Deploy a Static Website on Amazon S3 and CloudFront

ARCHITECTURAL DIAGRAM:

User Browser (player visits the game)

↓

CloudFront CDN (Delivers content globally, caches assets, provides HTTPS)

↓

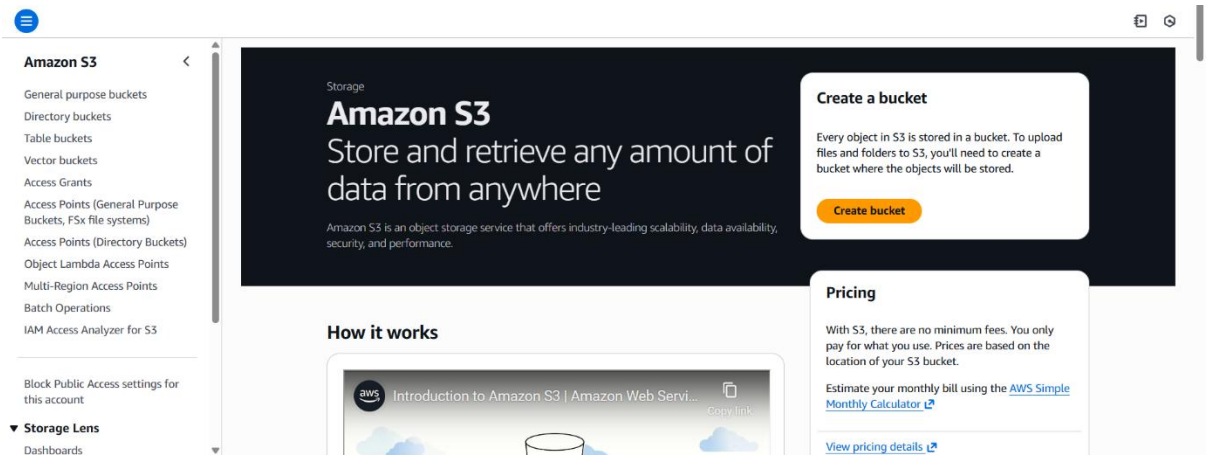
Origin Access Control (OAC) (Allows CloudFront to securely access private S3 content)

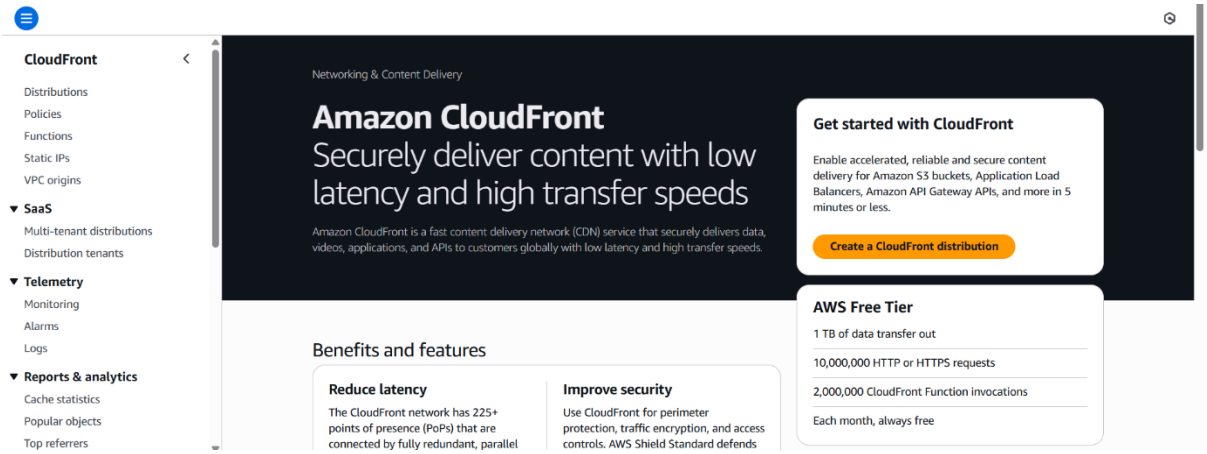
↓

S3 Bucket (Private) (Stores HTML, CSS, JS, images for the static game website).

DESCRIPTION:

- A fully deployed static website hosted on Amazon S3 and served securely through CloudFront with HTTPS.
- Includes optional custom domain setup using Route 53 and SSL certificates via AWS Certificate Manager.



