CLOUD CONCEPTS

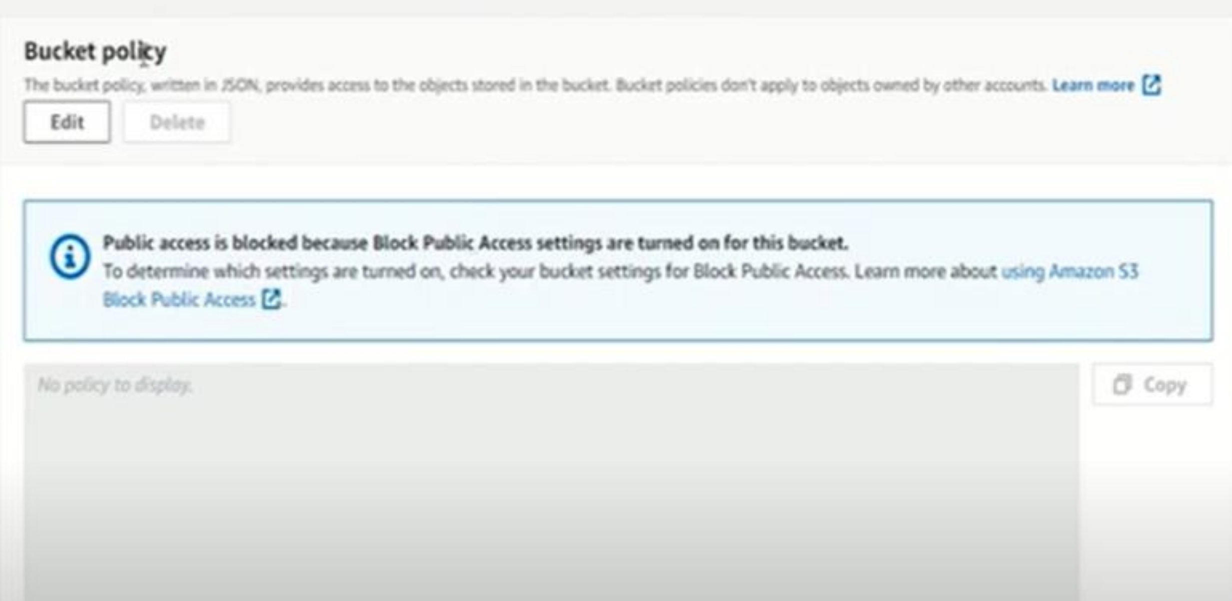
(AWS Core Services – Hands On)

1. AWS Storage

- ✓ S3 - Object Store
- ✓ EBS - Block Store
- ✓ EFS - File Store
- ▪ Data Backup - Central Service for backing up data.
- ▪ Data Transfer - On premise to cloud and vice versa.

Task 1: Managing S3 object access through policies.

1. Create a bucket in S3 and upload a single object in the bucket.
   a. Destination = standard
   b. Bucket versioning = disabled
   c. Encryption = disabled
   d. ACL = keep the default
2. Manage one IAM account -> block the object access to that account default public access is disabled.
   - **Set the bucket policies: Click on bucket -> Bucket Permissions -> scroll down -> bucket policies -> click edit and click policy generator.**

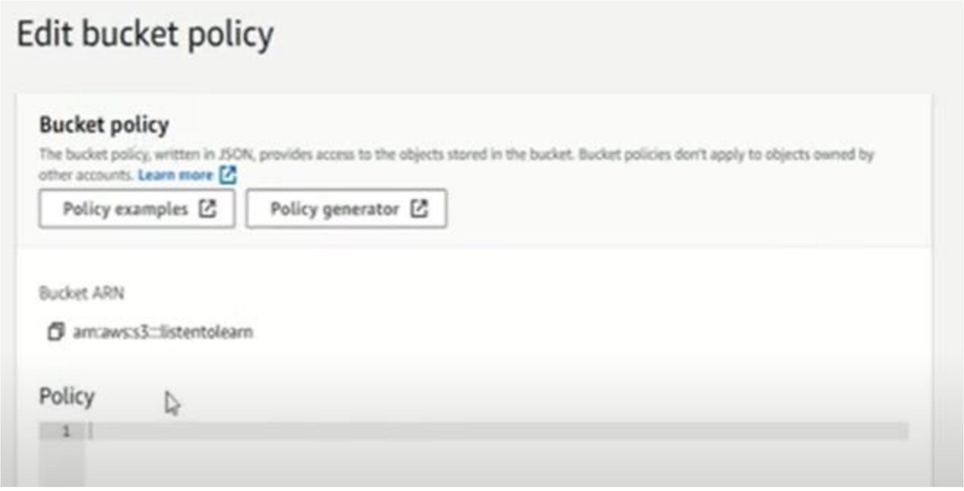


Bucket policy

The bucket policy, written in JSON, provides access to the objects stored in the bucket. Bucket policies don't apply to objects owned by other accounts. Learn more

Public access is blocked because Block Public Access settings are turned on for this bucket.
To determine which settings are turned on, check your bucket settings for Block Public Access. Learn more about using Amazon S3 Block Public Access.

No policy to display.



