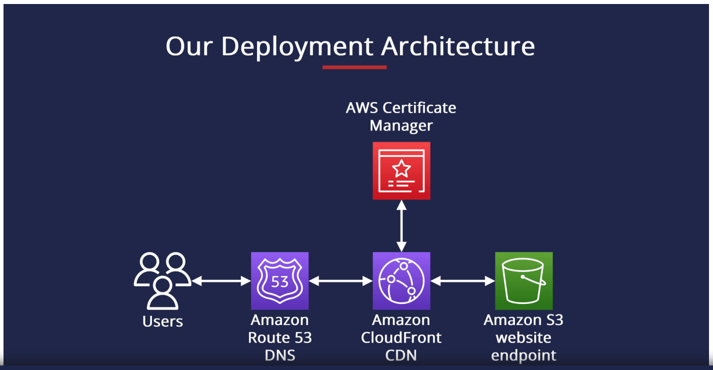
Category Articles

Setup an Amazon CloudFront distribution with SSL, custom domain and S3

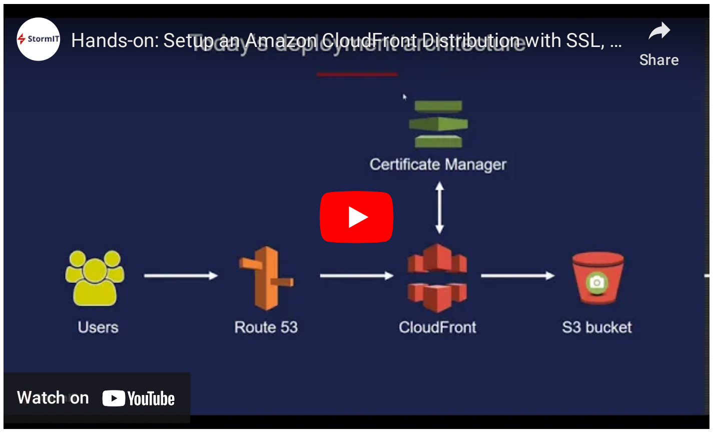


- Create a record set in Route 53
- Set up a CloudFront distribution and link it with a custom domain
- Secure the connection via SSL and AWS Certificate Manager

We will be hosting a static website in an S3 bucket. Then, we will set up a custom domain together with Route 53 DNS. And finally, we will spin up a CloudFront CDN, link it with our custom domain via Route 53, and secure the whole connection using an SSL certificate and AWS Certificate Manager.

As mentioned, you are going to need a custom domain for this particular setup. You can get one online for free for example at Freenom.

If you want, you can follow our instructions in the video below:


Hands-on: Setup an Amazon CloudFront Distribution with SSL, ...
Share
Today's deployment architecture
Certificate Manager
Users
Route 53
CloudFront
S3 bucket
Watch on YouTube

Our workflow today is going to be:

- Set up an S3 bucket
- Set up records in Route 53
- Create a CloudFront CDN distribution
- Create an SSL certificate for HTTPS
- Link the CloudFront CDN, S3, custom domain and SSL certificate via Route 53

S3 Bucket

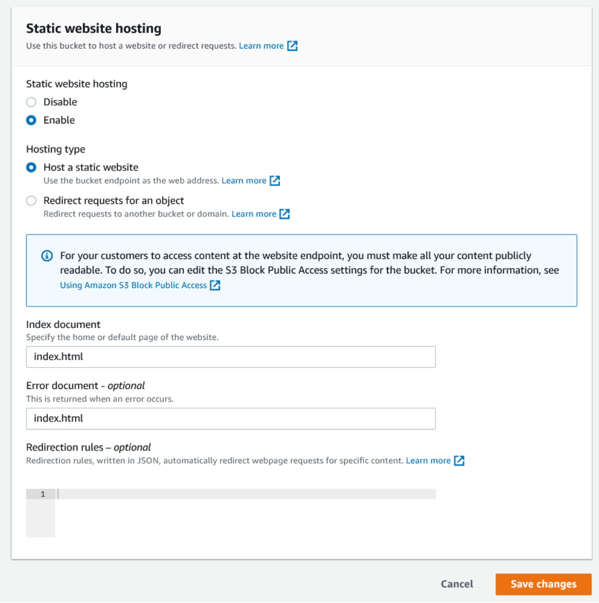
In this section, we are going to create an S3 bucket, open the bucket for public acess, set it up for static website hosting and upload a simple static website to it. Finally we will test that everything is working correctly by trying to view our website from outside of AWS.

