Static Website using AWS S3

Project Description:-

Amazon S3 (*Simple Storage Service*) is a highly scalable and reliable object storage service offered by AWS. It allows you to store and retrieve any amount of data from anywhere on the web. In this project, we will leverage the capabilities of AWS S3 to host a static website. Unlike dynamic websites, which require server-side processing and databases, static websites consist of HTML, CSS, JavaScript, and other client-side files that can be directly served to visitors

The main objectives of this project are as follows:

1. Upload the website files to an S3 bucket: We will create a new S3 bucket and upload the static website files to it. These files can include HTML pages, CSS stylesheets, JavaScript scripts, images, and other assets.
2. Configure the S3 bucket as a static website: S3 provides a feature to turn a bucket into a static website by enabling website hosting. Once configured, the bucket will act as a web server, serving the static content directly to visitors.
3. Set up appropriate permissions: We'll define the necessary permissions to ensure public access to the static website. This will allow anyone with the website URL to view the content.
4. Ensure cost-effectiveness and scalability: AWS S3 is known for its cost-effectiveness and scalability, making it an ideal choice for hosting static websites.

Now that we have a clear understanding of the project's objectives, let's proceed with the step-by-step resolution of the projec

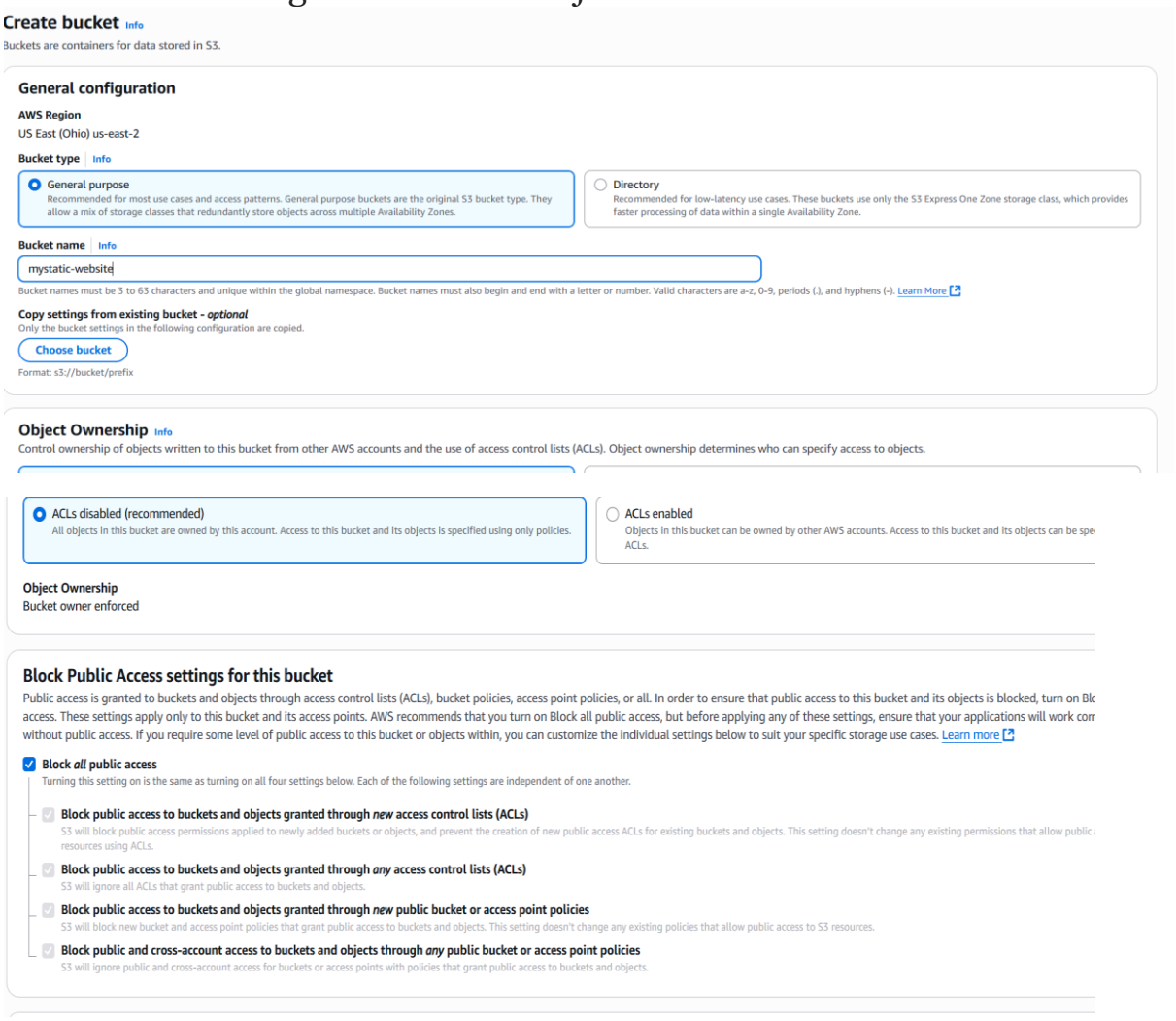
Hands-on Project: Hosting a Static Website on AWS S3