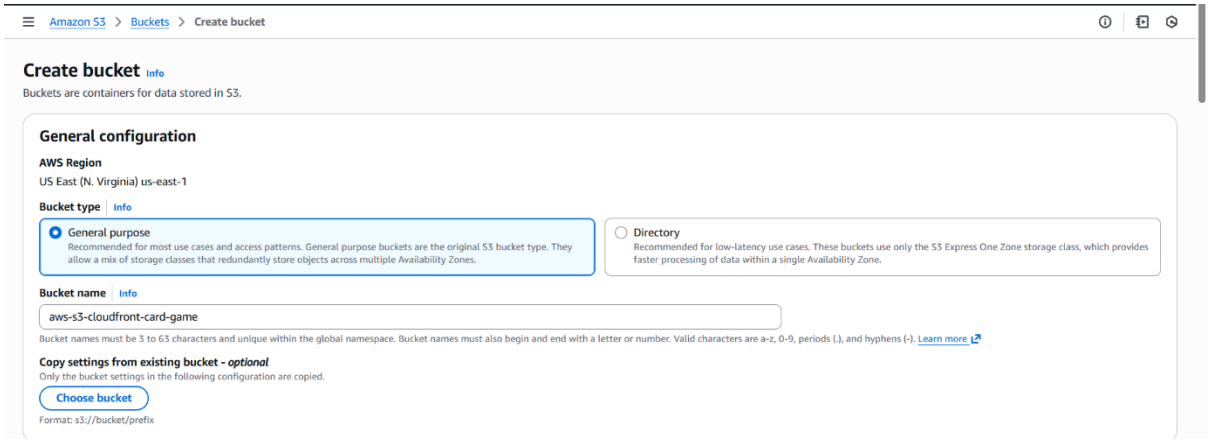


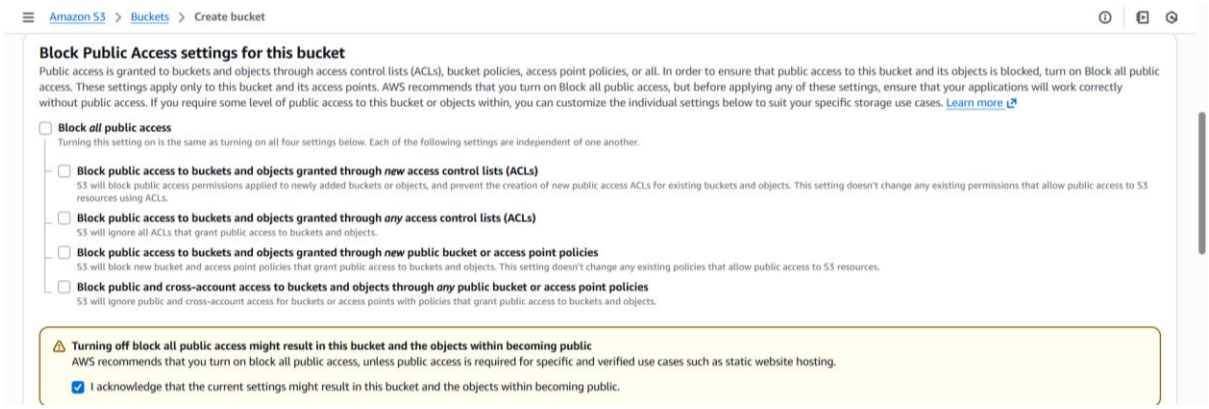
Here's a clear, complete walkthrough of **all steps involved** in creating and deploying a fully static website on **Amazon S3**, served securely over **CloudFront**, with optional **Route 53 + ACM** for a custom domain.

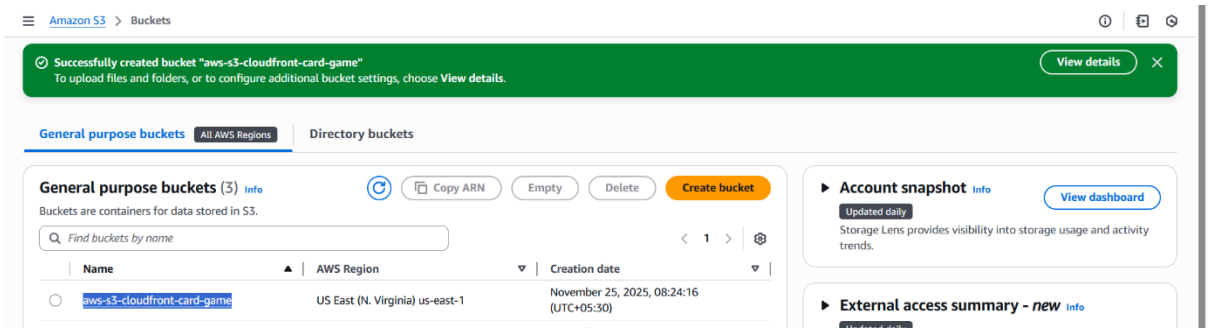
STEP 1 — CREATE AN S3 BUCKET:

1. Open AWS Console → **S3**.
2. Click "**Create bucket**".
3. Enter a bucket name

   Bucket name: **aws-s3-cloudfront-card-game**
4. Uncheck **"Block all public access."**
5. Click **Create bucket**.

