

# PERSONAL BLOG ON IBM CLOUD STATIC WEB APPS

College : Sri krishna engineering college

Domain : Information technology (IT)

Dept : B. Tech information technology

Submitted by

R.Surya prakash

Reg.no:412021205004

## Phase 4 :

IBM cloud is a suite of cloud computing services which offers both platform as a service (PaaS) and Infrastructure as a Service (IaaS) and Software as a service (SaaS) . With IBM, IaaS cloud organizations can deploy virtualized IT resources such as computing power, storage and networking over the internet.



## Characteristics:


IBM Cloud Object Storage is **easy to start small and can grow seamlessly with investment protection from TB to EB of capacity**. IBM Cloud Object Storage is a parallel storage system and provides concurrent access from anywhere with an any-to- any-to any architecture.



# Hosting Technique:

This blog post explains how to host a **static website** on **IBM Cloud**. These websites are rendered client-side by the browser from static assets, like HTML, CSS and JS files. They do not need a server-side component to create pages dynamically at runtime. Static websites are often combined with backend APIs to create **Single Page Applications**.





Hosting static websites on IBM Cloud uses **Cloud Object Storage (COS)** and **Cloud Internet Services (CIS)** (with **Page Rules** and **Edge Functions**). These services provide the following features needed to serve static websites.

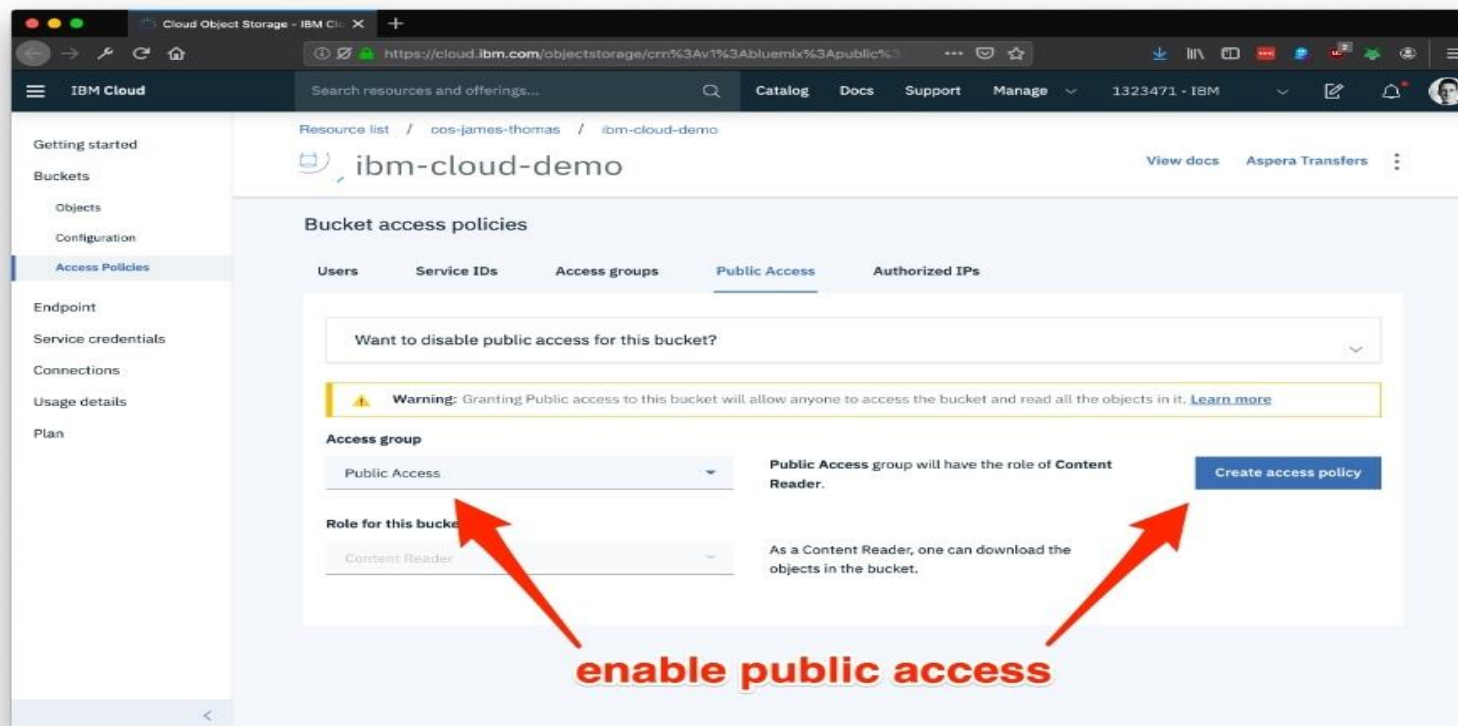
- **Auto-serving static assets from provider-managed HTTP service (Cloud Object Storage).**
- **Custom domain support to serve content from user-controlled domain name (CIS - Page Rules).**
- **Configurable Index and Error documents (CIS - Edge Functions).**