Step 1: Create an S3 Bucket

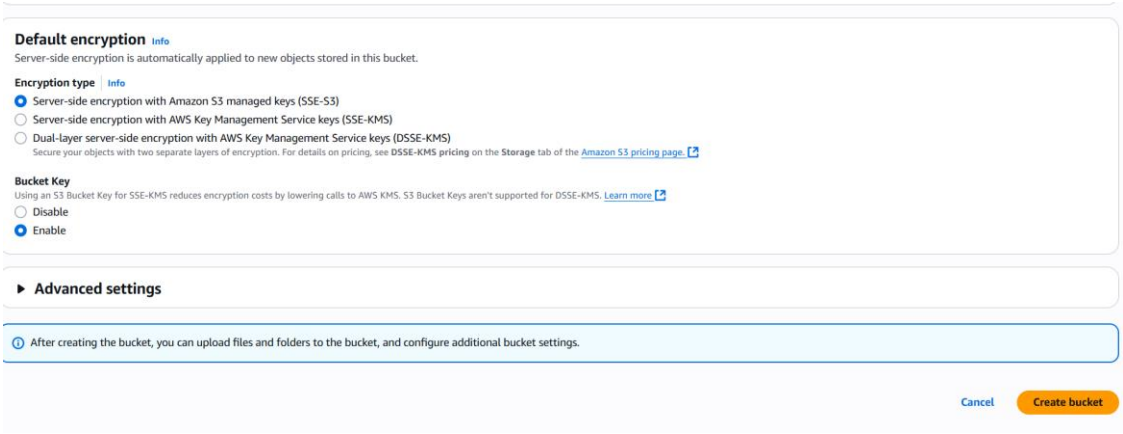
1. Log in to your AWS Management Console.
2. Navigate to the S3 service by clicking on “Services” in the top-left corner and selecting “S3” under the “Storage” section.
3. Click on the “Create Bucket” button.
4. Choose a unique bucket name that reflects your website (e.g., my-static-website).

5. Select the appropriate region for your website’s audience (choose the region closest to your target audience for better performance).