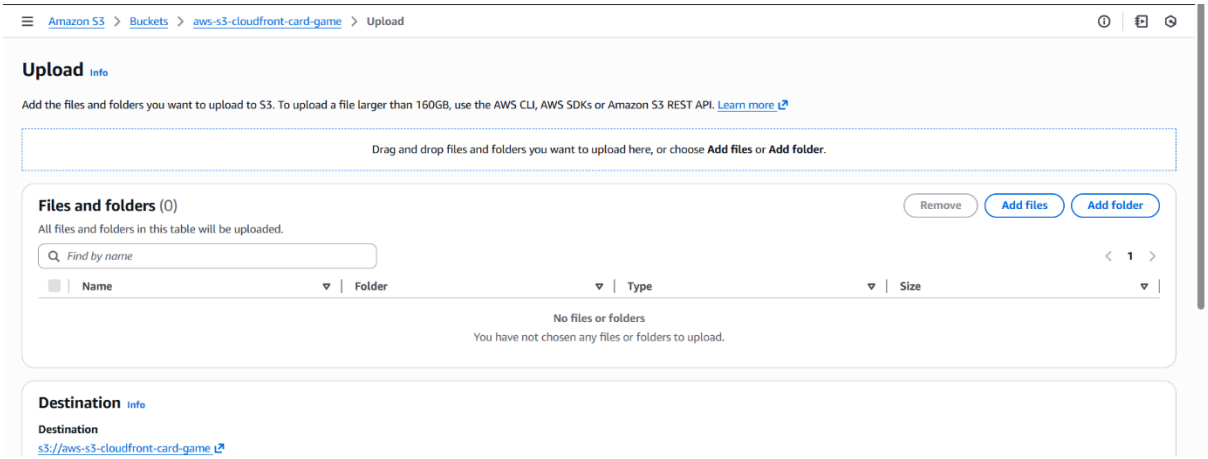


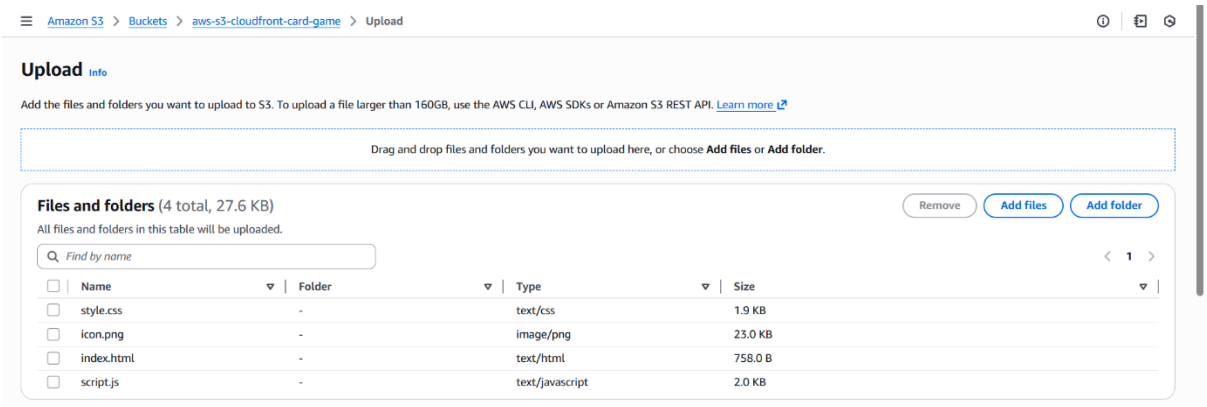


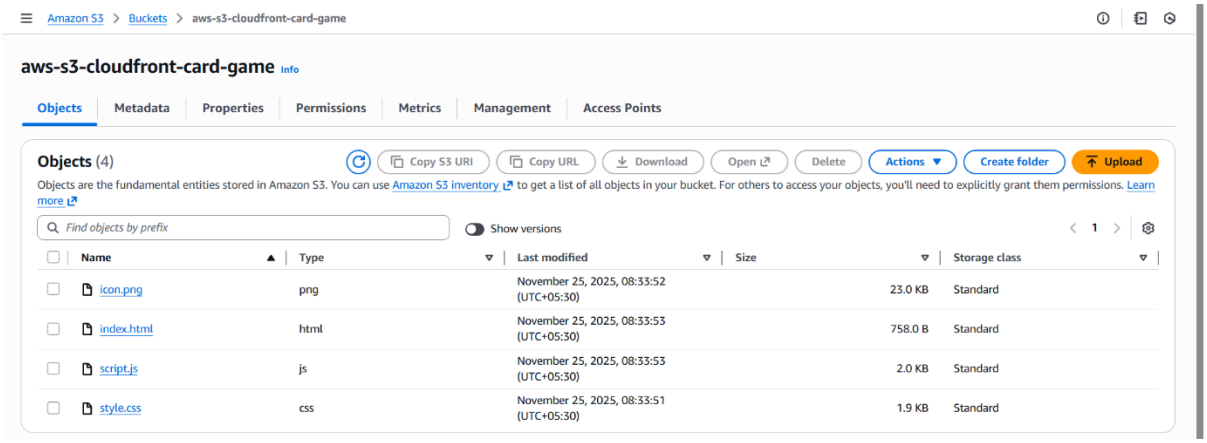


STEP 2 — UPLOAD WEBSITE FILES:

1. Open the bucket. (bucket - **aws-s3-cloudfront-card-game**)
2. Click **Upload → Add files**.
3. Upload:
    - index.html
    - CSS/JS/images folders
4. Click **Upload**.