First, we are going to create an S3 bucket:

- Go to your AWS console and search for S3. Click it.
- Click 'Create Bucket'. Choose a *globally unique* bucket name and AWS region.
- Untick "Block all public access" and confirm it, since we want our static website accessible from the internet.
- That's it, you don't need anything else in this menu and can click 'Create bucket'.

Then we need to open the bucket to public:

- Open the bucket you just created from the console. The bucket should be empty.
- Click *Properties*, and then sroll down to *Static website hosting* and click 'Edit'.
- Choose 'Enable' under "Static website hosting"
- Choose "Host a static website".
- Choose an Index document and an Error document. You can fill in "index.html" into both.
- Hit *Save changes*.



Static website hosting

Use this bucket to host a website or redirect requests. Learn more

Static website hosting

○ Disable

● Enable

Hosting type

● Host a static website

Use the bucket endpoint as the web address. Learn more

○ Redirect requests for an object

Redirect requests to another bucket or domain. Learn more

For your customers to access content at the website endpoint, you must make all your content publicly readable. To do so, you can edit the S3 Block Public Access settings for the bucket. For more information, see Using Amazon S3 Block Public Access

Index document

Specify the home or default page of the website.

index.html

Error document - *optional*

This is returned when an error occurs.

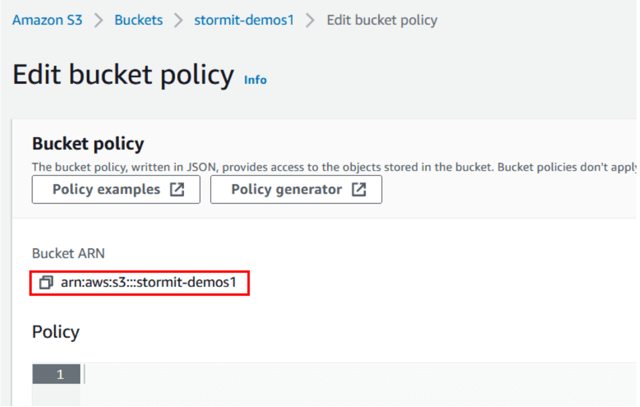
index.html

Redirection rules – *optional*

Redirection rules, written in JSON, automatically redirect webpage requests for specific content. Learn more

Finally, we have to edit the Bucket Policies.

- In S3, o over to *Permissions.* Then *Bucket Policy.*
- Here you need to fill in a policy which will allow the bucket to be publicly accessed. I recommend using the official AWS policy generator for these purposes, which can be found at: AWS Policy Generator
- At the generator, you need to choose *S3 bucket policy* as the Type of Policy. Principal will be "*" (without quotes), and Actions will be "GetObject".
- Fill in your ARN, which can be found in AWS Console where you were about to fill in the bucket policy. There, at the top of the page you should be able to find the ARN in the format of *arn:aws:s3:::<bucket_name>.*



Amazon S3 > Buckets > stormit-demos1 > Edit bucket policy

Edit bucket policy Info

Bucket policy

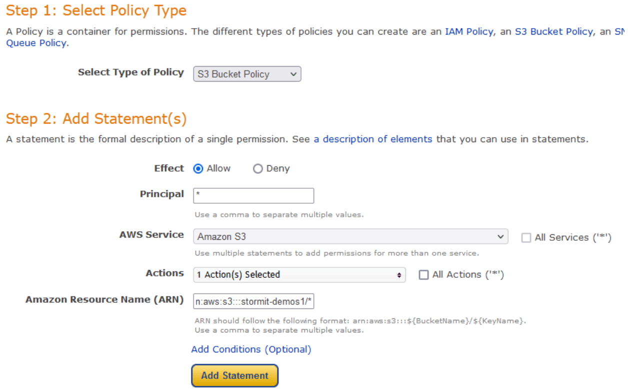
The bucket policy, written in JSON, provides access to the objects stored in the bucket. Bucket policies don't appl

