**HOW TO CREATE S3 BUCKET IN AWS STEP BY STEP:**

Amazon Web Services (AWS) Simple Storage Service (S3) enables users to organize and manage data in logical containers known as “buckets”. [AWS](https://www.varonis.com/blog/aws-devops?hsLang=en) continues to improve and simplify the bucket creation and configuration process, from simply clicking “Create bucket” in the user interface to enabling creation and configuration through code, via CloudFormation. This article will walk you through how to create S3 buckets via CloudFormation, allowing you to reap the benefits of Infrastructure as Code (IaC) practices.

**What is AWS S3?**

[AWS S3](https://aws.amazon.com/s3/) is a highly-customizable, secure object storage solution. The wide variety of customizations in relation to cost, security, scalability, and durability allow users to fine-tune their architecture to meet business and [compliance requirements](https://www.varonis.com/blog/tag/privacy-compliance?hsLang=en).

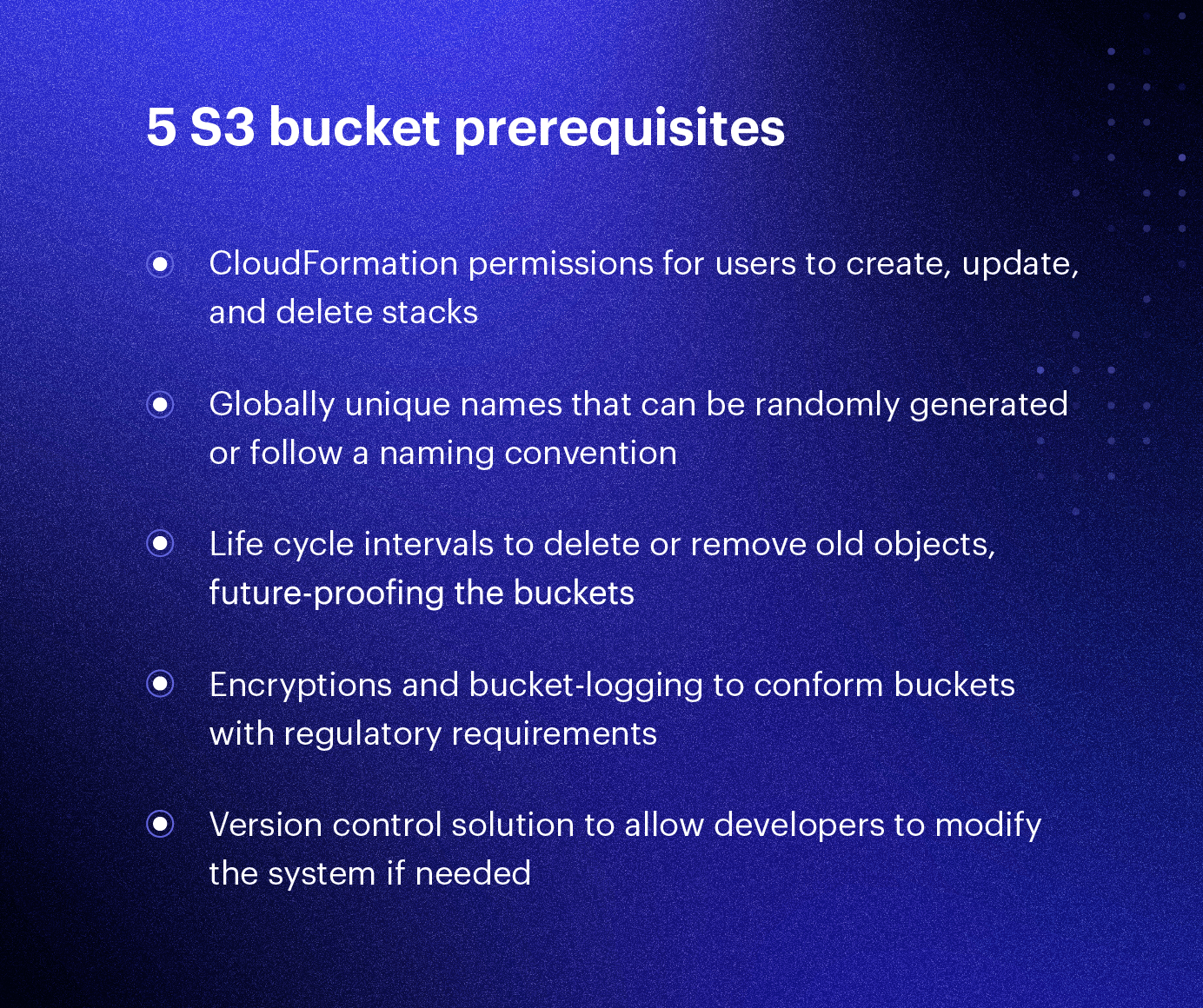
