**Little Bit Advance Labs**

**Part 1: EC2 with ELB and ASG**

**Objective**: Learn how to create a scalable and highly available web application environment using Amazon EC2 instances, ELB, and ASG.

**Approach**:

1. **Launch EC2 Instances**: Start by launching two or more EC2 instances. These instances will run a simple web application (e.g., a "Hello World" page or any basic web service).
2. **Configure Load Balancer**: Set up an Elastic Load Balancer (ELB) to distribute incoming web traffic across your EC2 instances. This step ensures high availability and fault tolerance.
3. **Set Up Auto Scaling Group (ASG)**: Create an ASG that uses the launched EC2 instances. Configure ASG policies to automatically scale the number of instances up or down based on criteria like CPU usage or network traffic.
4. **Test Your Setup**: Simulate traffic to test the scaling policies and the load balancer. Observe how ASG adds or removes instances and how ELB distributes traffic.
5. **Verify Website Functionality**: Ensure that the website hosted on EC2 instances remains accessible and functional during scaling operations.

**Goal**: By the end of this lab, students will have a hands-on understanding of setting up a load-balanced and auto-scaled web application using AWS services.

**Part 2: Hosting a Static Portfolio Website on S3**

**Objective**: Learn to host a static website (such as a personal portfolio) on Amazon S3.

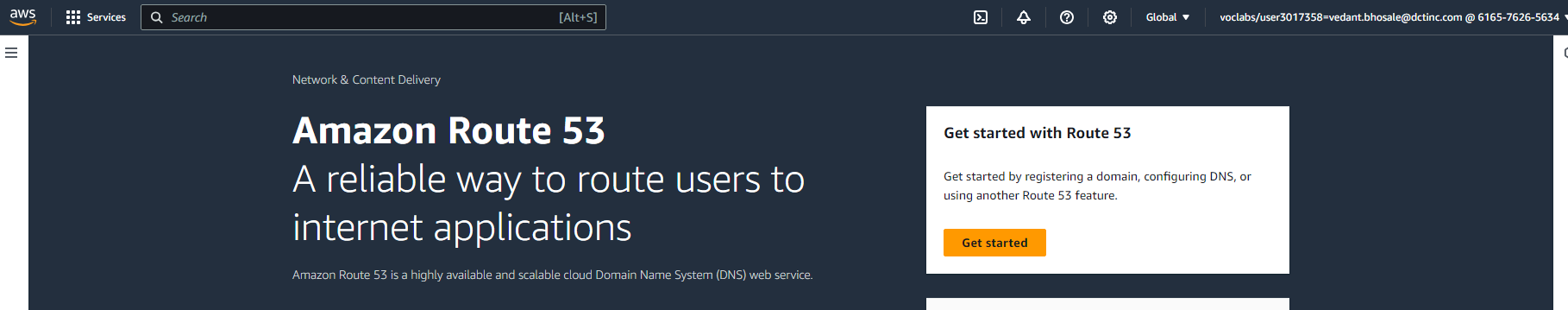
**Approach**:

1. **Create an S3 Bucket**: Start by creating a new S3 bucket. Configure the bucket for website hosting, which includes setting permissions to make the content publicly accessible.
2. **Upload Website Files**: Upload the static files of your portfolio website (HTML, CSS, JavaScript, images) to the S3 bucket.
3. **Configure DNS**: Use Amazon Route 53 or another DNS service to point a domain name to the S3 bucket. This makes the website accessible via a user-friendly URL.
4. **Enable Additional Features** (Optional): Implement features like HTTPS for secure access and CloudFront for content delivery optimization.

**Goal**: Students will understand how to use S3 for hosting static websites, manage bucket permissions, and integrate with other AWS services for a complete web hosting solution.

Steps:

1. To configure DNS we use Amazon Route 53
   * Click on get started in Amazon Route 53



1. Click on Register a domain to register a URL

A screenshot of a computer

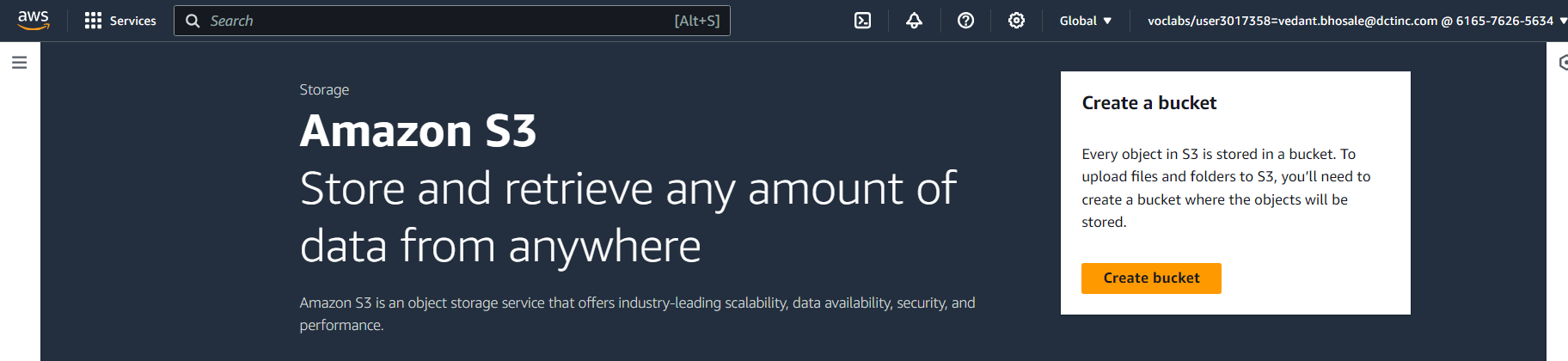
Description automatically generated

1. To register a domain enter your domain name in search for domain
   * As we don’t have access to create a domain name we will skip this part.

A screenshot of a computer

Description automatically generated

1. To static hosting of website from the s3 bucket
   * Click on create bucket



1. Give bucket name (same as DNS if you have created the DNS)
   * Select AWS region nearest to your location for minimal latency
   * Untick block all public access to access the bucket from the internet