Bucket ARN

arn:aws:s3:::stormit-demos1

Policy

- **Important:** before you paste in the ARN, append a slash and asterisk at the end, so it will look something like: arn:aws:s3:::stormit-demos/* (my bucket name is “stormit-demos1”). This will allow access to *everything* inside the bucket. Click *Add Statement.*



Step 1: Select Policy Type

A Policy is a container for permissions. The different types of policies you can create are an IAM Policy, an S3 Bucket Policy, an SN Queue Policy.

Select Type of Policy: S3 Bucket Policy

Step 2: Add Statement(s)

A statement is the formal description of a single permission. See a description of elements that you can use in statements.

Effect: ● Allow ○ Deny

Principal: *

Use a comma to separate multiple values.

AWS Service: Amazon S3 □ All Services ('*')

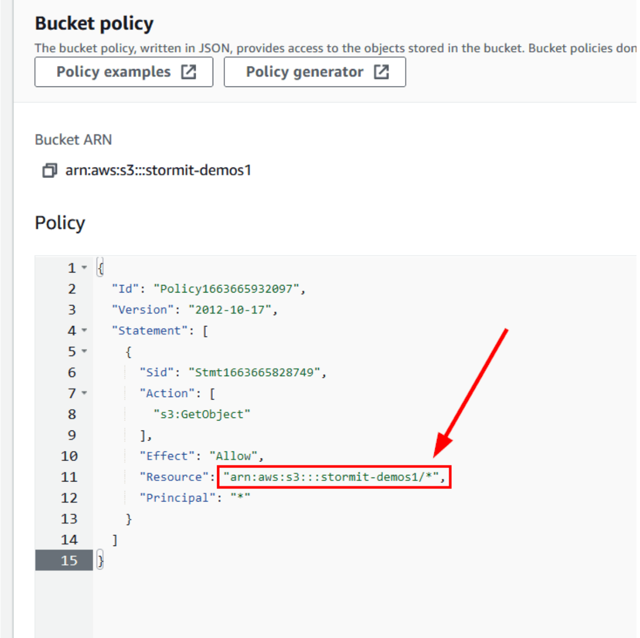
Use multiple statements to add permissions for more than one service.

Actions: 1 Action(s) Selected □ All Actions ('*')

Amazon Resource Name (ARN): n:aws:s3:::stormit-demos1/*

ARN should follow the following format: arn:aws:s3:::${BucketName}/${KeyName}.
Use a comma to separate multiple values.

- Finally, click *Generate policy.*
- Copy the whole policy over to the S3 AWS console



Bucket policy

The bucket policy, written in JSON, provides access to the objects stored in the bucket. Bucket policies don

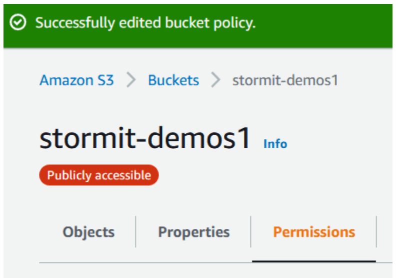
Bucket ARN

arn:aws:s3:::stormit-demos1

Policy

```
{
  "Id": "Policy1663665932097",
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Stmt1663665828749",
      "Action": [
        "s3:GetObject"
      ],
      "Effect": "Allow",
      "Resource": "arn:aws:s3:::stormit-demos1/*",
      "Principal": "*"
    }
  ]
}
```

That’s it for our S3 Bucket setup. When you open your bucket, you should see a red bubble with white text *Publicly accessible* under the name of your bucket. If you do, everything is correct and we can proceed.


Successfully edited bucket policy.