## Overview: AWS S3 bucket creation

Creating a bucket may seem like a simple enough task; the intuitive user interface makes things like configuring access control and bucket access logs, enabling encryption, and adding tags an easy process. However, these manual tasks require many specific, individual steps that can complicate the process.

The most common problem we’ve seen is accurate replication. Although a developer may manage to create and configure a single bucket successfully, the probability that they can replicate this process perfectly across multiple accounts and environments is quite low. CloudFormation is the solution to this problem.

### S3 buckets with AWS CloudFormation

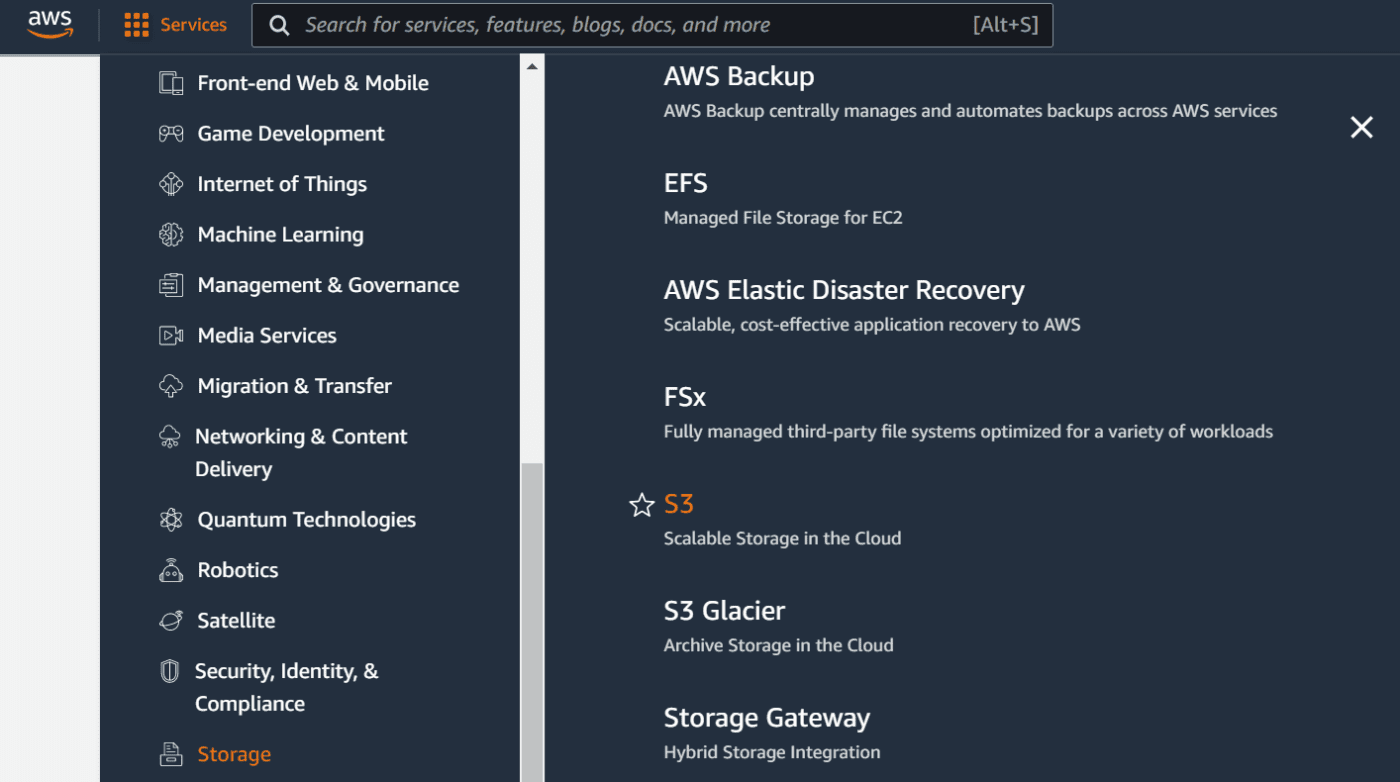
CloudFormation is used to create and configure AWS resources by defining those resources in a given IaC. Within the definition, a number of keys are used to define specific bucket attributes. This includes — but is not limited to — enabling encryption and bucket access logging. The AWS::S3::Bucket resource is used to build an Amazon S3 bucket.