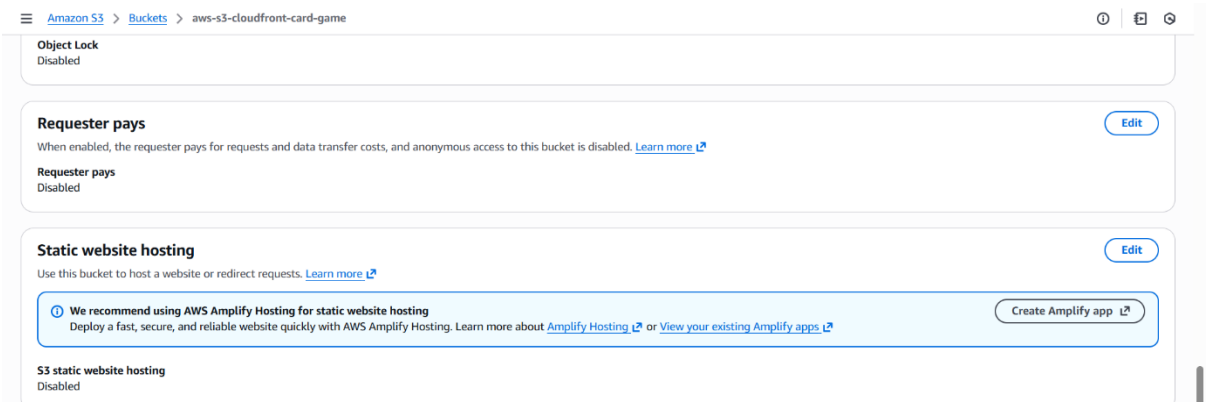


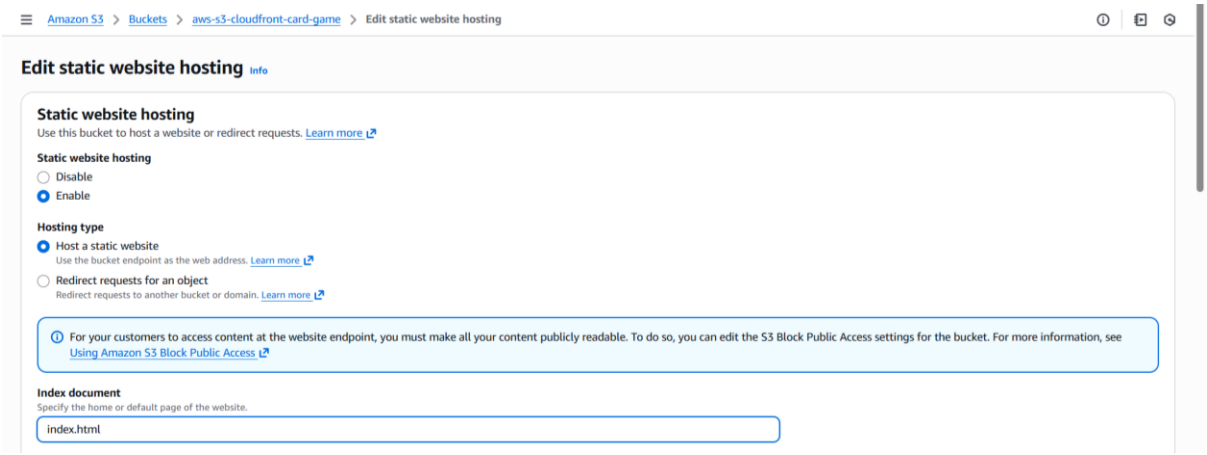


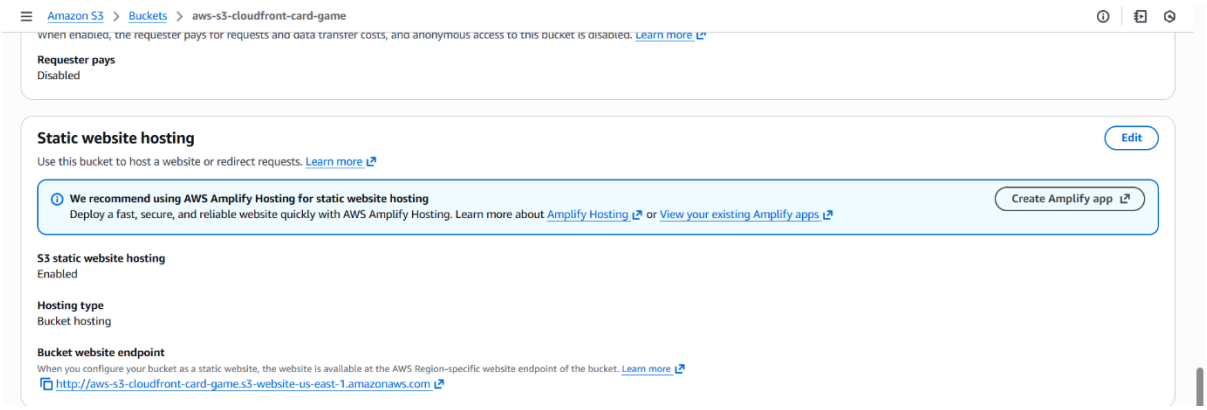


STEP 3 — ENABLE STATIC WEBSITE HOSTING:

1. Open your bucket.
2. Go to **Properties**.
3. Scroll to **Static website hosting**.
4. Click **Edit → Enable**.
5. Enter:
    - **Index document:** index.html
6. Click **Save changes**.
7. Copy the **Website endpoint URL** (for testing).





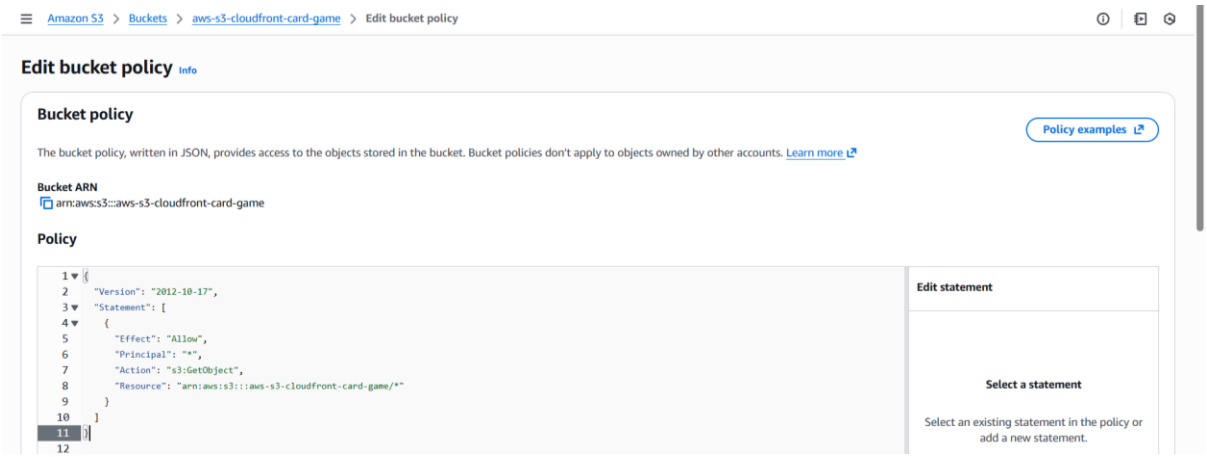


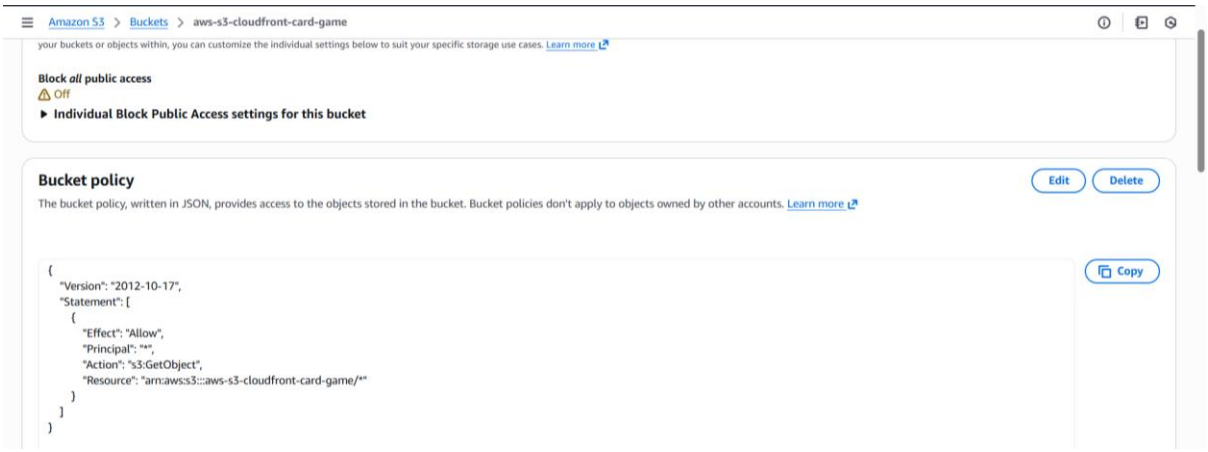
STEP 4— ADD BUCKET POLICY (SO FILES CAN BE VIEWED):