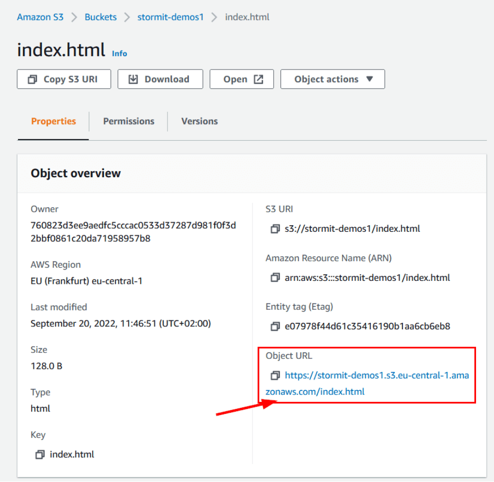
Amazon S3 > Buckets > stormit-demos1

stormit-demos1 Info

Publicly accessible

Next, we're going to upload dummy website files into the bucket & test access to it:

- Download a prepared static website. You can use some other website if you'd like, but keep in mind that the next steps might be a bit different for you if you do.
- In your AWS console inside your S3 Bucket hit *Upload*, then *Add files* and upload your index.html file. Hit *Upload*.
- Click *Create folder* and create a folder called "images". Do not use any encryption (default).
- Navigate into the images folder you just created and upload the provided GIF image, same as when we uploaded the index.html.
- Once both files are uploaded, click on index.html within the S3 bucket and locate "Object URL". Click it. You should see a GIF of a puppy playing with a pillow (assuming you used the provided files).


Amazon S3 > Buckets > stormit-demos1 > index.html

index.html Info

Object overview

Owner
760823d3ee9aedfc5cccac0533d37287d981f0f3d2bbf0861c20da71958957b8

AWS Region
EU (Frankfurt) eu-central-1

Last modified
September 20, 2022, 11:46:51 (UTC+02:00)

Size
128.0 B

Type
html

Key
index.html

S3 URI
s3://stormit-demos1/index.html

Amazon Resource Name (ARN)
arn:aws:s3:::stormit-demos1/index.html

Entity tag (Etag)
e07978f44d61c35416190b1aa6cb6eb8

Object URL
https://stormit-demos1.s3.eu-central-1.amazonaws.com/index.html

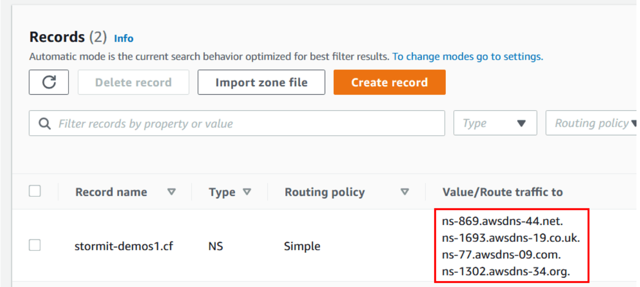
Congratulations – you've just set up an S3 bucket, uploaded a website and opened it for public access. Next, we're going to set up routing in Route 53.

Learn in our blog post how the S3 Intelligent-Tiering storage class can help you deliver automatic cost savings by moving data between access tiers: Amazon S3 Intelligent-Tiering: How it Helps to Optimize Storage Costs

Route 53 (first time around):

Create hosted zone in Route 53:

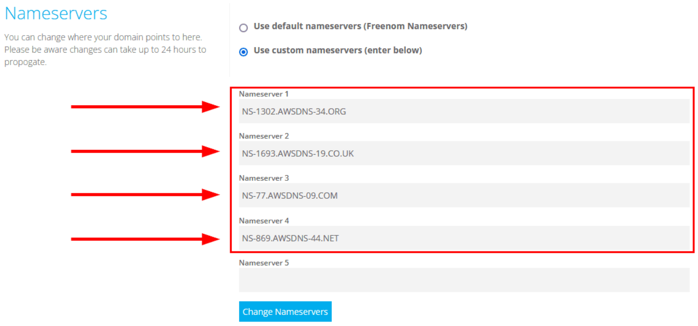
- In your AWS Console search for Route 53 under Services.
- Under 'DNS management', click *Create hosted zone.*
- Inside the 'Domain name' field input your domain name. You can obtain one for free from Freenom.
- 'Type' will be *Public hosted zone.*
- Click *Create hosted zone.*
- Now we need to link our domain with the records in Route 53.
- Note the values as marked on the picture. These are nameservers which we need to link with our domain, basically telling our domain "when someone tries to reach you, you redirect them to one of these servers".