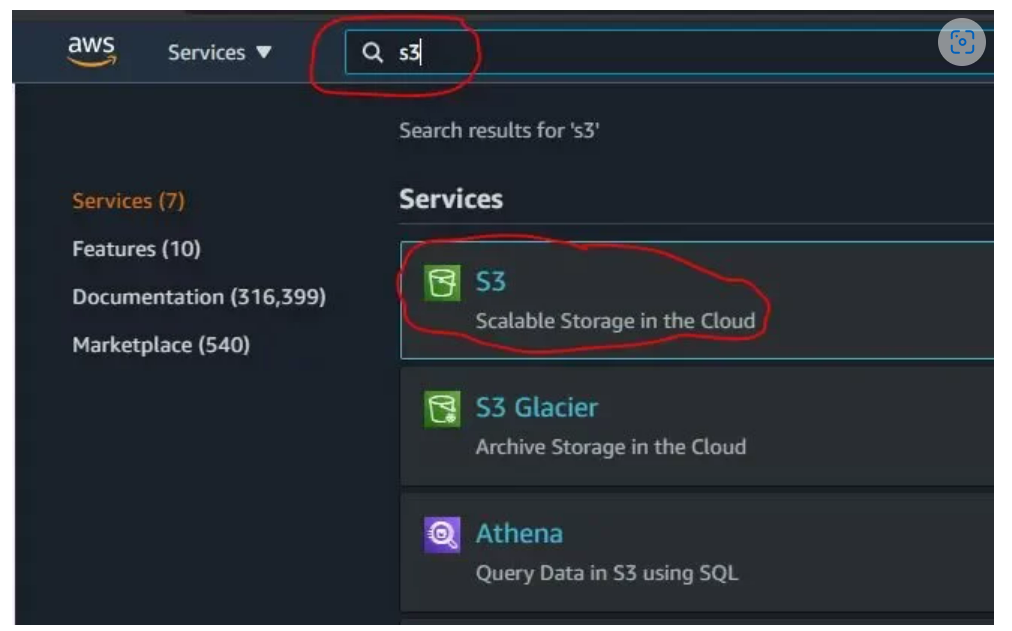


Before proceeding with bucket creation, there are a number of things to consider:

1. **CloudFormation permissions:** Does the user have permissions to create, update, and delete CloudFormation stacks? How about permissions to provision the resources listed in the CloudFormation template?
2. **Unique names:** S3 bucket names must be globally unique, making it impossible to create buckets with the same name across different accounts. This also makes it unlikely that short, simple names will be available. To avoid running into this problem, plan your names well and try to namespace them using the environment or account ID. Alternatively, you can allow CloudFormation to generate random unique identifiers instead of specifying names.
3. **Future-proofing:** If you think future analysis and reporting on a bucket is a possibility, think about how to best organize the bucket structure. For example, it's common practice to create subfolders per time period (year, month, day, etc.). Depending on how long data needs to be accessible, build life cycle rules to delete old objects or move objects between storage classes at fixed intervals.
4. **Regulatory requirements:** Business and regulatory requirements may drive configuration decisions, but regardless of requirements, it's generally a good idea to enable bucket encryption and bucket-logging anyway.
5. **Version control system:** To take advantage of IaC, resource files should be synced to a version control solution, such as [git](https://github.com/). This allows developers to quickly identify, provision, or roll back iterations of the solution.

**To Create an**S3 bucket, Login to the AWS Management Console and then click on “Services” -> “Storage” -> “S3”, once you are on the **S3 page you can see there’s a “Create Bucket” button to create a new bucket.**



You can also navigate to S3 by searching “S3” in the top search bar in your AWS console.