1. Go to **Permissions** tab.
2. Scroll to **Bucket Policy → Edit**.
3. Paste the policy and Save the policy.

BUCKET POLICY:

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





STEP 5 — CREATE A CLOUDFRONT DISTRIBUTION (NO CUSTOM DOMAIN):

1. Go to CloudFront → **Create distribution**.

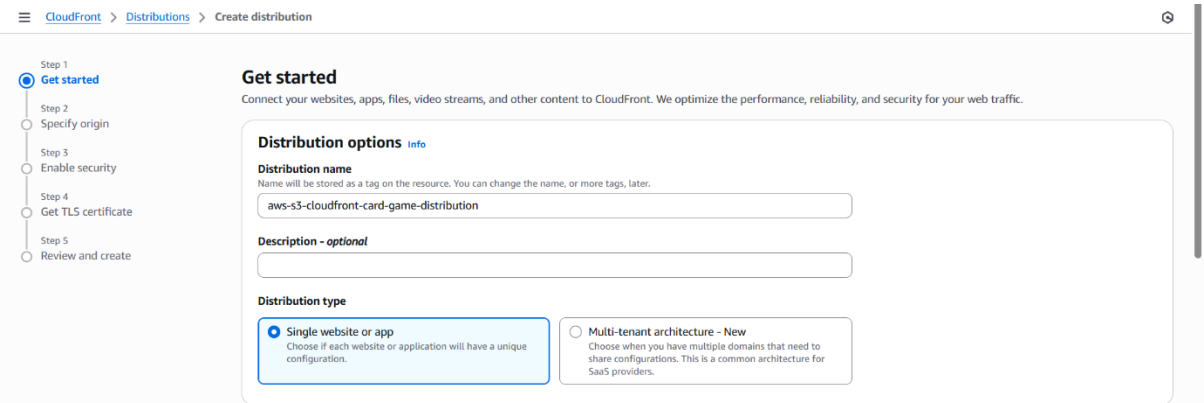
   Distribution name: aws-s3-cloudfront-card-game-distribution

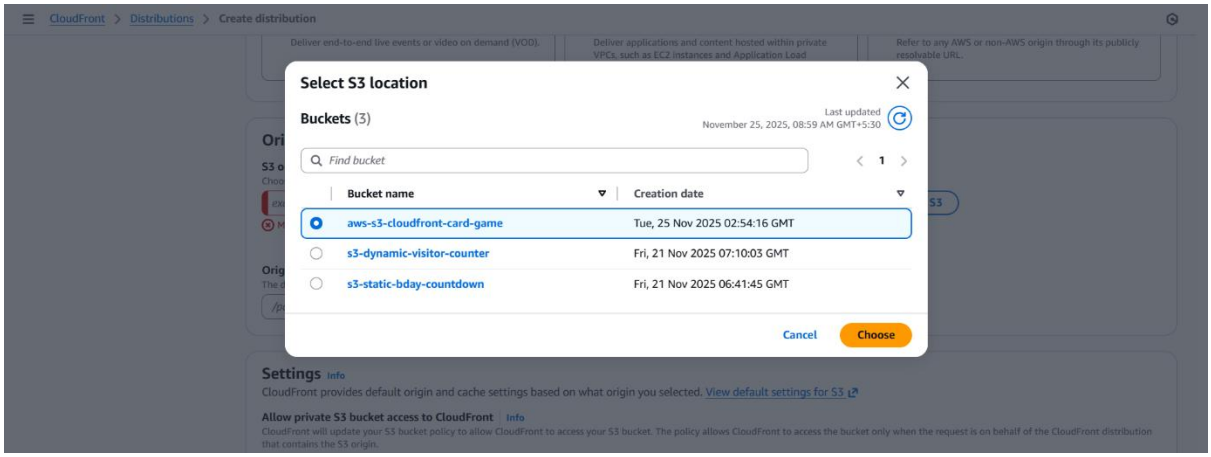
2. Under **Origin**, choose your **S3 static website endpoint**.
3. Set:
   - **Viewer protocol policy:** Redirect HTTP to HTTPS
4. At bottom:
   - **Default root object:** index.html
5. Click **Create distribution**.

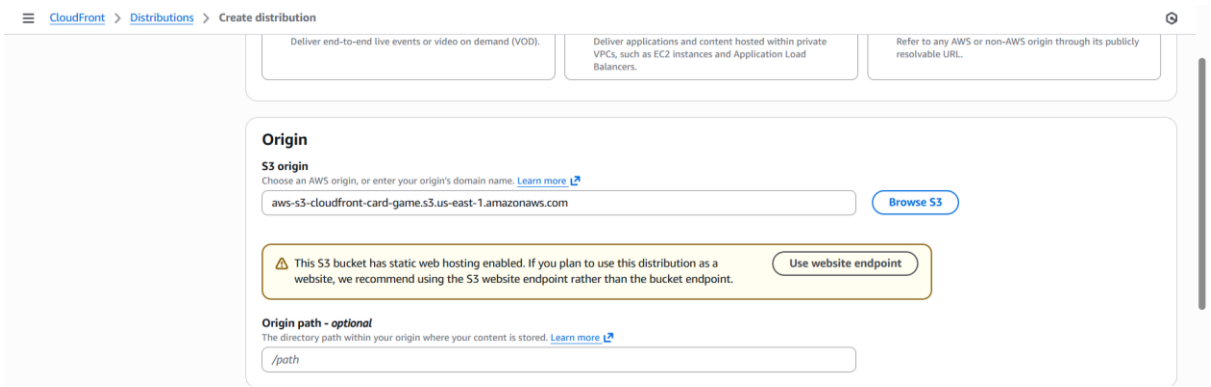
~ Wait 5–10 minutes.

