

HOST A WEBSITE ON AWS S3



Amazon
S3



Route53



AWS Amplify

AMAZON S3 BUCKET

- Pick your region
- **Create** an Amazon s3 bucket
- Name the s3 bucket- names are globally unique (name is as the domain you want)
- Pick **ACLs enabled**: a set of rules that decides who can get access to resources
- Remove **Block all public access**: so everyone can get access
- Check the box **I acknowledge that the current settings might result in this bucket and the objects within becoming public.**
- Enable **Bucket Versioning**
- **Create The Bucket!**

Create bucket

AWS Region
Israel (Tel Aviv) il-central-1

Bucket name [Info](#)
portfolio-website-moriel

Bucket name must be unique within the global namespace and follow the bucket naming rules. [See rules for bucket naming](#)

Copy settings from existing bucket - *optional*
Only the bucket settings in the following configuration are copied.

[Choose bucket](#)


Format: s3://bucket/prefix

Object Ownership [Info](#)

Control ownership of objects written to this bucket from other AWS accounts and the use of access control lists (ACLs). Object ownership determines who can specify access to objects.

☐ **ACLs disabled (recommended)**
All objects in this bucket are owned by this account. Access to this bucket and its objects is specified using only policies.

☒ **ACLs enabled**
Objects in this bucket can be owned by other AWS accounts. Access to this bucket and its objects can be specified using ACLs.

 We recommend disabling ACLs, unless you need to control access for each object individually or to have the object writer own the data they upload. Using a bucket policy instead of ACLs to share data with users outside of your account simplifies permissions management and auditing.

Object Ownership

☒ **Bucket owner preferred**
If new objects written to this bucket specify the bucket-owner-full-control canned ACL, they are owned by the bucket owner. Otherwise, they are owned by the object writer.

☐ **Object writer**

Block Public Access settings for this bucket

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 this bucket and its 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 this bucket 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.

 **Turning off block all public access might result in this bucket and the objects within becoming public**
AWS recommends that you turn on block all public access, unless public access is required for specific and verified use cases such as static website hosting.

 ☒ I acknowledge that the current settings might result in this bucket and the objects within becoming public.

Bucket Versioning

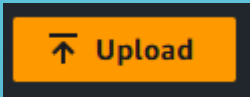
Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and restore every version of every object stored in your Amazon S3 bucket. With versioning, you can easily recover from both unintended user actions and application failures. [Learn more](#)

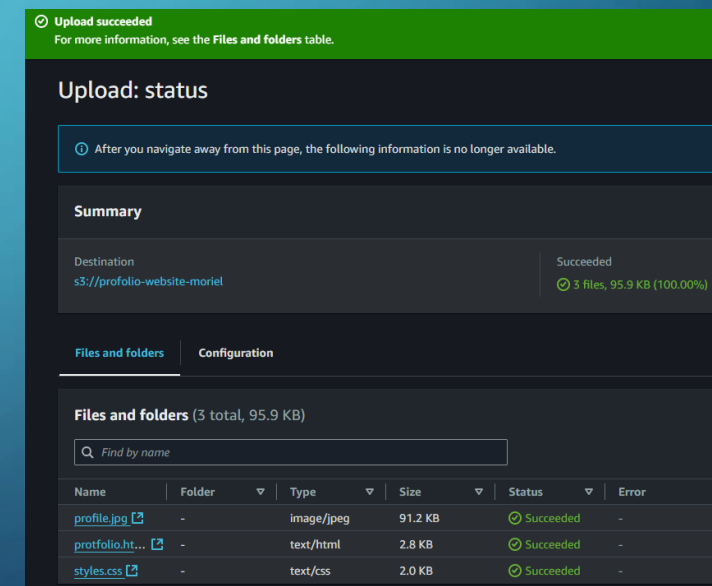
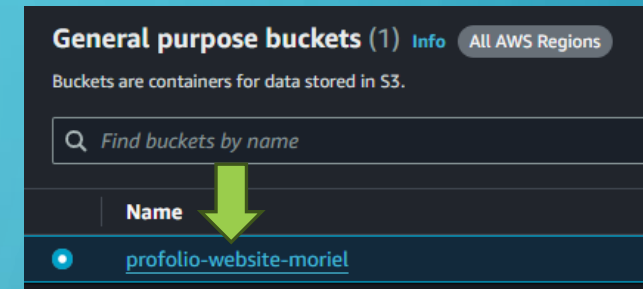
Bucket Versioning

☐ Disable

☒ Enable

UPLOAD TO S3 BUCKET

- After the bucket is created, we need to upload the HTML and CSS files
- Click on the bucket then click 
- Wait for it to finish



ROUTE 53: HOSTING ON YOUR DOMAIN

- Enter Route 53
- under Domains > Registered Domain
- Pick Register domains
- Check for availability and TLD
 - .click is the cheapest
- Click on Select
 - You have the option to auto renew
- Click on next and fill Registrant contact
- Wait until approval of the domain

Register domains [Info](#)

Pricing for domain names varies by top-level domain (TLD). For more information, view [price with different TLDs](#).

