📄 AWS S3 Bucket Attack & Defense Lab Report

Created By:

Rituraj Singh

---

1. Title

AWS S3 Public Bucket Misconfiguration & Remediation using Static Website Hosting

---

2. Objective

The objective of this lab is to understand how misconfigured AWS S3 buckets can lead to **public data exposure**, and how to identify, exploit, and fix such vulnerabilities using AWS security controls.

---

3. Overview

In this lab, we performed a practical demonstration of:

- Creating an S3 bucket with public access enabled
- Uploading a file and exposing it using Static Website Hosting
- Attempting public access via browser
- Encountering an **AccessDenied (403 Forbidden)** error
- Fixing the issue by applying a **Bucket Policy**
- Successfully accessing the file publicly
- Securing the bucket again using **Block Public Access**

---

4. Prerequisites

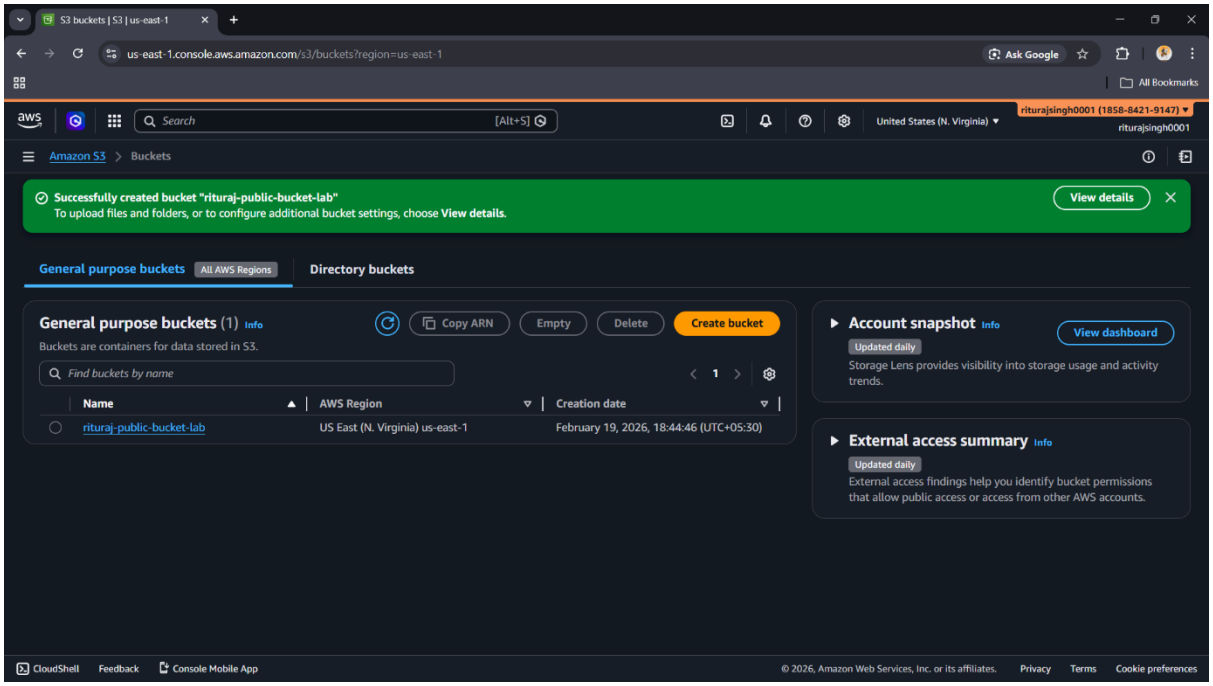
- AWS Account
- Basic knowledge of Amazon S3
- Internet browser (Incognito mode for testing)

---

5. Lab Implementation

Step 1 – Create Test Bucket

- Open AWS Console → Amazon S3
- Click **Create Bucket**
- Bucket name: lab-public-bucket-<random>
- Region: us-east-1
- Disabled **Block All Public Access**
- Created the bucket



✅ This configuration makes the bucket **potentially vulnerable** to public exposure.

