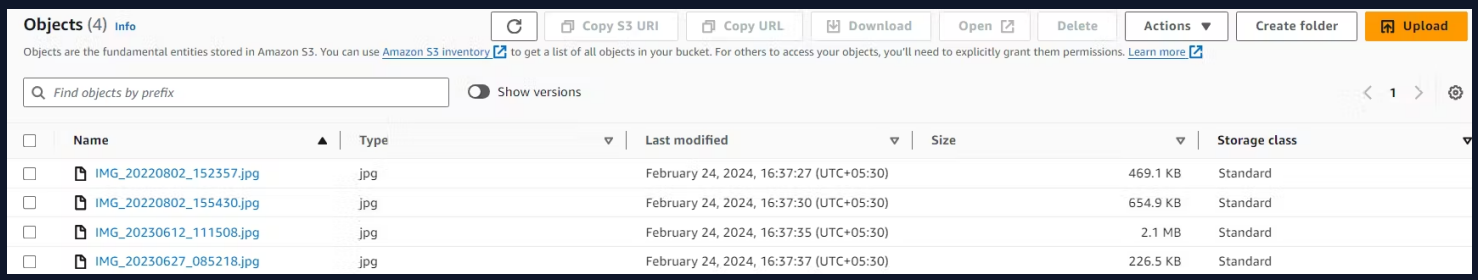
We can store data in a well-organized manner in S3 Bucket. For example, we can create folders for every different data type (image, video, music) and store sub-data in them.

Ex: We create an image folder for storing images, and then we create a sub-folder for storing different kinds of images (PNG, JPG, and GIF), and then we create a sub-folder for storing images for every month (Jan, Feb, and March), and we will store January month images in the January folder.









---

8. Configure Access Permissions For Secure Data Storage

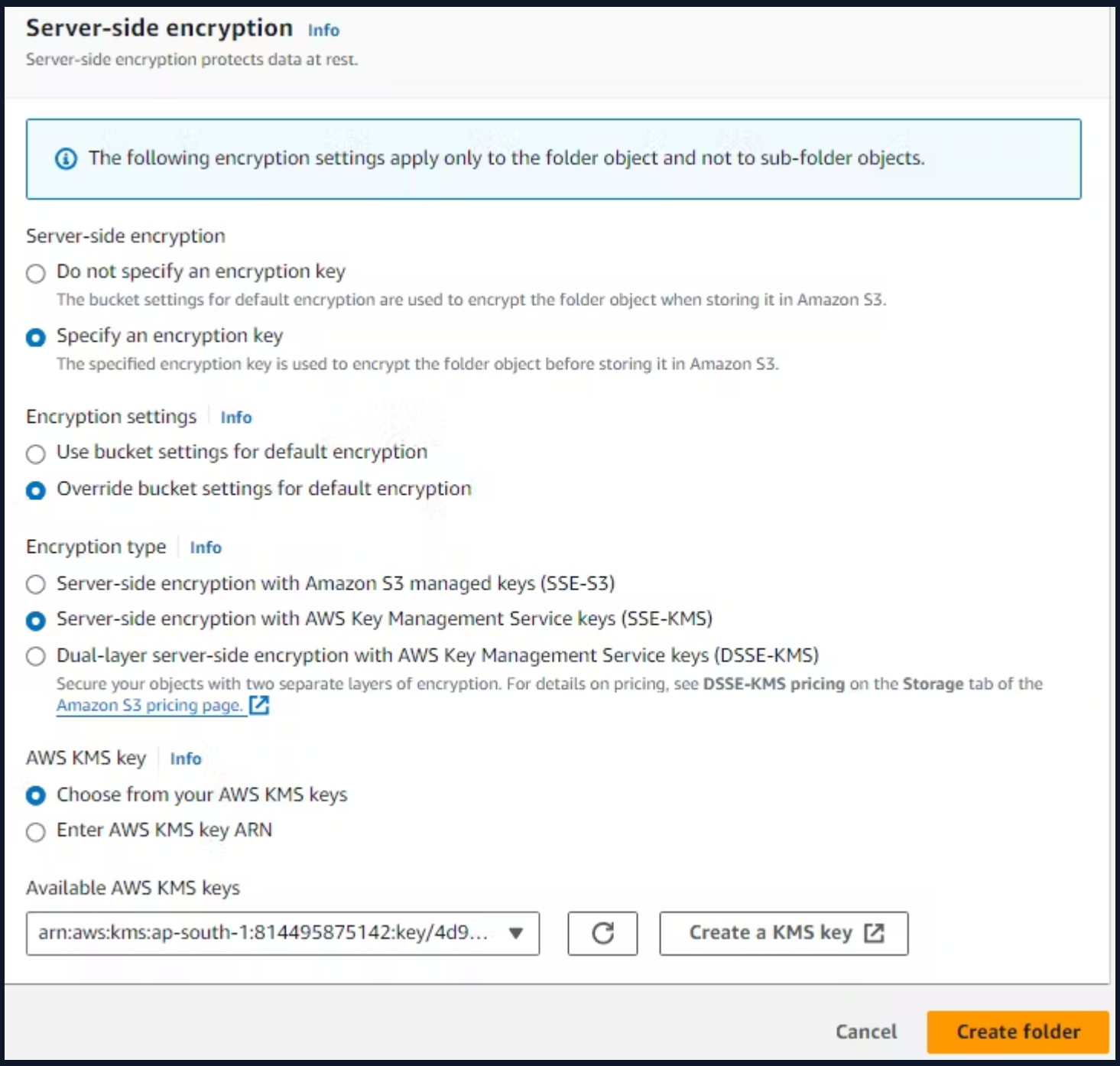
At point No 7 we have created an image folder and uploaded **.JPG** files to it. while creating folders and uploading files, we can configure permissions and secure our data storage. You just have to follow the respective points.