Records (2) Info

Automatic mode is the current search behavior optimized for best filter results. To change modes go to settings.

	Record name	Type	Routing policy	Value/Route traffic to
□	stormit-demos1.cf	NS	Simple	ns-869.awsdns-44.net. ns-1693.awsdns-19.co.uk. ns-77.awsdns-09.com. ns-1302.awsdns-34.org.

- Open a new tab in your browser (leave the Route 53 page with the nameservers open) and go to your domain management page, in my case freenom.com. Log in, open the list of your domains and click Manage Domain. Under Management Tools click Nameservers. You should see 5 text fields named Nameserver 1-5.
- Go back to your Route 53 nameserver page and copy the URLs one by one to the fields in your domain management page. **Important:** copy the URLS over without the dot at the end. So for example you'd copy "ns-869.awsdns-44.net." as "ns-869.awsdns-44.net" (without the dot at the end).


Nameservers

You can change where your domain points to here. Please be aware changes can take up to 24 hours to propogate.

○ Use default nameservers (Freenom Nameservers)

● Use custom nameservers (enter below)

Nameserver 1: NS-1302.AWSDNS-34.ORG

Nameserver 2: NS-1693.AWSDNS-19.CO.UK

Nameserver 3: NS-77.AWSDNS-09.COM

Nameserver 4: NS-869.AWSDNS-44.NET

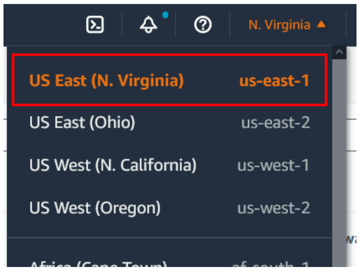
Nameserver 5:

- Once you've copied everything, confirm by clicking *Change Nameservers.*
- That's it for Route 53 for now, but we will return to it later.

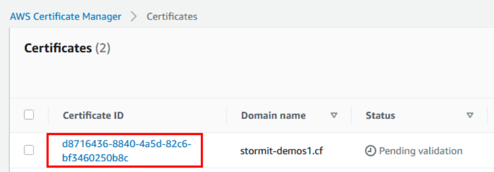
Now we want to set up an SSL certificate via the AWS Certificate Manager in order to secure our CloudFront custom domain with SSL.

AWS Certificate Manager

- In your AWS Console search for Certificate Manager and click the link.
- Important: You have to select the AWS North Virginia region(US-East-1). CloudFront recognizes only this region as it's ACM certificates.


N. Virginia
US East (N. Virginia) us-east-1
US East (Ohio) us-east-2
US West (N. California) us-west-1
US West (Oregon) us-west-2
Africa (Cape Town) af-south-1

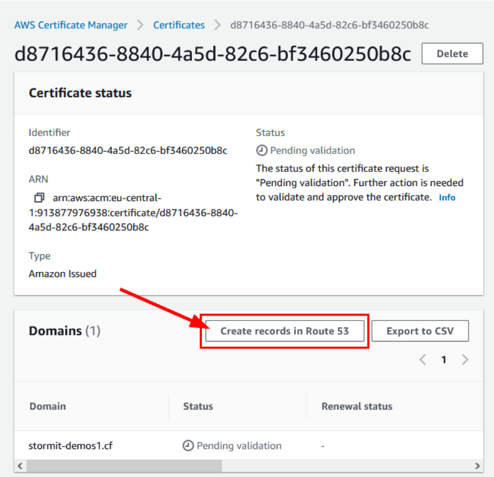
- Click *Get started* under Provision certificates
- Click *Request a public certificate* and then *Request a certificate.*
- Fill in the Domain name (in my case it is stormit-demos1.cf), click *Next.*
- Choose a method of validation, I recommend DNS validation.
- We do not need anything else, so continue clicking *Next* until you can *Confirm and request* the certificate.
- On the next screen you'll see that the certificate status is "Pending validation". In order for it to get validated we need to link it with our Route 53 records, and we can do that by clicking on the "Certificate ID".