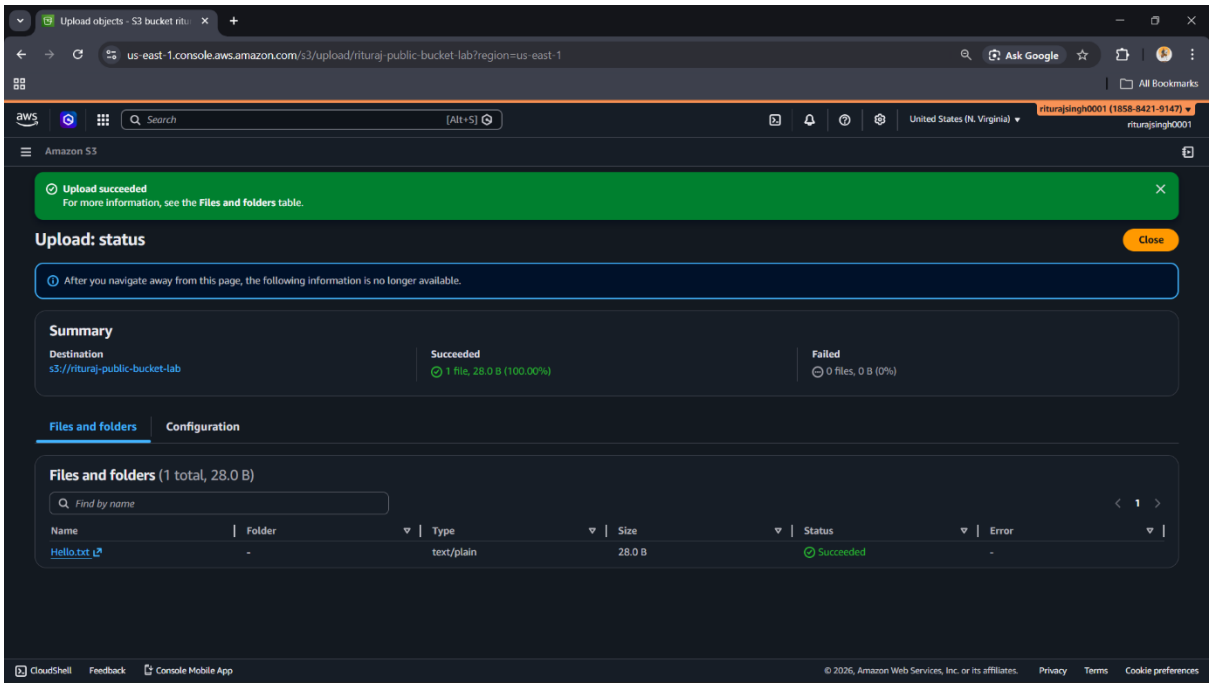
Step 2 – Upload a File

- Open bucket → **Objects tab**
- Click **Upload**
- Uploaded file: hello.txt

File Content:

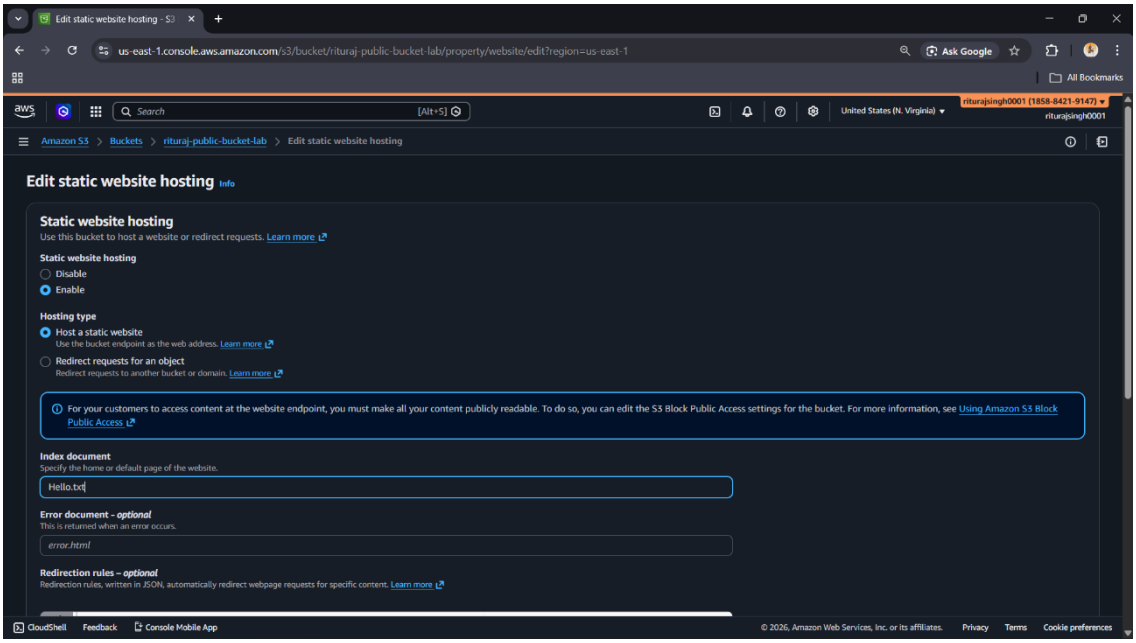
Hello,

This is demo lab

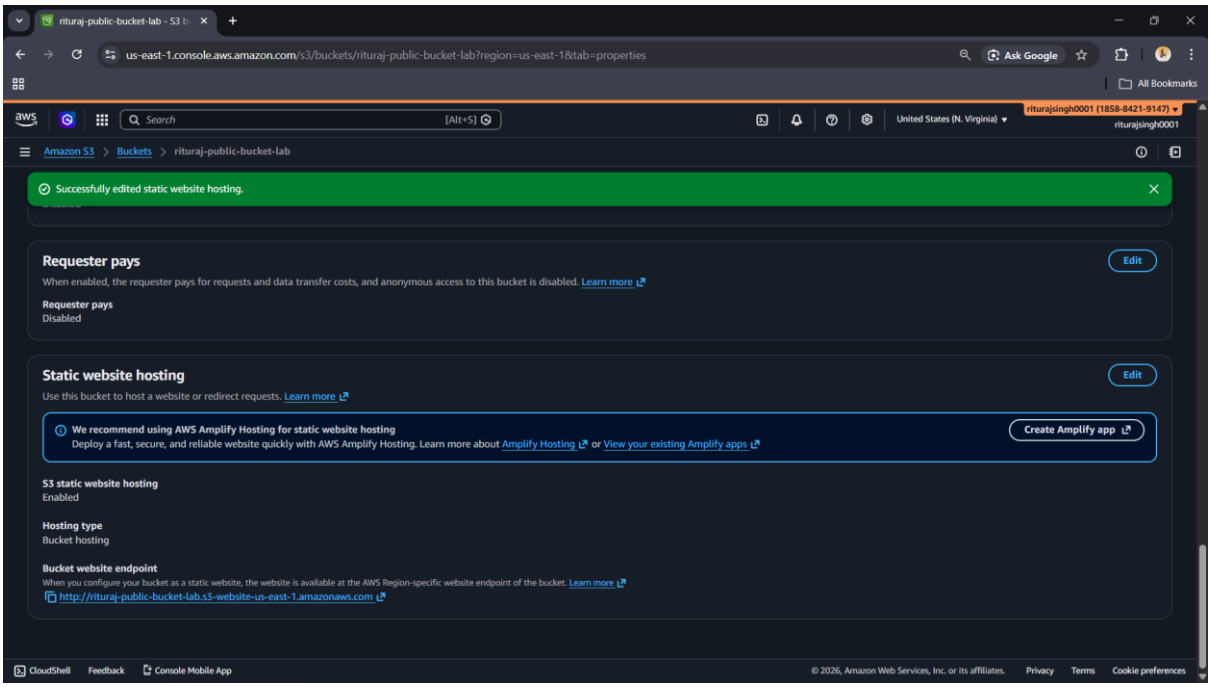


Step 3 – Enable Static Website Hosting (Attack Simulation)

- Go to **Properties tab**
- Enable **Static Website Hosting**
- Hosting type: Bucket hosting
- Index document: hello.txt



- Saved changes



- Copied website endpoint:

http://lab-public-bucket-xxxx.s3-website-us-east-1.amazonaws.com

Opened in **Incognito Mode**

❌ Issue Encountered: 403 Forbidden Error

Instead of accessing the file, the following error was observed:

403 Forbidden

Code: AccessDenied

Message: Access Denied

---

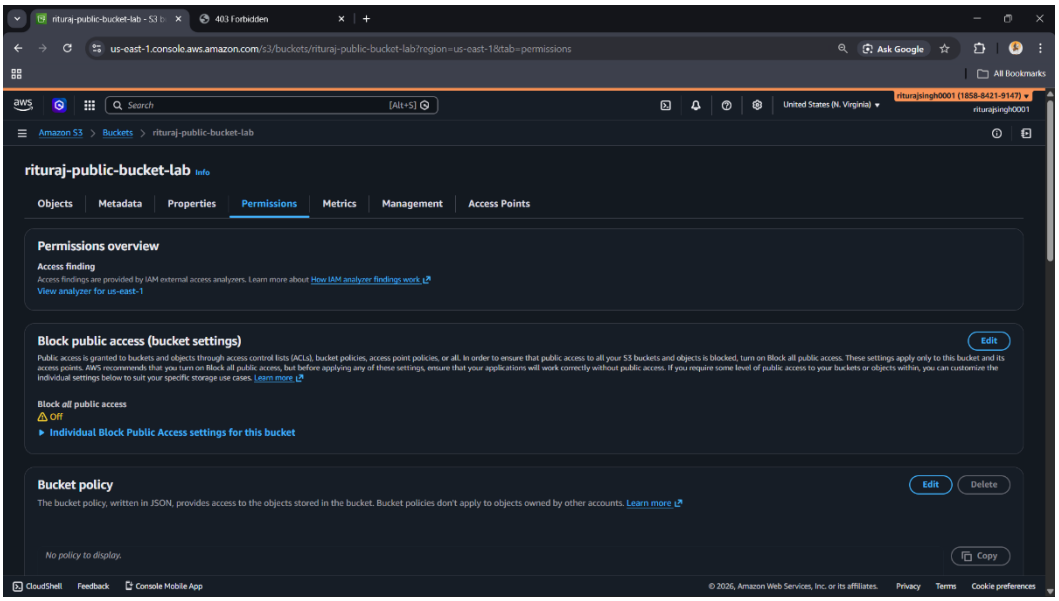
6. Root Cause Analysis

The error occurred because:

- Static Website Hosting was enabled ✅
- BUT **public read permissions were NOT granted** ❌

👉 AWS S3 requires explicit permission using:

- Bucket Policy



Without this, anonymous users cannot access data.

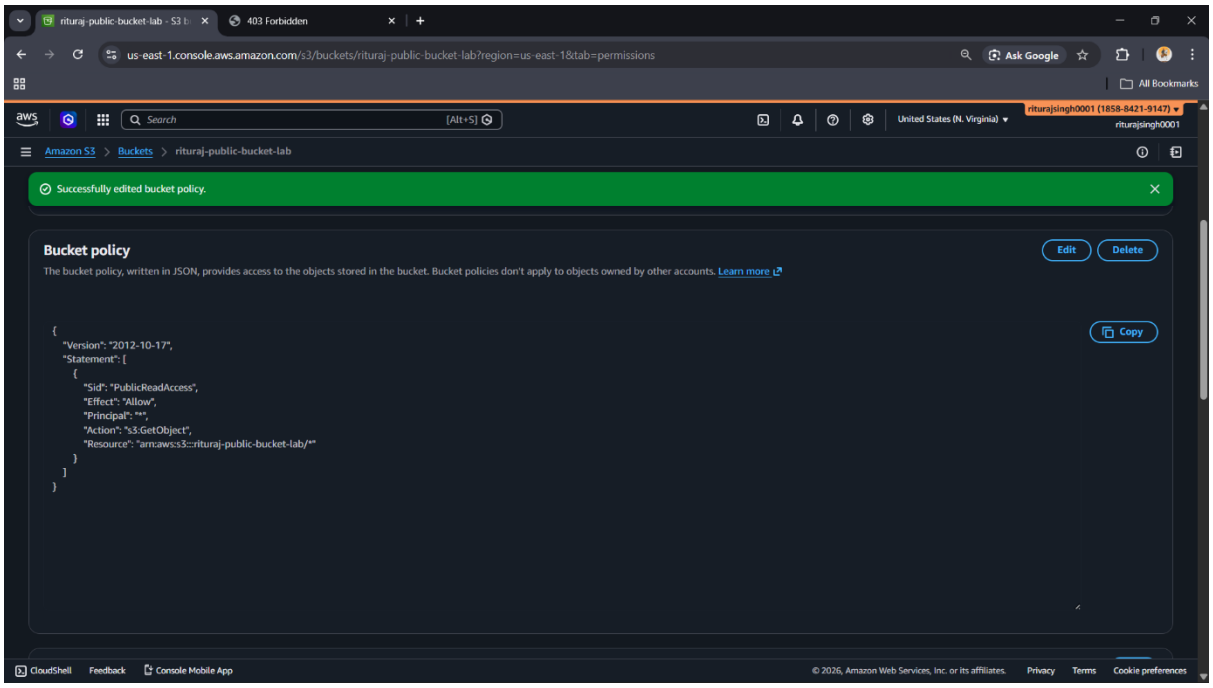
---

7. Fix – Applying Bucket Policy (Critical Step)

To resolve the issue, a **Bucket Policy** was added:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "PublicReadAccess",
```