# Enable public access to bucket files

- Click the “*Access Policies*” menu item from the bucket level menu.
- Click the “*Public Access*” tab from the bucket access policy page.
- Check the Access Group drop-down has “*Public Access*” option selected.
- Click the “*Create access policy*” and then “*Enable*” on the pop menu.

# Diagram:



# Check bucket files are accessible:

- Open the “*Configuration*” panel on the bucket page.
- Retrieve the **public endpoint** shown, e.g.  
`s3.<REGION>.cloud-object-storage.appdomain.cloud`

IBM Cloud

Search resources and offerings...

Catalog Docs Support Manage 1323471 - IBM

Getting started

Buckets

Objects

**Configuration**

Access Policies

Endpoint

Service credentials

Connections

Usage details

Plan

### Endpoints

Endpoints are used hand in hand with your credentials (i.e. keys, CRN, bucket name) to tell your service where to look for this bucket. Depending on where your service or applications is located you will want to use one of the below endpoint types. [Learn more](#)

**Note: Endpoints have been updated please use these endpoints listed below for any new applications.**

### Public Endpoint Hostname

<b>Private</b> ⓘ Use private endpoints to point applications or services that are hosted in the IBM cloud (excluding Cloud Foundry services).	<b>Public</b> ⓘ Use public endpoints to point applications or services that are hosted outside of the IBM cloud or for Cloud Foundry applications hosted in the IBM cloud.
<code>s3.private.eu-gb.cloud-object-storage.appdomain.cloud</code>	<code>s3.eu-gb.cloud-object-storage.appdomain.cloud</code>





# URLs:

vhost addressing

```
<BUCKET_NAME>.s3.eu-gb.cloud-object-  
storage.appdomain.cloud/index.html
```

url path addressing

```
s3.<REGION>.cloud-object-  
storage.appdomain.cloud/<BUCKET_NAME>/index.html
```


## Coding :

```
const INDEX_DOCUMENT = 'index.html'
const ERROR_DOCUMENT = '404.html'


addEventListener('fetch', event => {
  event.respondWith(handleRequest(event))
})

async function handleRequest(request)
{
  const url = new URL(request.url)

  // if request is a directory path,
  append the index document.
```



```
if (url.pathname.endsWith('/')) {  
    url.pathname =  
    `${url.pathname}${INDEX_DOCUMENT}`  
    request = new Request(url,  
request)  
}  
  
let response = await fetch(request)  
  
// if bucket file is missing,  
return error page.
```



```
if (response.status === 404) {  
  url.pathname = ERROR_DOCUMENT  
  request = new Request(url,  
request)  
  response = await fetch(request)  
  
  response = new  
Response(response.body, {  
    status: 404,  
    statusText: 'Not Found',  
    headers: response.headers  
  })  
}  
  
return response  
}
```



# Text index and error pages:

- Confirm that `http://<SUB_DOMAIN>.<CUSTOM_DOMAIN>/` returns the same page as `http://<SUB_DOMAIN>.<CUSTOM_DOMAIN>/index.html`
- Confirm that `http://<SUB_DOMAIN>.<CUSTOM_DOMAIN>/missing-page.html` returns the error page. This should be different to the XML error response returned by visiting `<BUCKET_NAME>.s3.<REGION>.cloud-object-storage.appdomain.cloud/missing-page.html`.



# Conclusion:

Static web sites can be hosted on IBM Cloud using Cloud Object Storage and Cloud Internet Services.

Cloud Object stores page files needed to render the static website. Anonymous bucket file access means files are accessible as public HTTP endpoints, without having to run infrastructure to serve the assets.