Prerequisite: KMS Key with permissions you want

I. Create a folder with access permissions

👉 Go into the S3 bucket → Click on **Create Folder** → Give a name to folder → In the **Server-side encryption** Check the **Specify an encryption key option** → Then check to **Override bucket settings for default encryption option** -> Then check **Server-side encryption with AWS Key Management Service**

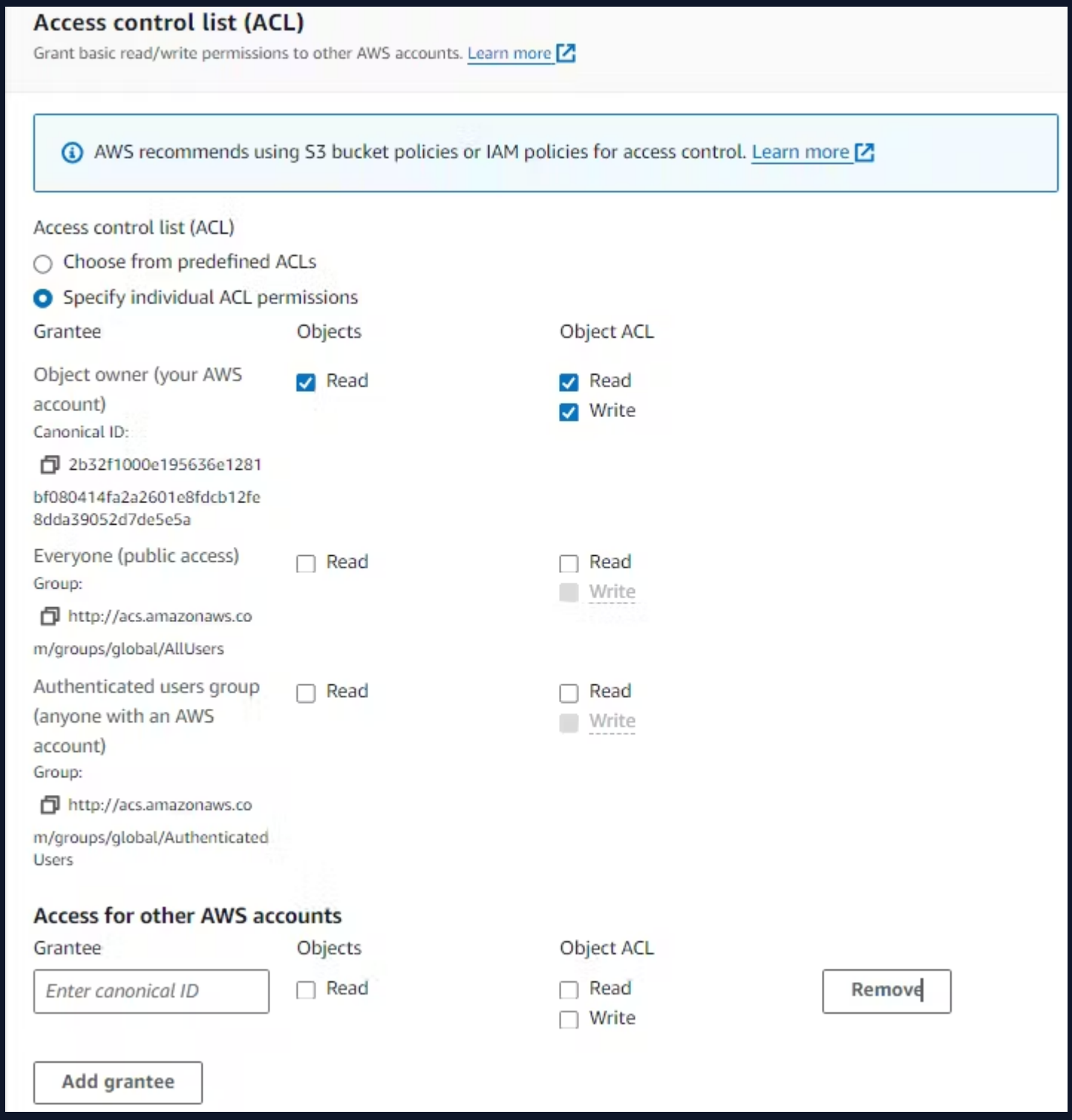
keys (SSE-KMS) → … AWS KMS keys option → Choose key … older