**Create S3 Bucket in AWS**

Once you are navigated to Create Bucket page, You should fill the following fields to create the form.

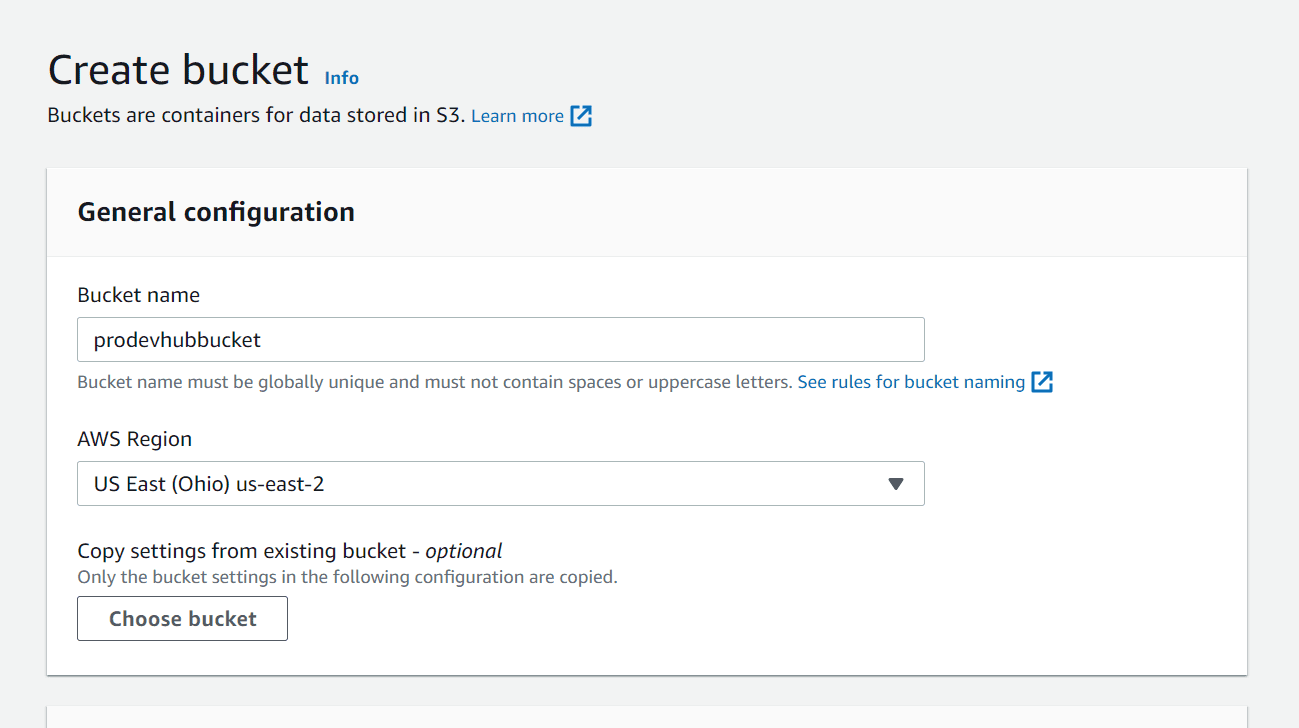
**Bucket Name:** Bucket name is a unique name globally, think of it as a username for the bucket.

**Region:** AWS has many data center regions across the world. Select any region in which you feel that your web traffic is mostly dependent to improve the performance.

**Object Ownership:**Select ACLs disabled, AWS recommends users to select ACLs disabled. which enforces the following terms