1. Here we see the bucket created

A screenshot of a computer

Description automatically generated

1. To upload files for static website click on upload

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

1. Here we see the index file uploaded

A screenshot of a computer

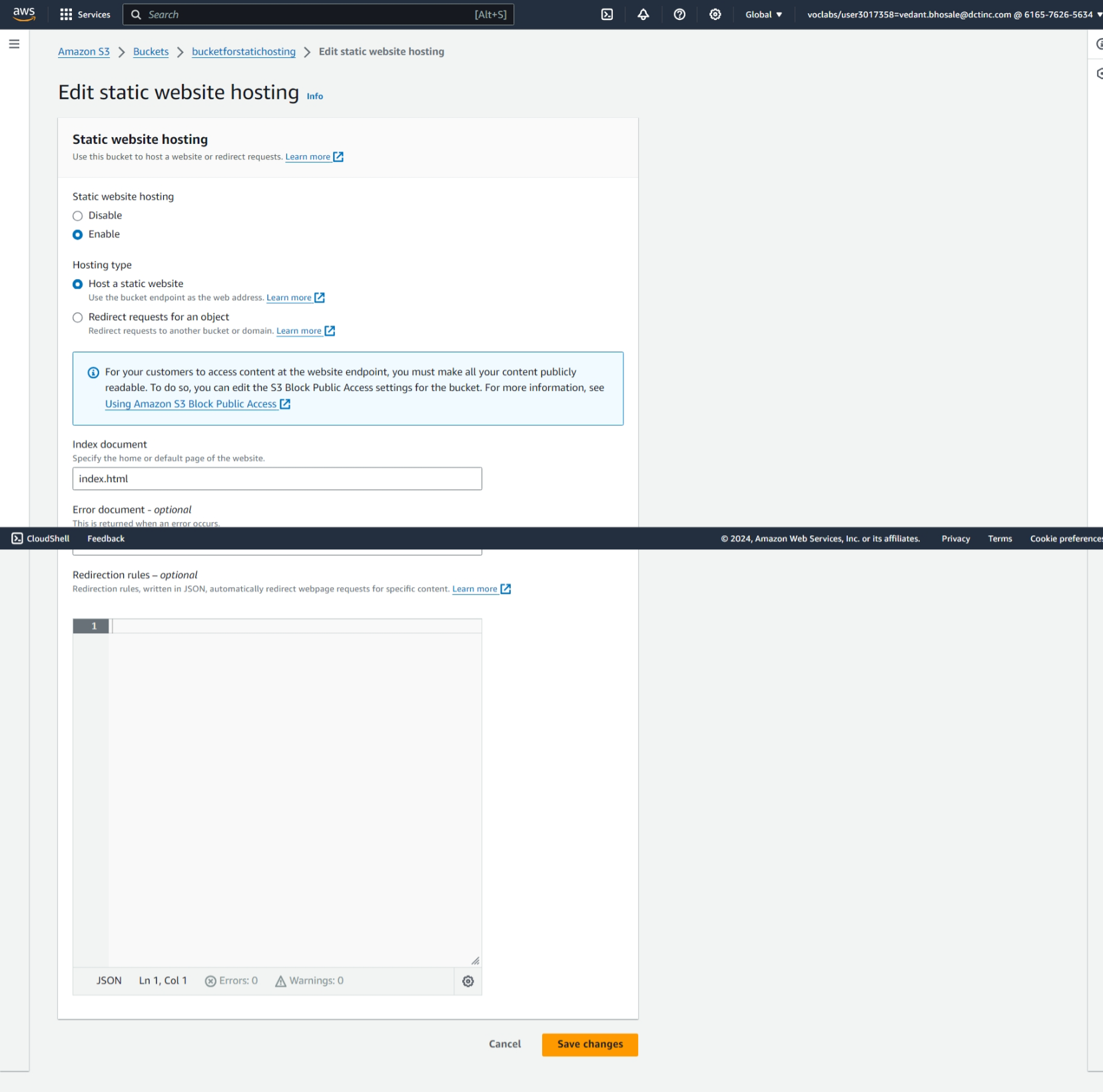
Description automatically generated

1. Here we can see that the static website hosting is disabled
   * To enable it click on edit static web hosting

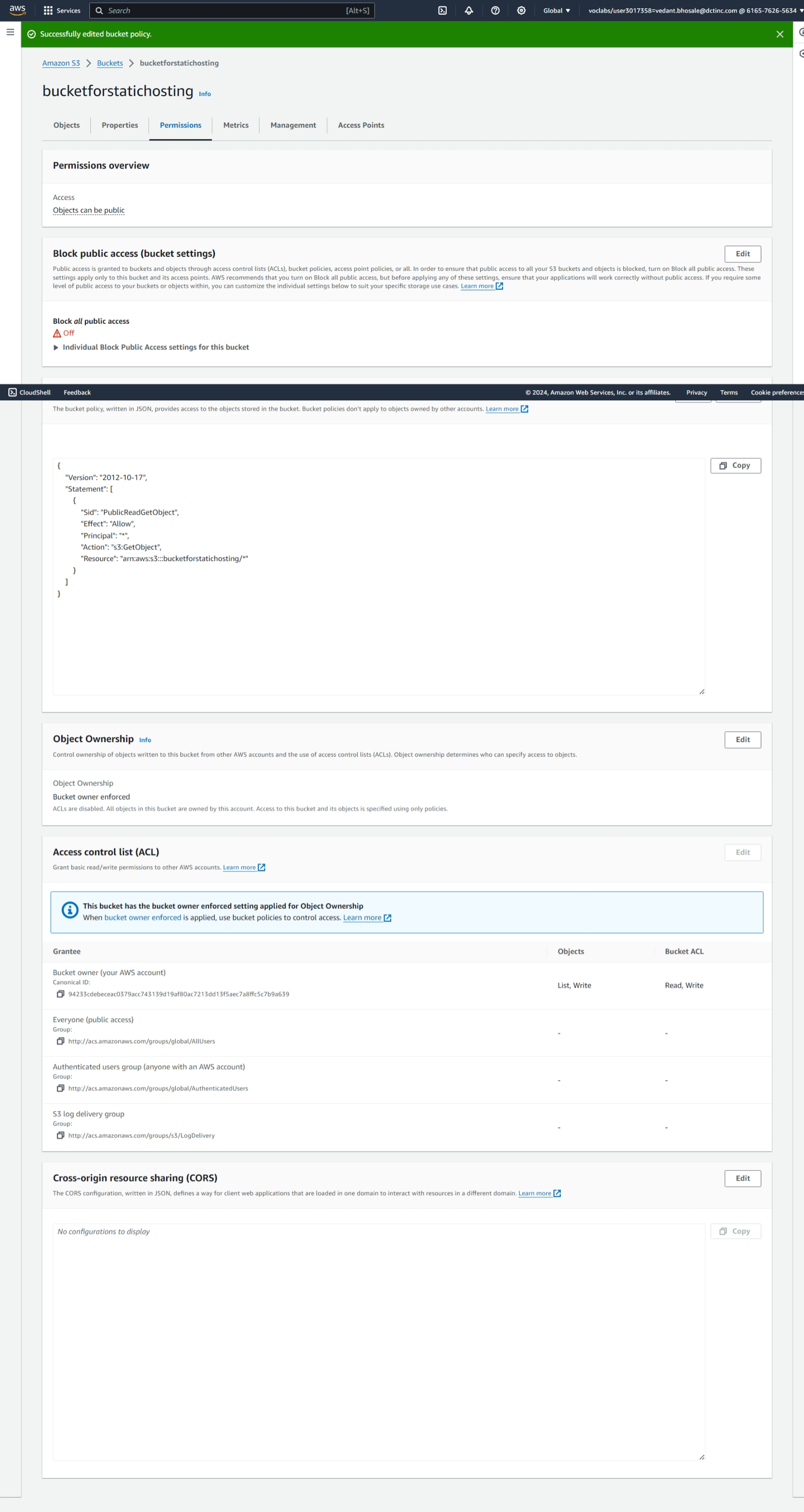
A screenshot of a computer

Description automatically generated

1. Here we enable static web hosting give a bucket policy



1. Here we can see the bucket policy



1. here we can see the website hosted from amazon s3