~ You get a URL like:

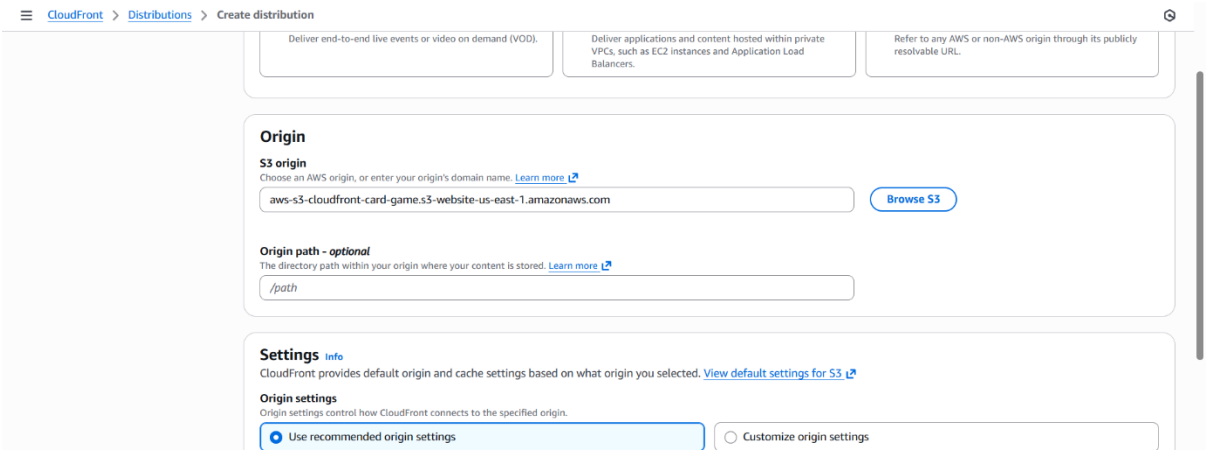
https://d15ofsgzywtlqm.cloudfront.net ← **your site now works with HTTPS!**







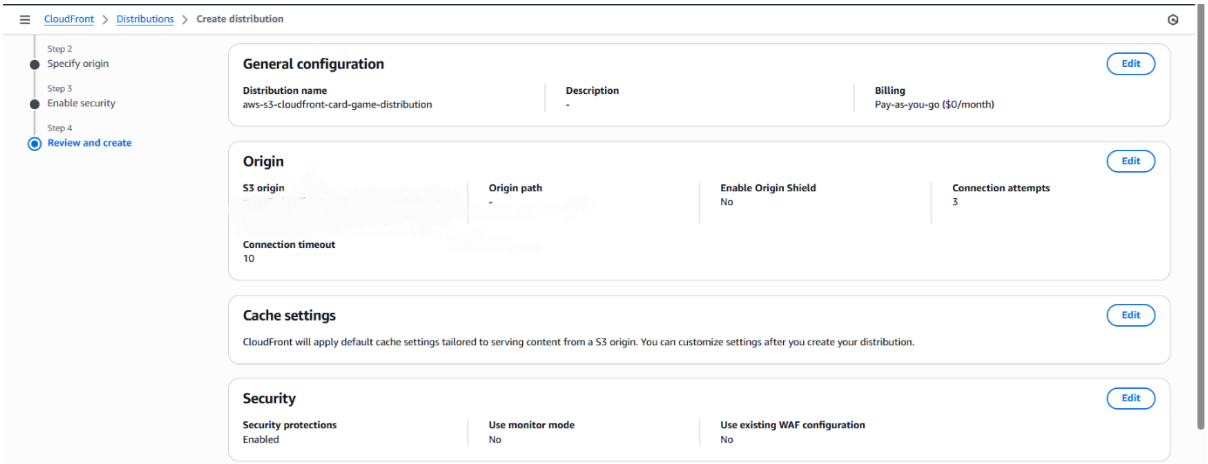
~ must select **“use Website endpoint”**.