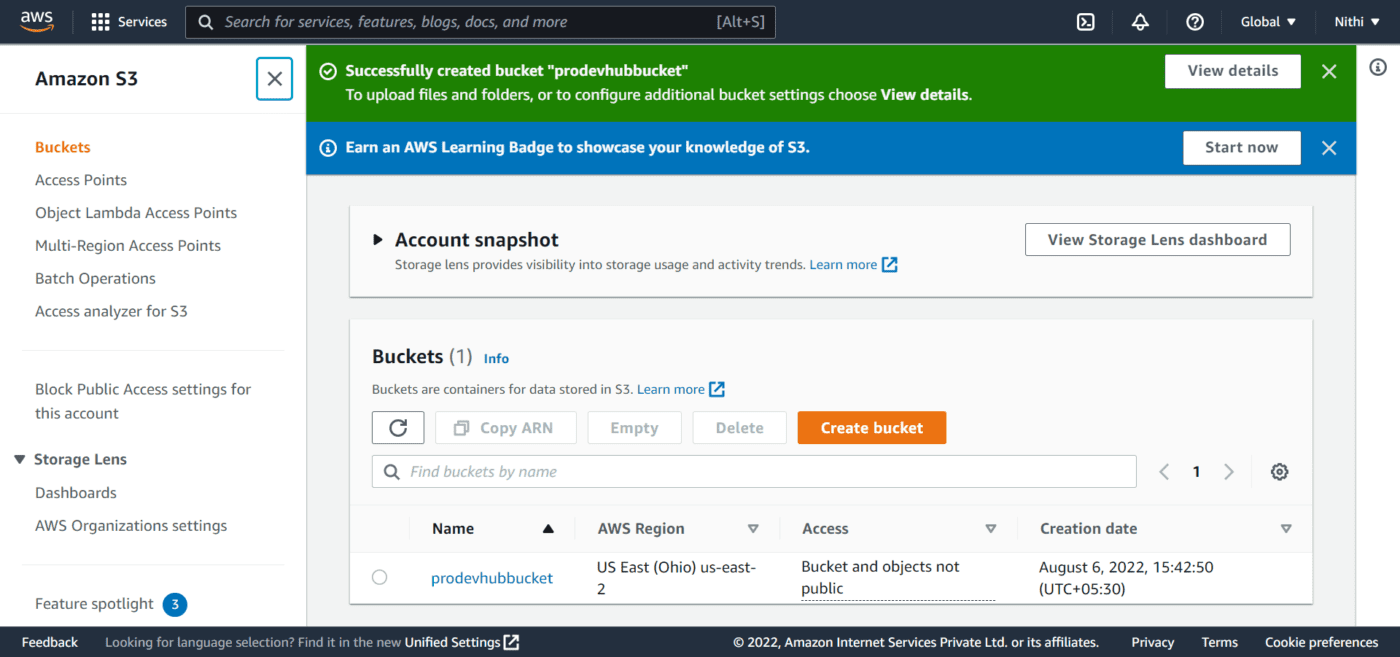
*ACLS disabled: “All objects in this bucket are owned by this account. Access to this bucket and its objects is specified using only policies.”*

Leave the rest of the fields as recommended by AWS. Click the **Create Bucket** button at the end of the form to create the bucket.



