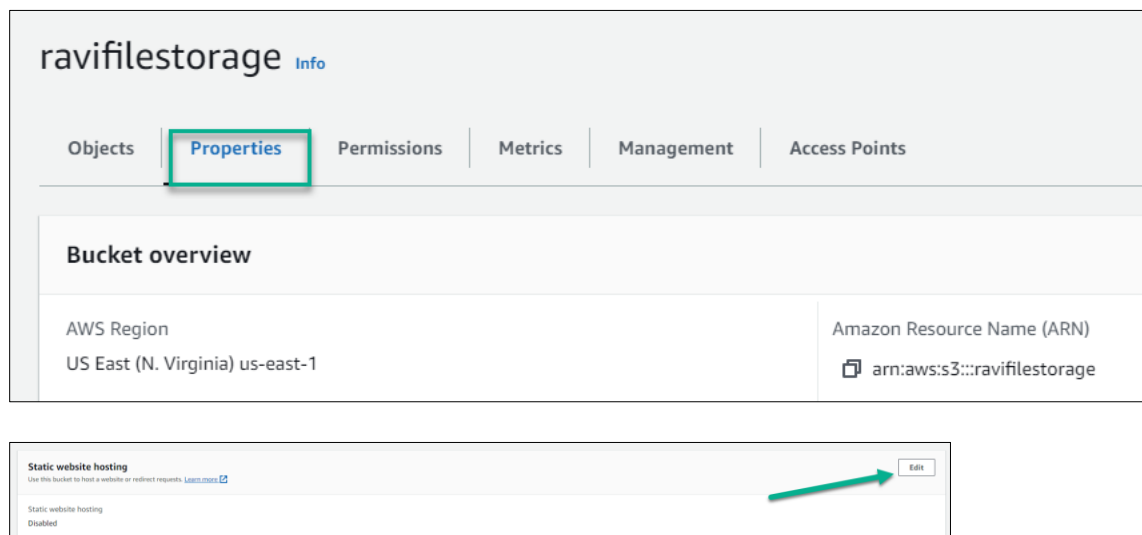# S3 Static Website Hosting

Problem Statement:

You work for Techarkit Corporation. Their application requires a storage service that can store files and publicly share them if required. Implement S3 for the same.

Tasks to Be Performed:

1. Use the created bucket in the previous task to host static websites, and upload an index.html file and error.html page.

2. Add a lifecycle rule for the bucket:

   a. Transition from Standard to Standard-IA in 60 days

   b. Expiration in 200 days

Answer

Now select the bucket you want to use for creating a static website, Click on Properties

Edit the **Static Website hosting**

Select Enable

Provide the index.html and error.html file name they should be case sensitive and name should be matching as per the apache default configuration

Click "**Save Changes**"

http://ravifilestorage.s3-website-us-east-1.amazonaws.com
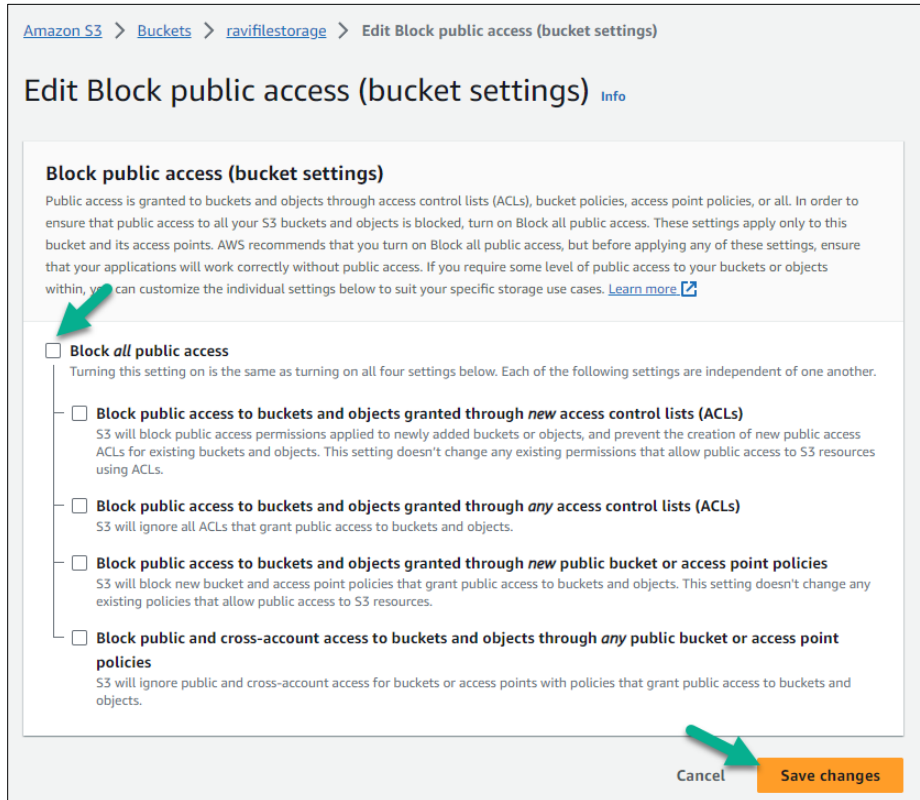
Once the website is enable it will provide you the endpoint details, you have to copy the URL and then browse is using the browser.

Before that, you have to upload index.html and error.html to the S3 bucket.