Search for domain

Check availability for a domain

► **Standard pricing**

Pricing for domain names varies by top-level domain (TLD), such as .com or .org.

Search result

Domain	Price/year	Actions
morielmauni.com Exact match	14.00 USD Renews at 14.00 USD	Select

Pricing [Info](#)

Domain pricing options

Domain name Duration (price) Auto-renew ☒ On

Auto-renew is turned on for 1 domain.
We will send an email to the registrant contact before expiration to remind you that auto-renew is currently turned on. You can turn it off at any time by using the Route 53 console. For more information, see [Renewing Registration for a Domain](#).

Subtotal: **14.00 USD**
Applicable taxes will be calculated at checkout.

[Cancel](#) [Next](#)

Route 53

Dashboard
Hosted zones
Health checks
Profiles [New](#)

▼ IP-based routing

CIDR collections

▼ Traffic flow

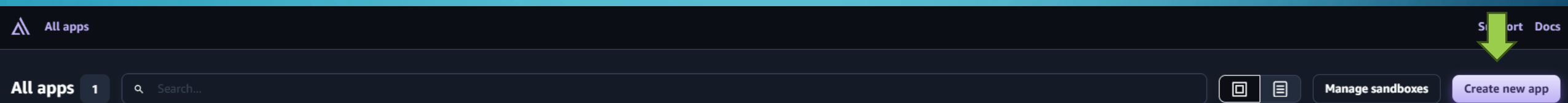
Traffic policies
Policy records

▼ Domains

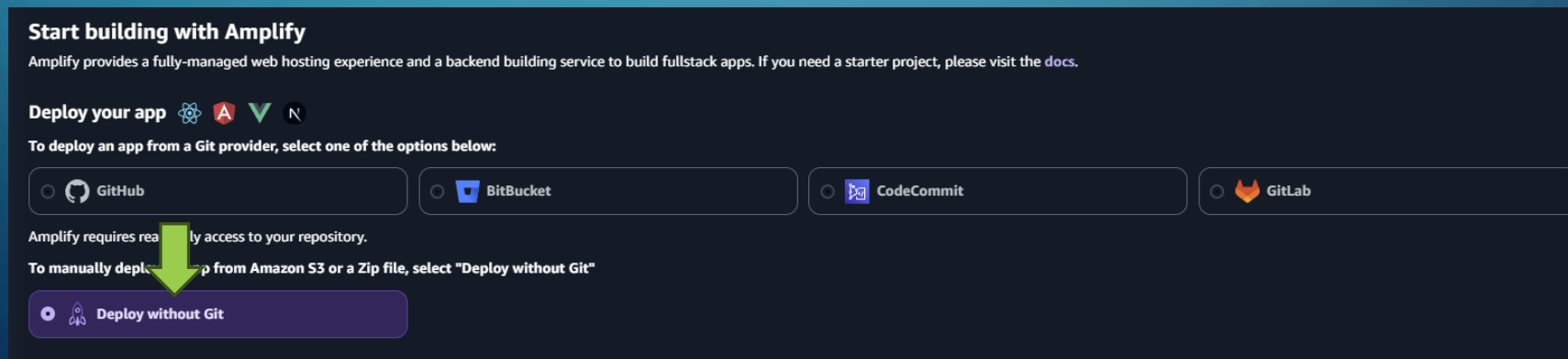
[Registered domains](#)
Requests

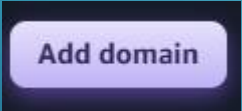
AMAZON AMPLIFY

- After we got our s3 bucket with all the files and the domain we want from Route 53, now we going to host out website online with HTTPS certification using Amazon Amplify.
- Also, Amazon Amplify secures the s3 bucket with the polices we need.



- After Creating new app, pick Deploy with Git



- Name the app
- Pick Amazon s3
- And then Save and Deploy
- After the app is live
- Let's add out custom domain
- Under Hosting > Custom domains
- Click 
- Pick your domain- if the domain was made with Route 53, you can pick it from it or just write the domain name.

Start a manual deployment

Manually upload objects to deploy your app. You can choose to drag and drop the artifacts directly, pull a zip from an existing S3 bucket or any other URL.


App name: Branch name:

Method





☐ Drag and drop ☒ Amazon S3 ☐ Any URL


S3 location of objects to host

All apps 1

moriel-profolio		
Deployed ✓		
Prod branch	Last update	
staging	42 minutes ago	

moriel-profolio <<

 **Hosting** ▲

Build notifications


Access control

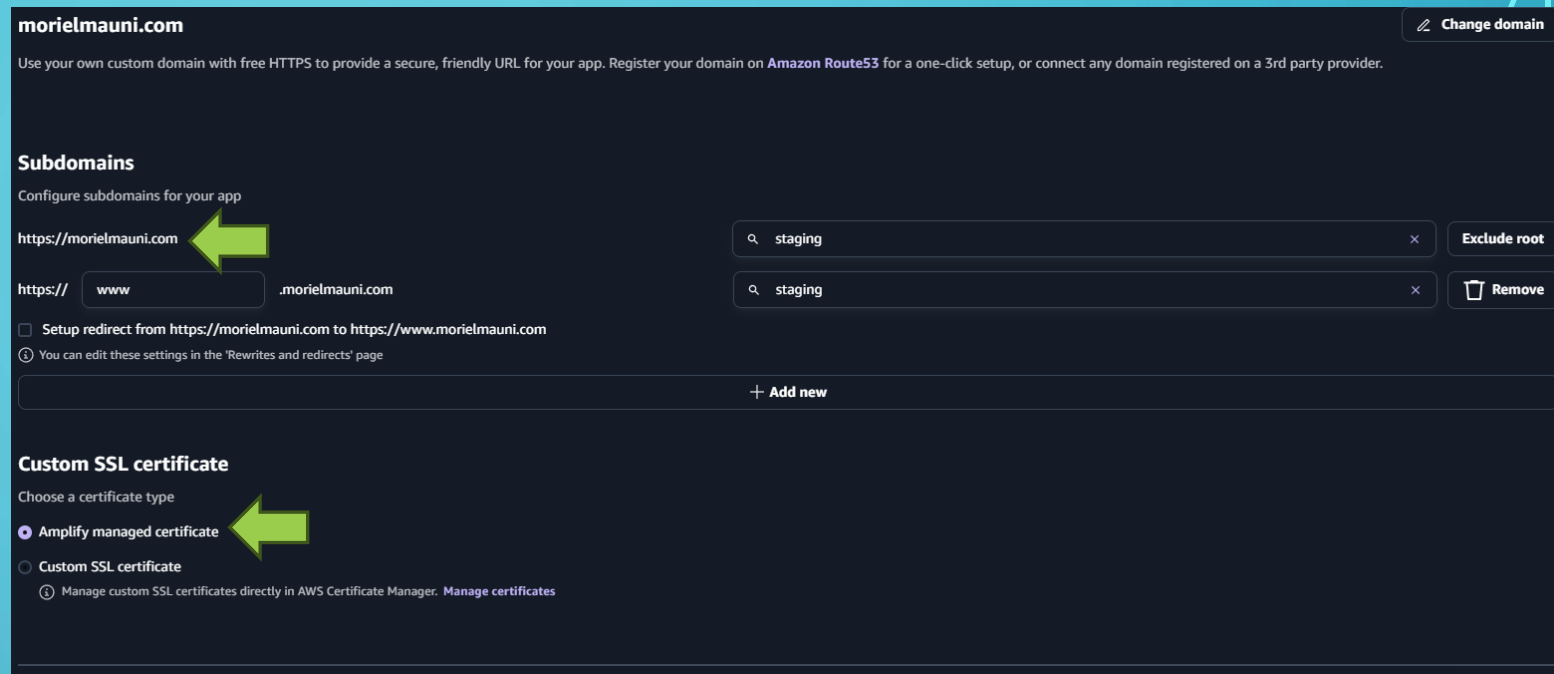
Rewrites and redirects

Custom domains ←

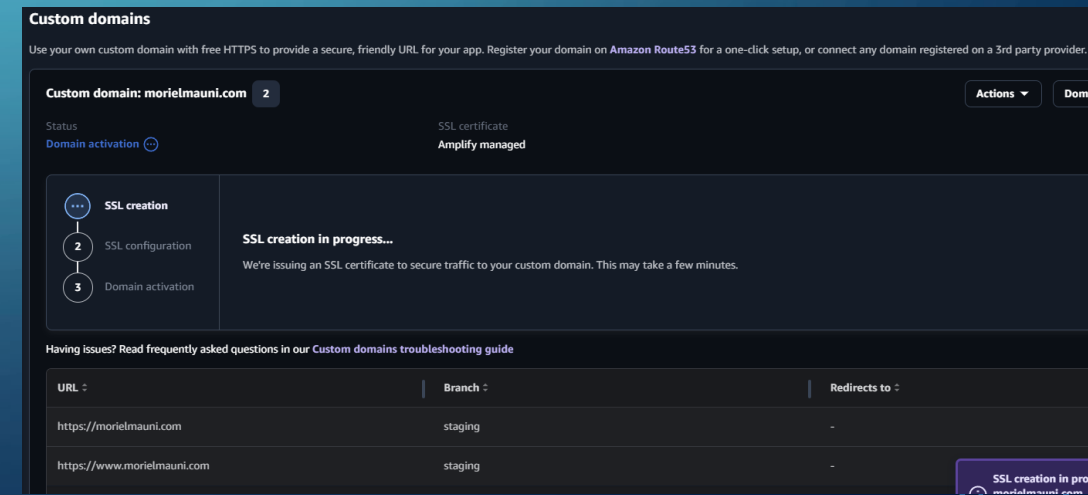
Custom headers

Monitoring

 **App settings** ▼



- The default settings are good most of the time unless you want a custom SSL
- Let AWS Amplify work his charms and get you SSL Certificate
- There are 3 steps, after them you can visit the website with your custom domain.



CONGRATULATIONS!

- We did it! We have a full Website with SSL certification
- Note: if you want to update the website, update the files on s3 bucket then:

staging

Deployed ✓

Domain

<https://morielmauni.com>

Last deployment

1 day ago



Deploy updates

★ Production branch