6. Leave other settings as default or adjust them as needed.

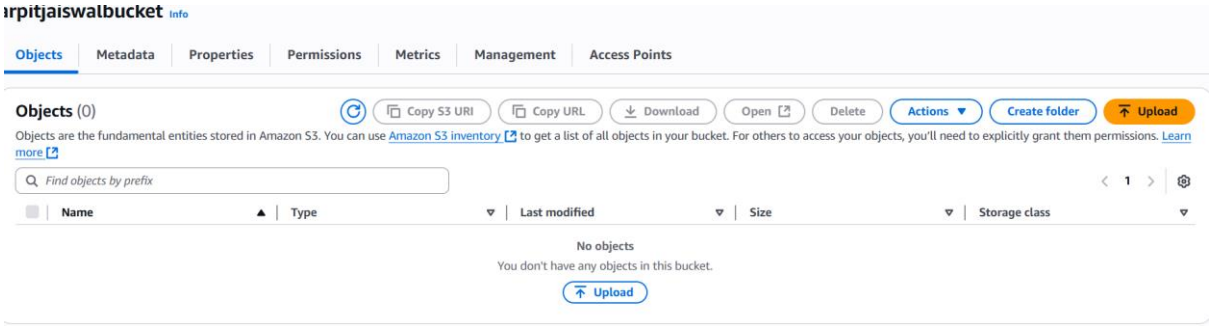


7. Click on “Create Bucket” to create your S3 bucket.

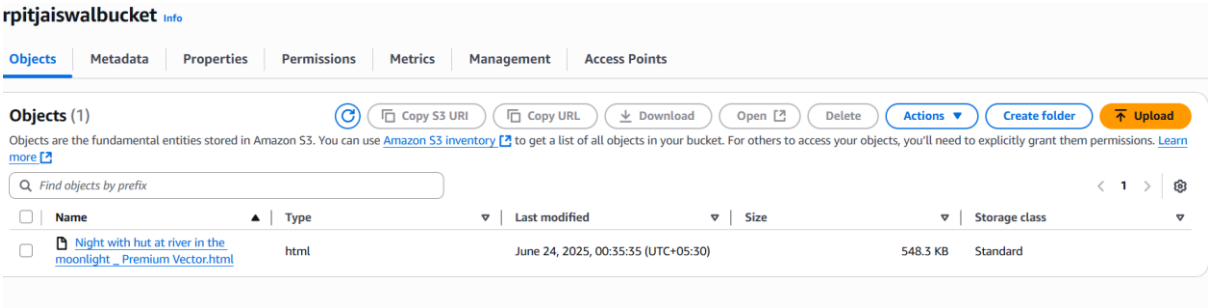


Step 2: Upload Website Files

1. Once the S3 bucket is created, click on its name to open the bucket details.



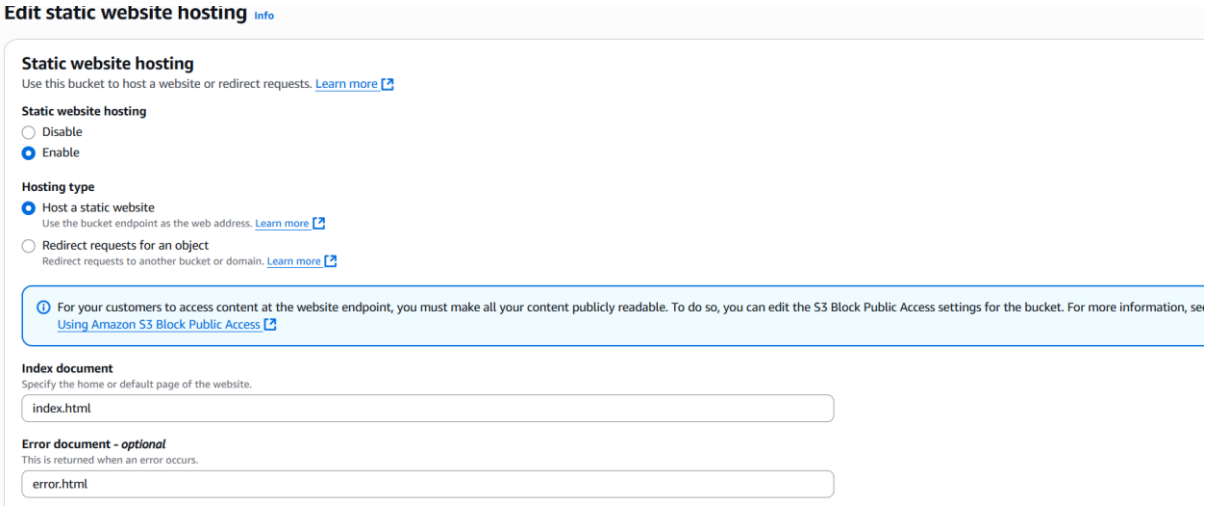
2. Click on the “Upload” button to upload your website files.
3. Drag and drop or choose the files from your local system.
4. Make sure to include your HTML, CSS, JavaScript, images, and any other assets required for the website.