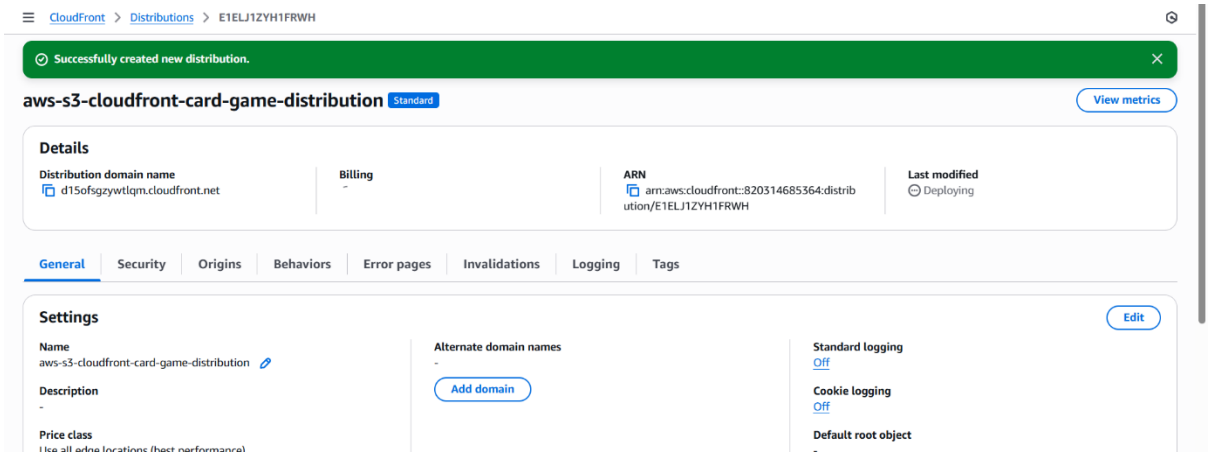


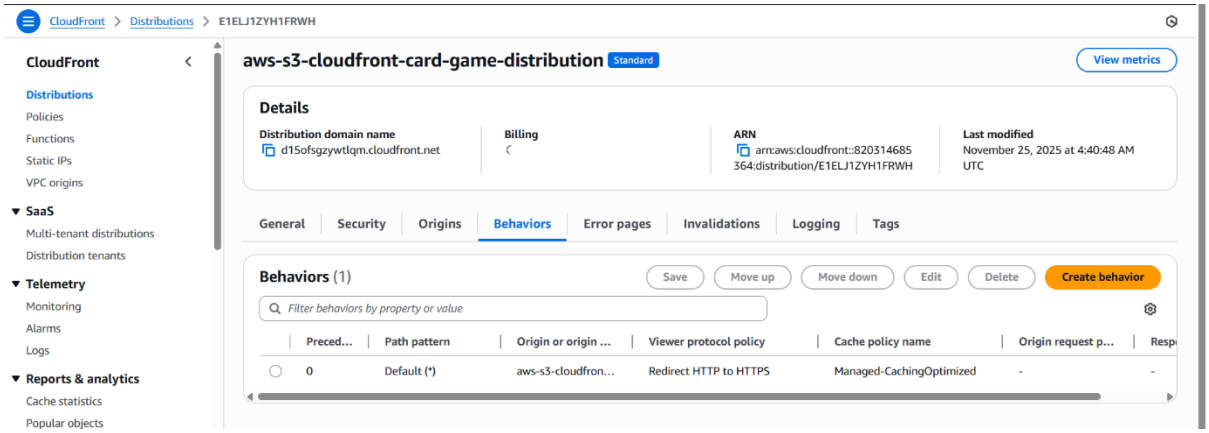
AWS WAF:

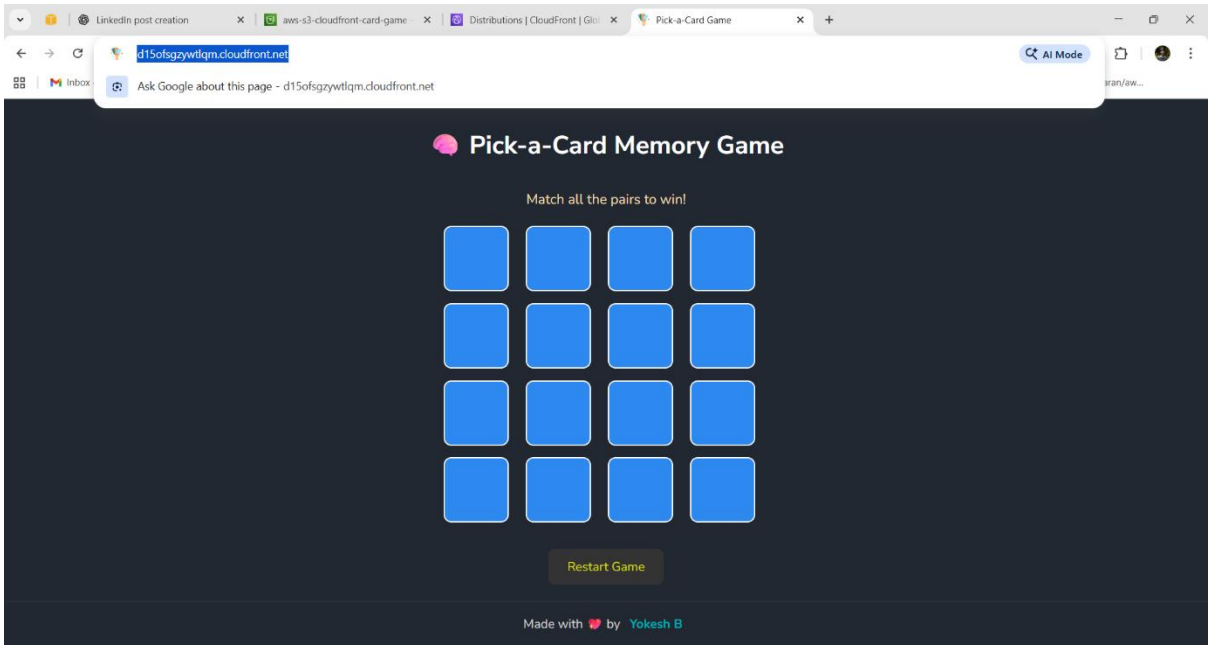
- It is a web application firewall that protects apps and APIs from common attacks like SQL Injection (SQLi) and Cross-Site Scripting (XSS).
- It lets you create rules to block, allow, or monitor HTTP requests.
- Works with CloudFront, ALB, API Gateway, and offers managed security rules.

~ **AWS WAF** is enabled to prevent cyber-attacks.




CloudFront > Distributions > E1ELJ1ZYH1FRWH
Successfully created new distribution.
aws-s3-cloudfront-card-game-distribution
Standard
View metrics
Details
Distribution domain name
d15ofsgzywtlqm.cloudfront.net
Billing
ARN
arn:aws:cloudfront::820314685364:distribution/E1ELJ1ZYH1FRWH
Last modified
Deploying
General
Security
Origins
Behaviors
Error pages
Invalidations
Logging
Tags
Settings
Edit
Name
aws-s3-cloudfront-card-game-distribution
Description
Price class
Alternate domain names
Add domain
Standard logging
Off
Cookie logging
Off
Default root object


CloudFront > Distributions > E1ELJ1ZYH1FRWH
CloudFront
Distributions
Policies
Functions
Static IPs
VPC origins
SaaS
Multi-tenant distributions
Distribution tenants
Telemetry
Monitoring
Alarms
Logs
Reports & analytics
Cache statistics
Popular objects
aws-s3-cloudfront-card-game-distribution
Standard
View metrics
Details
Distribution domain name
d15ofsgzywtlqm.cloudfront.net
Billing
ARN
arn:aws:cloudfront::820314685364:distribution/E1ELJ1ZYH1FRWH
Last modified
November 25, 2025 at 4:40:48 AM UTC
General
Security
Origins
Behaviors
Error pages
Invalidations
Logging
Tags
Behaviors (1)
Save
Move up
Move down
Edit
Delete
Create behavior
Filter behaviors by property or value
Preced...
Path pattern
Origin or origin ...
Viewer protocol policy
Cache policy name
Origin request p...
0
Default (*)
aws-s3-cloudfron...
Redirect HTTP to HTTPS
Managed-CachingOptimized


d15ofsgzywtlqm.cloudfront.net
Ask Google about this page - d15ofsgzywtlqm.cloudfront.net
AI Mode
Pick-a-Card Memory Game
Match all the pairs to win!
Restart Game
Made with by Yokesh B