```
    "Effect": "Allow",
    "Principal": "*",
    "Action": "s3:GetObject",
    "Resource": "arn:aws:s3:::your_bucket_name_/*"
  }
 ]
}
```

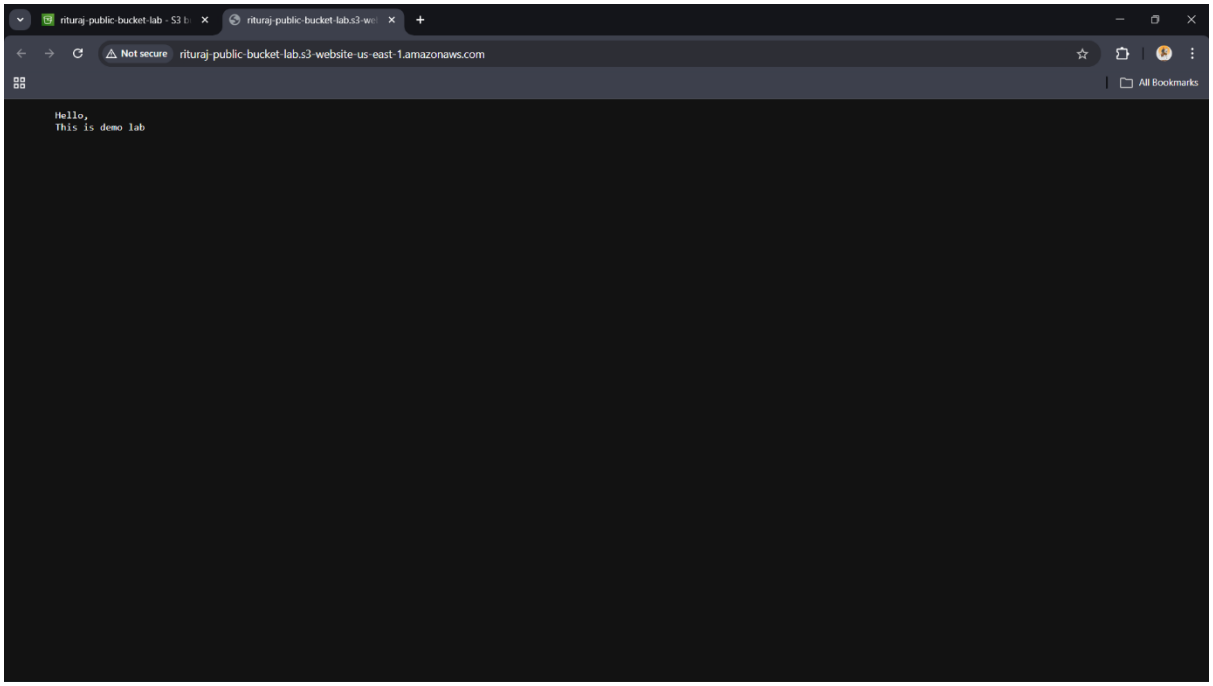


✅ This policy allows:

- Public users (*)
- To perform s3:GetObject
- On all objects inside the bucket

✅ **Result After Fix**

- Opened website endpoint again in incognito
- Successfully accessed content:



---

8. Security Risk (Attack Perspective)

This configuration demonstrates a **real-world vulnerability**:

- Sensitive data can be exposed publicly
- No authentication required
- Anyone with the URL can access data

👉 This is a common **cloud misconfiguration vulnerability**

---

9. Defense – Securing the Bucket

To mitigate the issue:

- Enabled **Block Public Access**
- Removed public access permissions
- Ensured no public bucket policy exists

🔒 **Verification**

- Opened website endpoint again
- Received:

---

10. Cleanup

- Deleted hello.txt
- Deleted S3 bucket

---

11. Conclusion

This lab demonstrated that:

- Enabling Static Website Hosting alone does NOT make content public
- Improper configurations can lead to **data exposure risks**
- Bucket Policies play a critical role in access control
- Security best practice is to **block public access unless absolutely required**