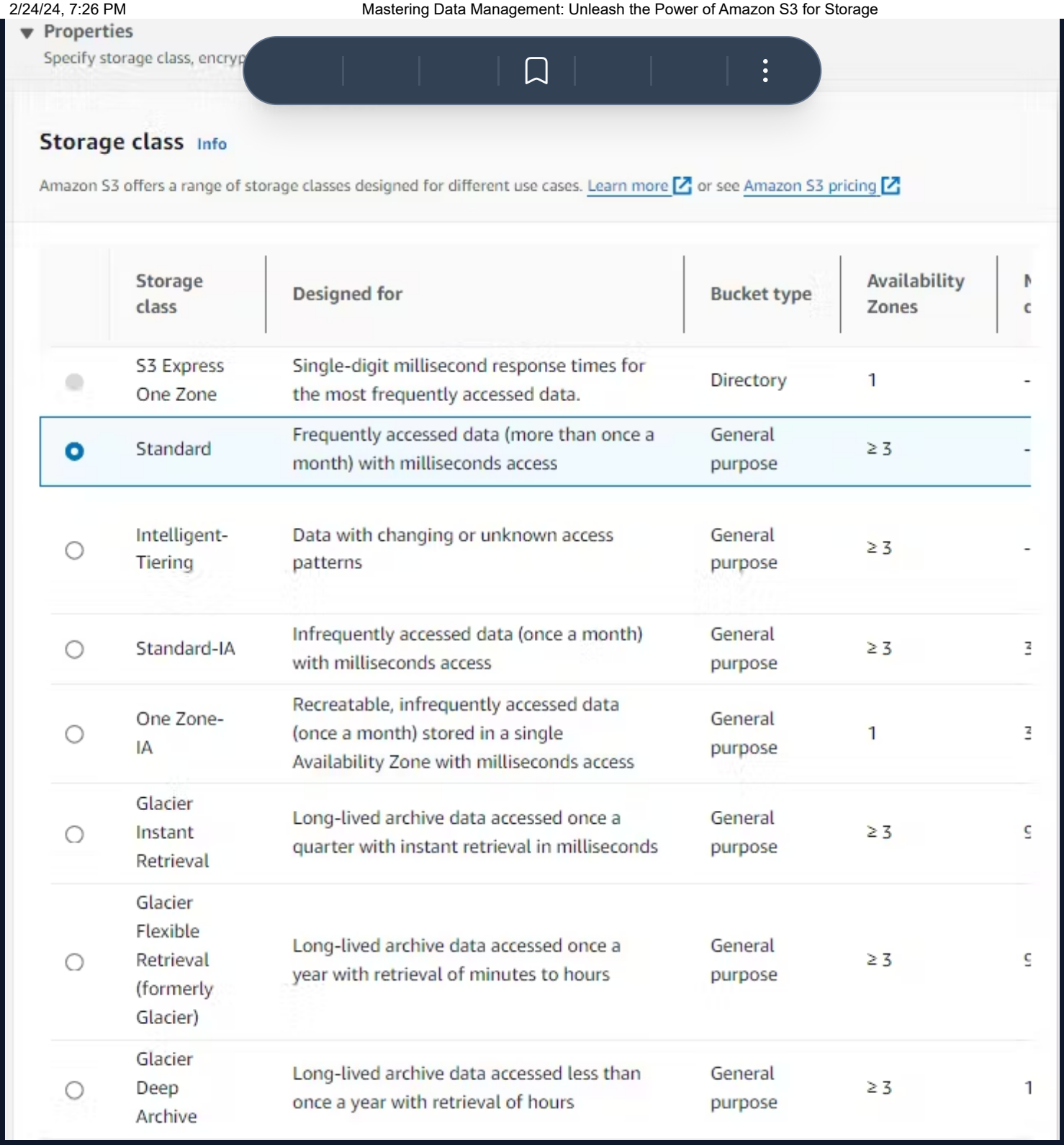
II. Upload files with access permissions