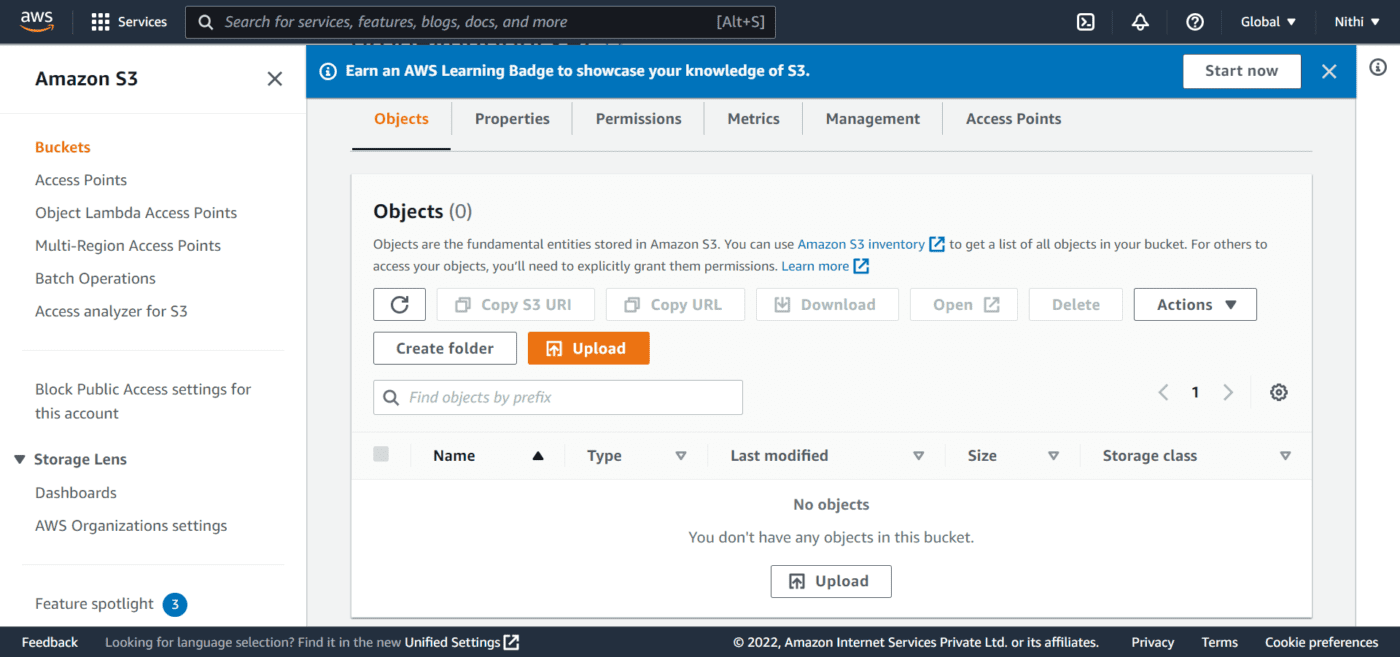
## List of Buckets in S3

Once your bucket is successfully created, you can see the list of buckets under the buckets menu by clicking on the menu in sidebar “Buckets”