5. Step 3: Configure the S3 Bucket for Website Hosting

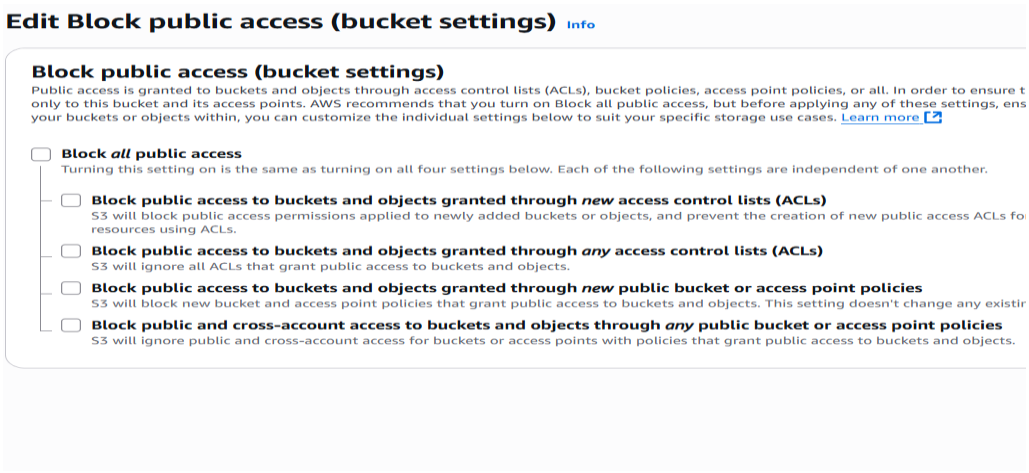
6. In the bucket properties, navigate to the “Static website hosting” section.

7. Select the “Use this bucket to host a website” option.
8. For the “Index document,” enter the filename of your default homepage (usually “index.html”).
9. For the “Error document,” enter the filename of your custom error page (optional, usually “error.html” or “404.html”).
10. Click on “Save changes” to enable website hosting for the bucket.



Step 4: Edit S3 Block Public Access settings and Set Bucket Policy for Public Access

1. In the bucket properties, navigate to the “Permissions” tab.
2. Under Block public access (bucket settings), choose Edit.
3. Clear Block *all* public access, and choose Save changes.



4. Under Bucket Policy, choose Edit.

5. To grant public read access to your website, copy the following bucket policy, and paste it into the Bucket policy editor.

6. {

```
"Version": "2012-10-17",
"Statement": [
  {
    "Sid": "PublicReadGetObject",
    "Effect": "Allow",
    "Principal": "*",
    "Action": [
      "s3:GetObject"
    ],
    "Resource": [
      "arn:aws:s3:::Bucket-Name/*"
    ]
  }
]
}
```

Click on “Save” to apply the bucket policy, allowing public read access to

objects in the bucket

Policy

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "PublicReadGetObject",
            "Effect": "Allow",
            "Principal": "*",
            "Action": [
                "s3:GetObject"
            ],
            "Resource": [
                "arn:aws:s3:::arpitjaiswalbucket/*"
            ]
        }
    ]
}
```

Step 5: Test and Verify

Test your website by accessing the S3 bucket endpoint (e.g., my-static-website.s3.amazonaws.com) in a web browser

You should be able to see your static website live and accessible to the public.

ıptop Wallpaper Vectors