Certificates (2)

☐	Certificate ID	Domain name	Status
☐	d8716436-8840-4a5d-82c6-bf3460250b8c	stormit-demos1.cf	Pending validation

- Next click on *Create record in Route 53* button under *Domains*.


AWS Certificate Manager > Certificates > d8716436-8840-4a5d-82c6-bf3460250b8c
Delete
Create records in Route 53
Export to CSV
< 1 >

d8716436-8840-4a5d-82c6-bf3460250b8c

Certificate status

Identifier
d8716436-8840-4a5d-82c6-bf3460250b8c

ARN
arn:aws:acm:eu-central-1:913877976938:certificate/d8716436-8840-4a5d-82c6-bf3460250b8c

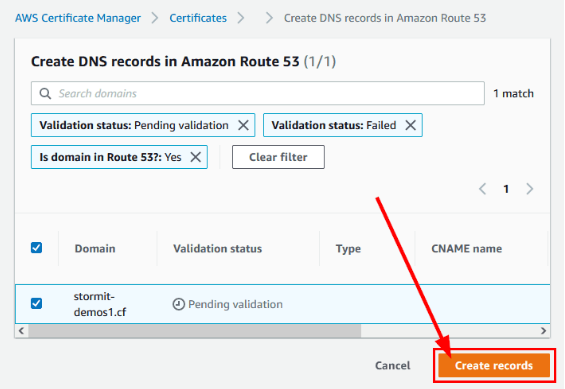
Type
Amazon Issued

Status
Pending validation
The status of this certificate request is "Pending validation". Further action is needed to validate and approve the certificate. Info

Domains (1)

Domain	Status	Renewal status
stormit-demos1.cf	Pending validation	-

- Then click *Create records.*


AWS Certificate Manager > Certificates > > Create DNS records in Amazon Route 53
Search domains
Clear filter
< 1 >
Cancel
Create records

Create DNS records in Amazon Route 53 (1/1)

1 match

Validation status: Pending validation ✕ | Validation status: Failed ✕ | Is domain in Route 53?: Yes ✕

☑	Domain	Validation status	Type	CNAME name
☑	stormit-demos1.cf	Pending validation		

- That's it for the Certificate Manager. It can take up to 30 minutes for the certificate to be issued so try to be patient.

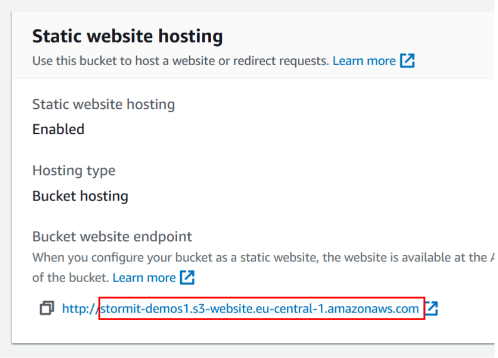
After the certificate is issued, we can set up the last element in our deployment, which is a CloudFront distribution.

CloudFront Ditribution

In this section, we are going to spin up a CloudFront CDN distribution with an S3 bucket so that when anyone tries to access our distribution, they will actually be accessing our S3 bucket while taking advantage of the speed of CloudFront.

Create CloudFront distribution

- In your AWS Console search for CloudFront and click the link.
- Click *Create Distribution*. On the next page click *Get Started* in the Web section, since we will be wanting a web distribution.
- You should see an **Origin settings** page with many options and fields.
- For the **Origin Domain Name**, you will need to go back to your S3 management console (Services -> S3), but leave the CloudFront setup page open. Once in the S3 management console, click on your bucket name, then click on *Properties* and scroll down to *Static website hosting*. Copy over the highlighted part of the Endpoint URL:



Static website hosting

Use this bucket to host a website or redirect requests. Learn more

Static website hosting

Enabled

Hosting type

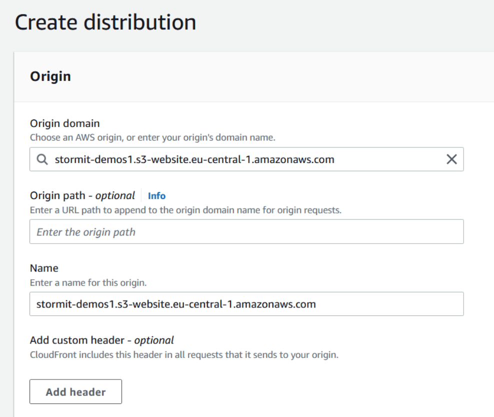
Bucket hosting

Bucket website endpoint

When you configure your bucket as a static website, the website is available at the /
of the bucket. Learn more

http://stormit-demos1.s3-website.eu-central-1.amazonaws.com

- Go back to the CloudFront setup page and paste the URL inside the *Origin Domain Name* field. You may notice that AWS already offers you an S3 option from the drop down list, but that wouldn't point to the actual website.



Create distribution

Origin

Origin domain

Choose an AWS origin, or enter your origin's domain name.

stormit-demos1.s3-website.eu-central-1.amazonaws.com

Origin path - *optional* Info

Enter a URL path to append to the origin domain name for origin requests.

Enter the origin path

Name

Enter a name for this origin.

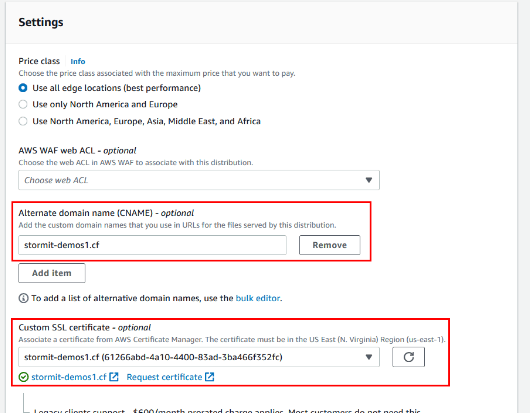
stormit-demos1.s3-website.eu-central-1.amazonaws.com

Add custom header - *optional*

CloudFront includes this header in all requests that it sends to your origin.

CloudFront SSL certificate:

- Under Viewer Protocol Policy choose *Redirect HTTP to HTTPS*
- Now scroll all the way down to the Alternate Domain Names (CNAMEs) field and type in your domain name without http(s), i.e. example_domain.com
- Under **Settings**, choose *Custom SSL Certificate* for CloudFront SSL encryption. Click into the field below - the SSL certificate you provisioned earlier should be offered to you. Choose it.



Settings

Price class Info

Choose the price class associated with the maximum price that you want to pay.

● Use all edge locations (best performance)

○ Use only North America and Europe

○ Use North America, Europe, Asia, Middle East, and Africa

AWS WAF web ACL - *optional*

Choose the web ACL in AWS WAF to associate with this distribution.

Choose web ACL

Alternate domain name (CNAME) - *optional*

Add the custom domain names that you use in URLs for the files served by this distribution.

stormit-demos1.cf

To add a list of alternative domain names, use the bulk editor.

Custom SSL certificate - *optional*

Associate a certificate from AWS Certificate Manager. The certificate must be in the US East (N. Virginia) Region (us-east-1).

stormit-demos1.cf (61266abd-4a10-4400-83ad-3ba466f352fc)

✓ stormit-demos1.cf Request certificate

Legacy clients support - $600/month prorated charge applies. Most customers do not need this

- That's it. Click *Create Distribution* at the very bottom of the page.
- The distribution takes about 10-20 minutes to provision. Once it is provisioned (Status = Deployed, State = Enabled), you can test it by pasting the URL under Domain Name into your browser. You should see the website you uploaded into S3.

We are almost done, we have S3, CloudFront SSL certificate, but there's one more step we need to take. If you go ahead and try to load your domain name in your web browser, you'll notice that the page can't be loaded. That is because all we have to do now is to create a record in Route 53 and basically tell it to direct traffic from your domain to your CloudFront distribution.