Amazon S3 > Buckets > ravifilestorage > Edit Block public access (bucket settings)
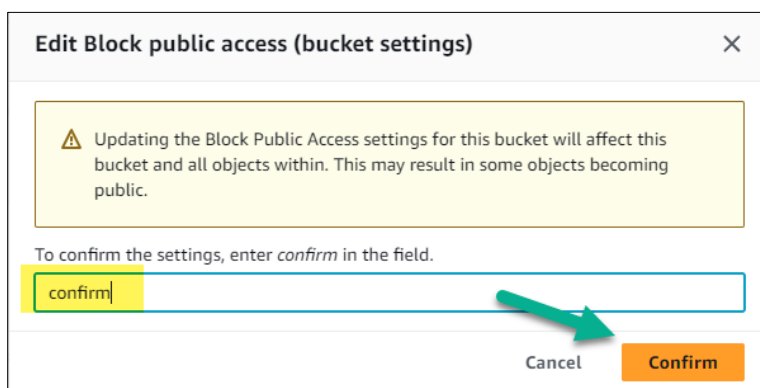
Edit Block public access (bucket settings) Info

Block public access (bucket settings)

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to all your S3 buckets and objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to your buckets or objects within, you can customize the individual settings below to suit your specific storage use cases. Learn more

☐ **Block all public access**
Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

☐ **Block public access to buckets and objects granted through new access control lists (ACLs)**
S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.

☐ **Block public access to buckets and objects granted through any access control lists (ACLs)**
S3 will ignore all ACLs that grant public access to buckets and objects.

☐ **Block public access to buckets and objects granted through new public bucket or access point policies**
S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.

☐ **Block public and cross-account access to buckets and objects through any public bucket or access point policies**
S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.

Cancel   **Save changes**

Edit Block public access (bucket settings)   ✕

⚠ Updating the Block Public Access settings for this bucket will affect this bucket and all objects within. This may result in some objects becoming public.
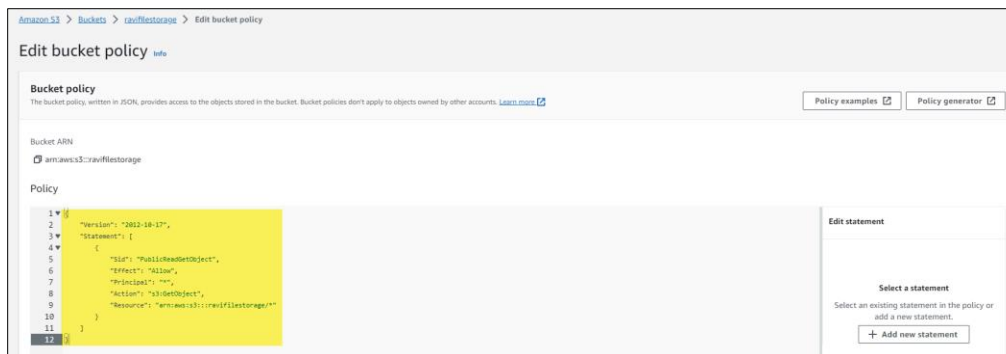
To confirm the settings, enter confirm in the field.

confirm

Cancel   **Confirm**

Once your enable the public access on the S3 bucket, you need to write the bucket policy otherwise it will give the below error message.
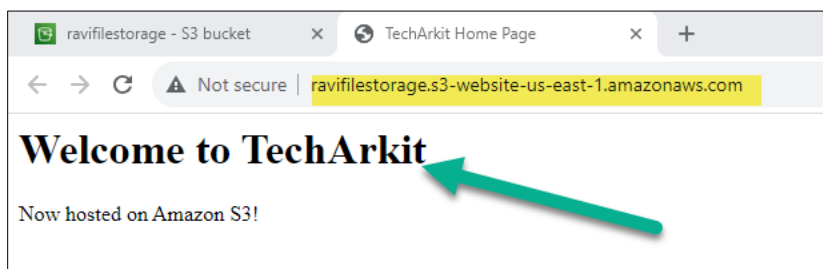
```
{
        "Version": "2012-10-17",
        "Statement": [
                {
                        "Sid": "PublicReadGetObject",
                        "Effect": "Allow",
                        "Principal": "*",
                        "Action": "s3:GetObject",
                        "Resource": "arn:aws:s3:::ravifilestorage/*"
                }
        ]
}
```

Add the bucket policy as shown above that's it your static website is published successfully.

Adding Life Cycle Policies

Select the S3 bucket you want to apply the Lifecycle rules, **S3 bucket → Management → Lifecycle rules → Create lifecycle rule**

Add Lifecycle rule name "StorageClassChange"

- Move current versions of objects between storage classes
- Move noncurrent versions of objects between storage classes
- Expire current versions of objects

To move current and noncurrent versions to Infrequent Access.

Minimum 30Days is required to objects to become noncurrent

That's All LifeCycle policies are configured successfully.

More Articles and Videos

Foundations of Cloud Computing

- AWS Certified Cloud Practitioner
- The Cloud
- 6 Advantages of Cloud
- Cloud Deployment Models
- AWS Global Infrastructure
- AWS Free Tier Account

Technology

- AWS Cloud Compute

- AWS Lambda Function-as-a-service
- AWS Fargate
- What is CDN
- AWS S3
- AWS Storage Services
- AWS CloudFront
- AWS VPC
- AWS Networking Services
- How to Launch the First EC2 instance
- AWS Database Services Deep Dive
- AWS Migration Services
- AWS Analytics Services
- AWS Reckognize
- Amazon Machine Learning
- AWS Development Tools