Go into the **S3 bucket** → Click on **Create Folder** → Click on **Upload file** → Choose your desired files → In the Access control list section, Select **Specify individual ACL permissions** → Now you can choose whom to give **read permissions** → We enabled **object locks** so no one can overwrite

except the ow… …counts by entering its c…



👉 In the properties section, you can choose the storage class for your object.



Properties

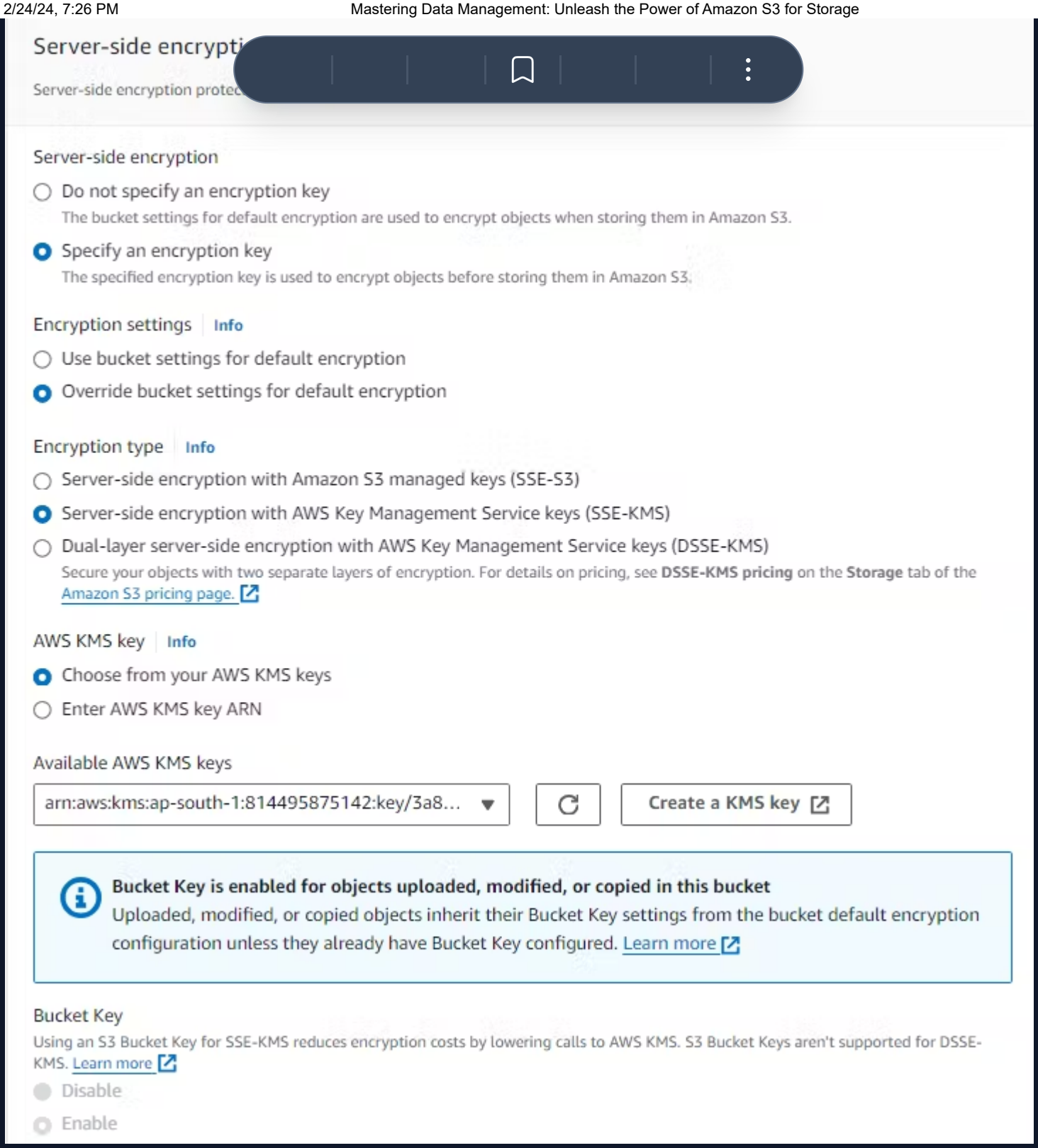
Specify storage class, encryp

Storage class Info

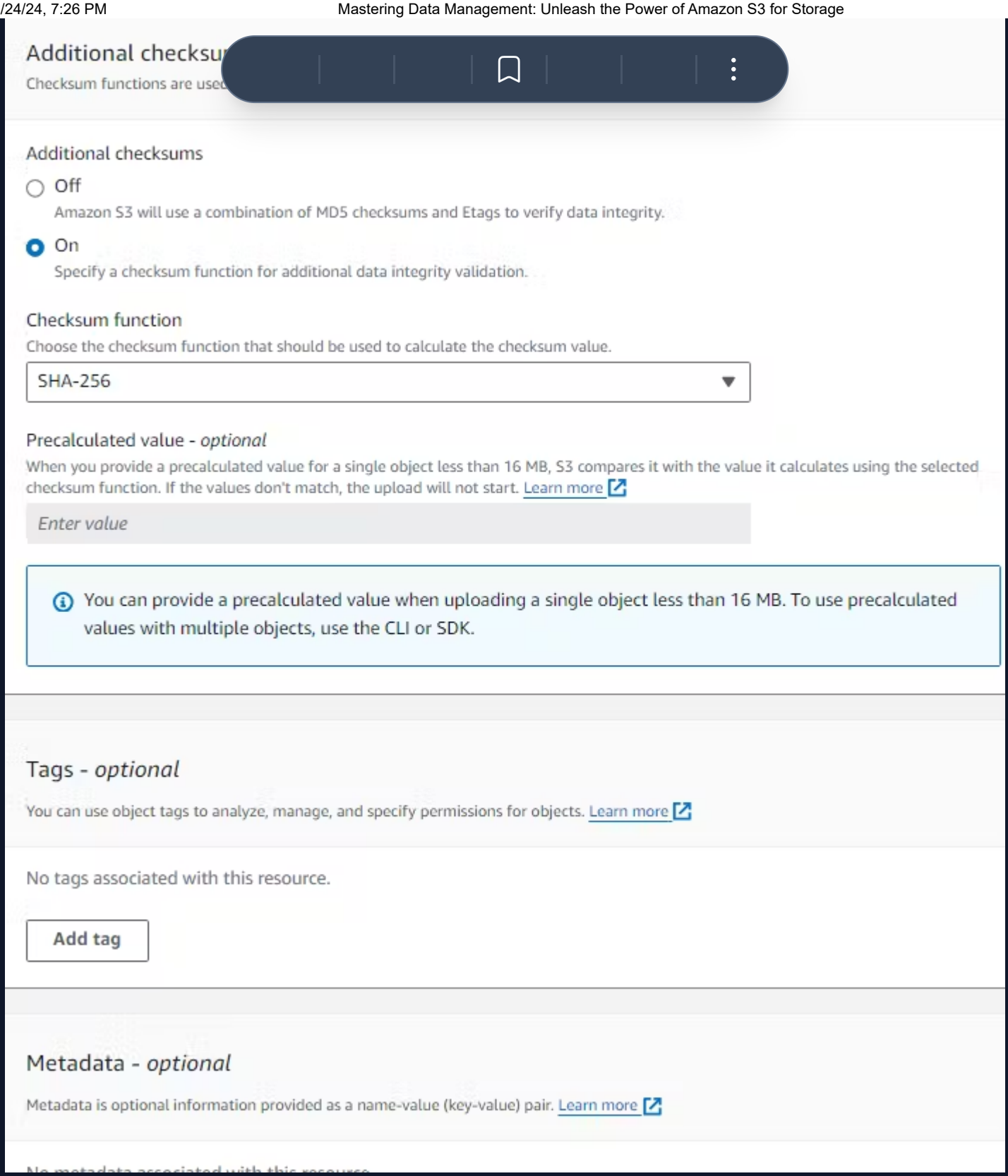
Amazon S3 offers a range of storage classes designed for different use cases. Learn more or see Amazon S3 pricing