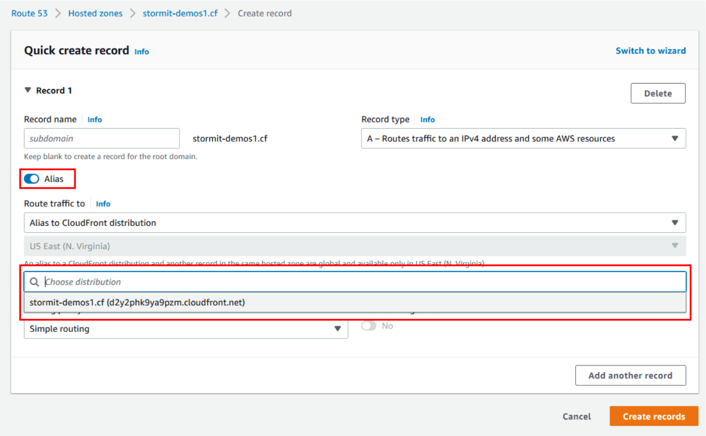
StormIT team has a proven track record of experience and delivered projects with Amazon CloudFront and is proud to be designated as the AWS Service Delivery Partner for the Amazon CloudFront CDN service.

We provide special pricing for organizations transferring as little as 1 TB of data per month and have saved companies up to 60% on their CDN data transfer-out costs.

ESTIMATE SAVINGS

Route 53 (revisited)

- Click *Services* and type in Route 53, click the link. Click the name of your hosted zone.
- Click *Create record.* Choose *Simple routing,* click *Next* and then *Define record.*
- Click the dropdown menu at Value/Route traffic to and choose *Alias to CloudFront distribution.*
- Choose the distribution you provisioned previously.
- Notice that the only available region is US East (N. Virgina)[us-east-1]. This is due to the fact that we provisioned an SSL Certificate via AWS Certificate manager. This service is only available in US East 1.
- Click *Create Records.*



Route 53 > Hosted zones > stormit-demos1.cf > Create record

Quick create record Info

Switch to wizard

▼ Record 1

Delete

Record name Info: subdomain stormit-demos1.cf

Keep blank to create a record for the root domain.

Record type Info: A – Routes traffic to an IPv4 address and some AWS resources

Alias

Route traffic to Info: Alias to CloudFront distribution

US East (N. Virginia)

An alias to a CloudFront distribution and another record in the same hosted zone are global and available only in US East (N. Virginia).

Choose distribution

stormit-demos1.cf (d2y2phk9ya9pzm.cloudfront.net)

Simple routing

No

Add another record

Cancel

Create records

And that's it. All you have to do now is wait a few minutes for the DNS records to update and try to load your domain name. If everything went correctly, you should see your static website distributed via CloudFront.

If not, please try to retrace your steps following this guide or have a look at our Video Guide.

Hope you enjoyed this guide and happy clouding!

Are you ready to accelerate your business to the cloud?

CONTACT US

Author

Ivo Fibiger, AWS Solutions Architect

Ivo Fibiger, AWS Solutions Architect

Certified AWS Solution Architect, with experience in BigData and analytics projects with strong attention to detail.

Do you like this article? Share it with your colleagues and friends

Similar blog posts

SEE ALL POSTS

Category Articles

Scalability in Cloud Computing: Horizontal vs. Vertical Scaling

ng

Look deeper into horizontal and vertical scaling and also into AWS scalability and which services you can use.

FIND OUT MORE

Category Articles

Does Putting CloudFront in Front of API Gateway Make Sense?

Learn about API Gateway endpoint types and the difference between Edge-optimized API gateway and API Gateway with CloudFront distribution.

FIND OUT MORE

Category News

StormIT Achieves AWS Service Delivery Designation for AWS WAF

StormIT is excited to announce that we have received AWS Web Application Firewall (WAF) Service Delivery designation.

FIND OUT MORE

C

W
W

Stc
Clc
grc

FII

Are you ready to accelerate your business to the cloud?

Our team of certified Amazon Web Services consultants is ready to handle your next cloud project. Get in touch today to speak with a cloud expert and discuss how we can help.

✉ info@stormit.cloud

☎ +420 277 771 771 PRAGUE, CZ

Name *

E-mail *

Message *

□ I accept terms & conditions

Send

Services

About StormIT

Luxembourg

Prague

Follow us