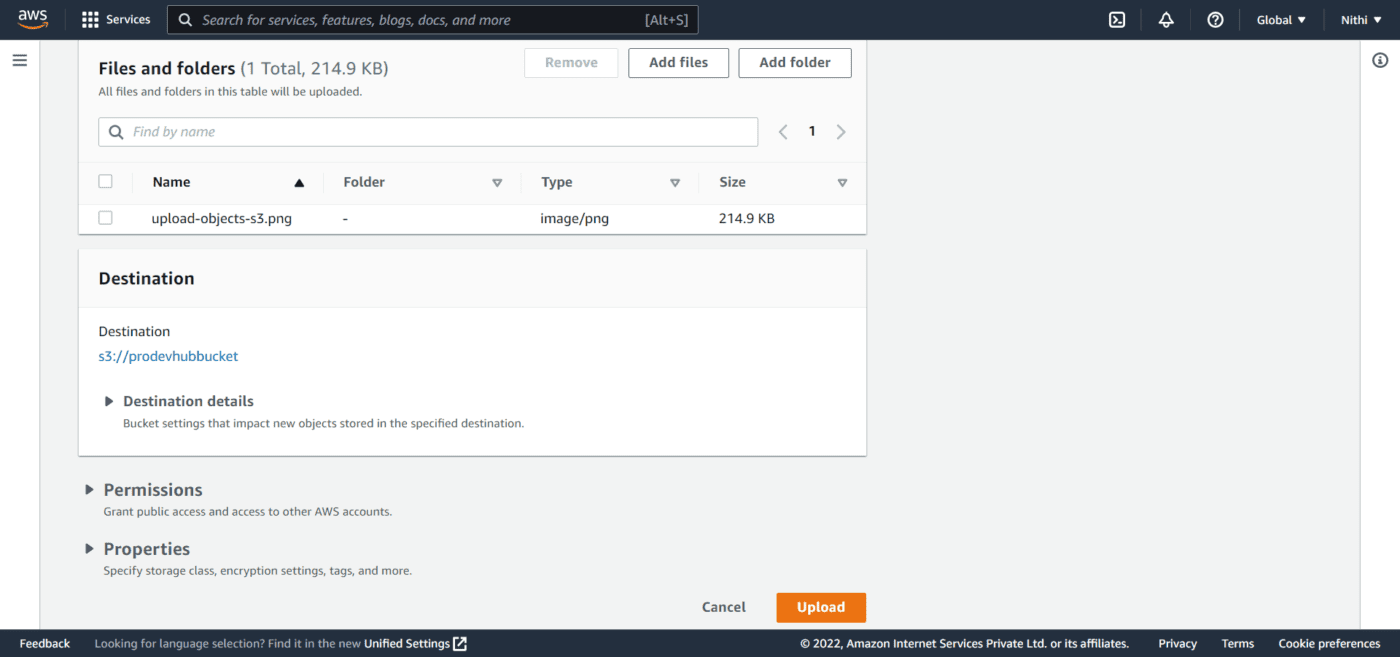


## Upload Objects/Files to your newly Created S3 Bucket

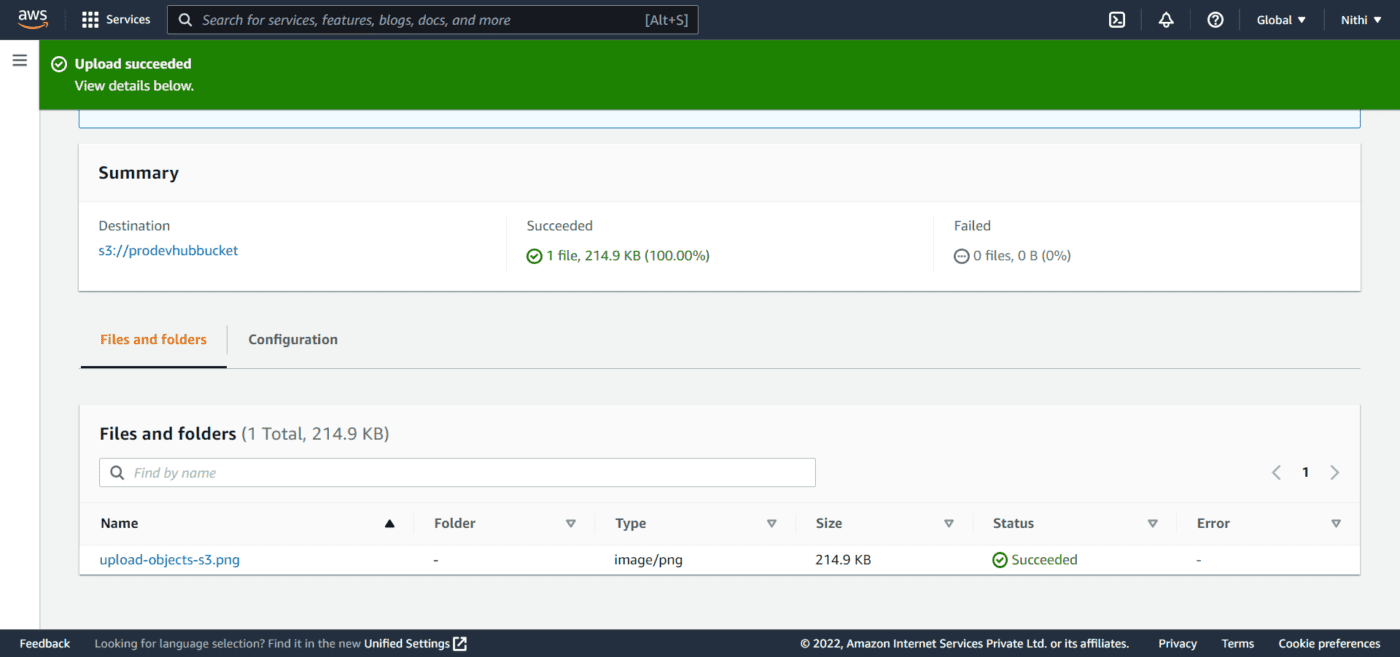
Click on the bucket name and then you can upload a file using the upload button at the top of the page.



Then, you will be redirected to the upload form where you can add files/folders by clicking the add files/add folder button.



Click the upload button once you have selected a file. Then you will be navigated to the page where you can see the list of files under the page.



Click the file to see the file details.

## Allow Public Access to Bucket

By default, all the buckets are in private access. You can give the public access to the bucket/files by clicking on the **Permissions -> Edit -> Uncheck block public -> Save Changes.**

**To make files accessible to the public, select files from the uploaded list, and then in the “Actions” dropdown select “Make Public”. Now your files are accessible to the public.**

## Conclusion

In this step-by-step S3 Concept, We have learned to create an S3 bucket in AWS. We went through the creation of the S3 Bucket using the AWS console in a step-by-step manner.

We also uploaded a file and made the object publicly accessible.