	Storage class	Designed for	Bucket type	Availability Zones	N c
○	S3 Express One Zone	Single-digit millisecond response times for the most frequently accessed data.	Directory	1	-
◉	Standard	Frequently accessed data (more than once a month) with milliseconds access	General purpose	≥ 3	-
○	Intelligent-Tiering	Data with changing or unknown access patterns	General purpose	≥ 3	-
○	Standard-IA	Infrequently accessed data (once a month) with milliseconds access	General purpose	≥ 3	≥
○	One Zone-IA	Recreatable, infrequently accessed data (once a month) stored in a single Availability Zone with milliseconds access	General purpose	1	≥
○	Glacier Instant Retrieval	Long-lived archive data accessed once a quarter with instant retrieval in milliseconds	General purpose	≥ 3	9
○	Glacier Flexible Retrieval (formerly Glacier)	Long-lived archive data accessed once a year with retrieval of minutes to hours	General purpose	≥ 3	9
○	Glacier Deep Archive	Long-lived archive data accessed less than once a year with retrieval of hours	General purpose	≥ 3	1

💡 In the **server-side encryption section** → you can do the same as we did for folder encryption → you can enable the **bucket key** only if you enabled it while creating your bucket.



👉 You can`t specify Object Lock from here → In **Additional Checksums**, you can secure your secret data by applying checksum formats to it like **SHA-1, SHA-256**, etc. → You can give **Tags** or **Metadata** to recognize your data easily → Finally, Click on **Upload**



9. Implement Data Backup And Retrieve

Backing Up Data in S3:

1. **Create a Bucket**: First you need to create a bucket in S3 where you'll store your data.
2. **Upload Data**: Next, you can upload your data (like files, images, etc.) to this bucket.
3. **Enable Versioning**: To protect your data from accidental deletion or overwriting, you can enable versioning on your bucket. you can see
4. **Use AWS Backup**: For additional protection, you can use AWS Backup, a service that automates backup tasks.

Retrieving Backed Up Data from S3:

1. **Access the Bucket**: Open the Amazon S3 console and navigate to your bucket.
2. **Access the Object:** Navigate to the folder of the **deleted object**. Turn on **Show versions**.
3. **Find the File**: Choose the file that you want to open or download. Select the previous version of the object. Don't select the delete marker.
4. **Download the File**: Choose 'Actions', and then choose 'Open' or 'Download'.

---

10. Clean Up

- [x] Navigate to the Bucket you want to delete but first we have to delete all data from it → Click on the **Empty button** → type ***permanently delete*** *in the box and click on* ***Empty***

- [x] Now we have an empty bucket, let`s delete it now → Select the bucket → click on the **Delete button** → type the bucket name, and click on **Delete**.