⭐ OPTIONAL (RECOMMENDED FOR SECURE SETUP)

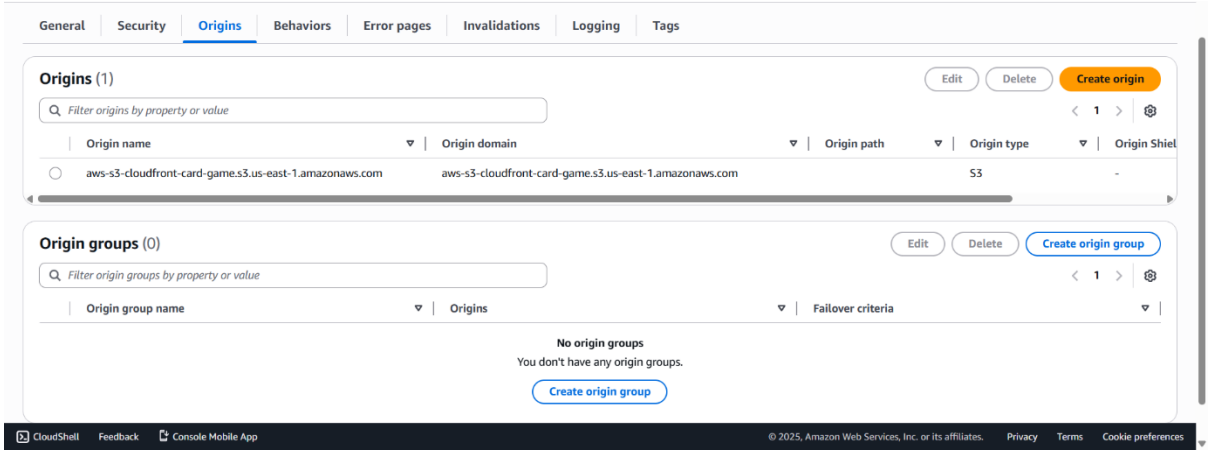
STEP 6 — TURN OFF PUBLIC ACCESS ON S3 AND USE OAC (ORIGIN ACCESS CONTROL):

~ This ensures your S3 bucket is **PRIVATE**, and only CloudFront can access it.

1. Go to **CloudFront → Distributions → Your Distribution → Origins**
2. Select your origin → **Edit**
3. Under "Origin access", choose:

   Origin Access Control (recommended)

   → Create new OAC
4. Save changes
5. CloudFront shows a yellow banner:
   "Update the S3 bucket policy"
6. Click **Copy Policy**
7. Go to **S3 → Your Bucket → Permissions → Bucket policy → Edit**
8. Replace any existing policy with the CloudFront OAC policy
9. Save



~ Now your S3 bucket becomes **private**, and CloudFront becomes the only gateway.

STEP 7 — UPLOAD YOUR STATIC WEBSITE TO S3:

1. Go to **S3 → Your bucket**
2. Click **Upload**
3. Add all site files:
   - index.html
   - CSS files
   - JS files
   - Images

~ CloudFront automatically fetches the content from S3.

STEP 8 — TEST YOUR WEBSITE:

→ Open your CloudFront URL:

https://d15ofsgzywtlqm.cloudfront.net

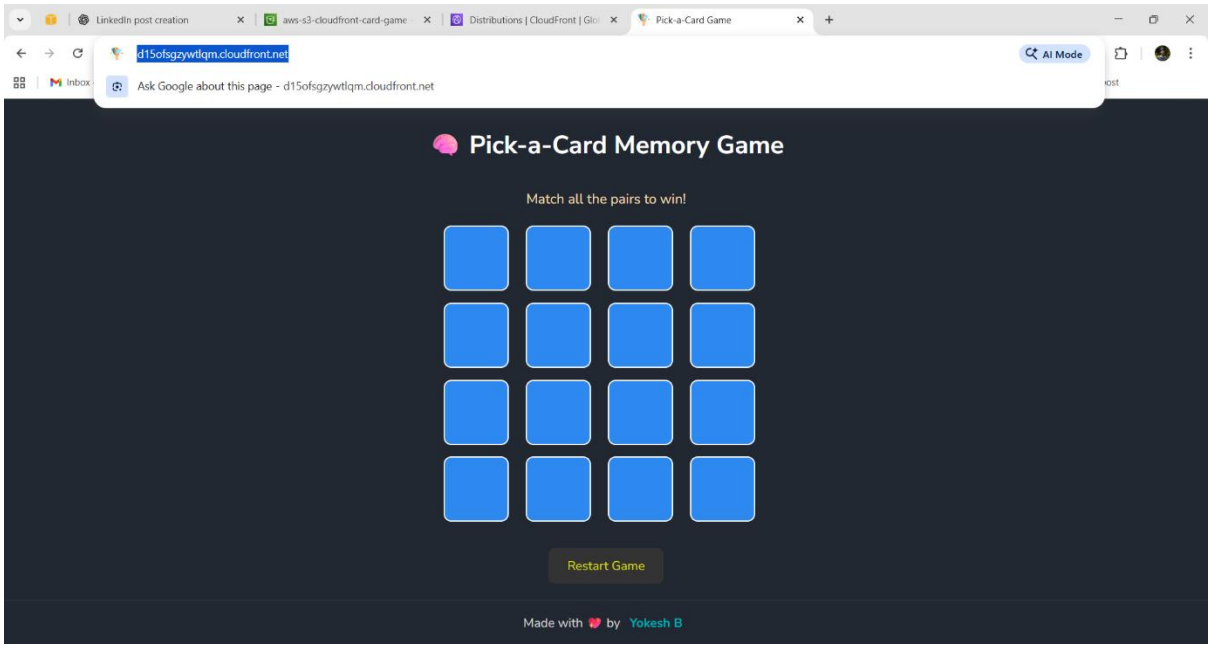
→ You should see your website.

✓ Works with HTTPS

✓ Global CDN performance

✓ Free Tier friendly

✓ No domain needed.



🎉 **DONE**

Your static site is now:

- ✓ Hosted on **S3**
- ✓ Served securely by **CloudFront**
- ✓ Uses **OAC** for security