Edit bucket policy

Bucket policy

The bucket policy, written in JSON, provides access to the objects stored in the bucket. Bucket policies don't apply to objects owned by other accounts. Learn more

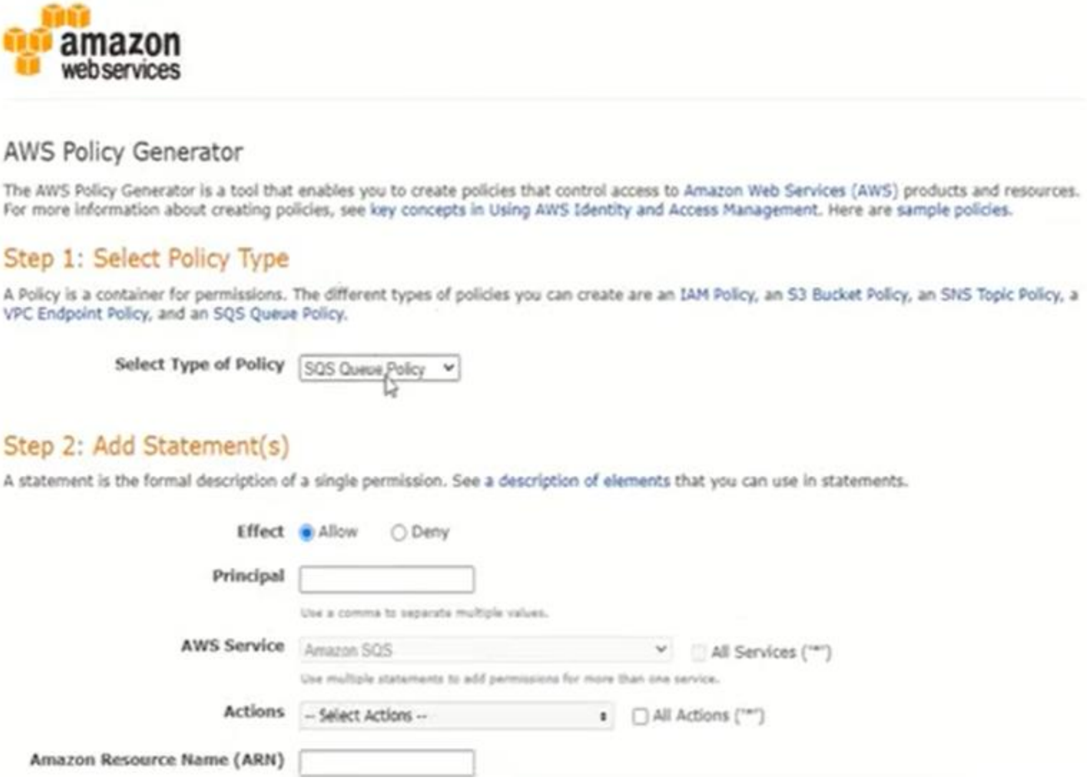
Bucket ARN

arn:aws:s3:::listentolearn

Policy

```
1 |
```

- It’s a JSON generator for policies. (who can or cannot access the object.)
    a. Policy type = S3
    b. Effect = deny
    c. Principal = IAM user ARN
    d. AWS Service = S3
    e. Actions = Get Object
    f. Amazon Resource Name = ARN of the object (copy ARN of the bucket append it with object key)



AWS Policy Generator

The AWS Policy Generator is a tool that enables you to create policies that control access to Amazon Web Services (AWS) products and resources. For more information about creating policies, see key concepts in Using AWS Identity and Access Management. Here are sample policies.

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 SNS Topic Policy, a VPC Endpoint Policy, and an SQS Queue Policy.

Select Type of Policy: SQS Queue 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.

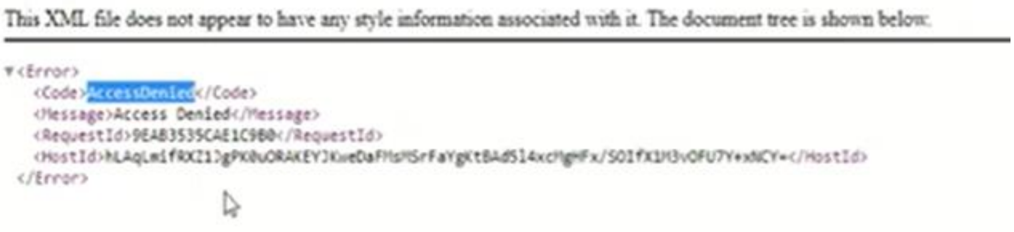
AWS Service: Amazon SQS □ All Services ("*")

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

Actions: -- Select Actions -- □ All Actions ("*")

Amazon Resource Name (ARN)

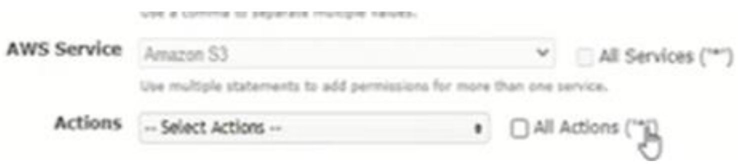
3. Click add statement and Generate Policy.
4. Copy the JSON from the JSON document.
5. Paste it in S3 Console Policy and Save it.
6. Output must be as shown below.



This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
▼<Error>
   <Code>AccessDenied</Code>
   <Message>Access Denied</Message>
   <RequestId>9EAB3535CAE1C9B0</RequestId>
   <HostId>hLAqLmifRXZ1JgPK0uORAKEYJKueDaFMsMSrFaYgKtBAdS14xcMgHFx/SO1fX1M3vOFU7Y+xNCY=</HostId>
 </Error>
```

- 💣 You can also allow or deny action in a single statement to all services.



Use a comma to separate multiple values.

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

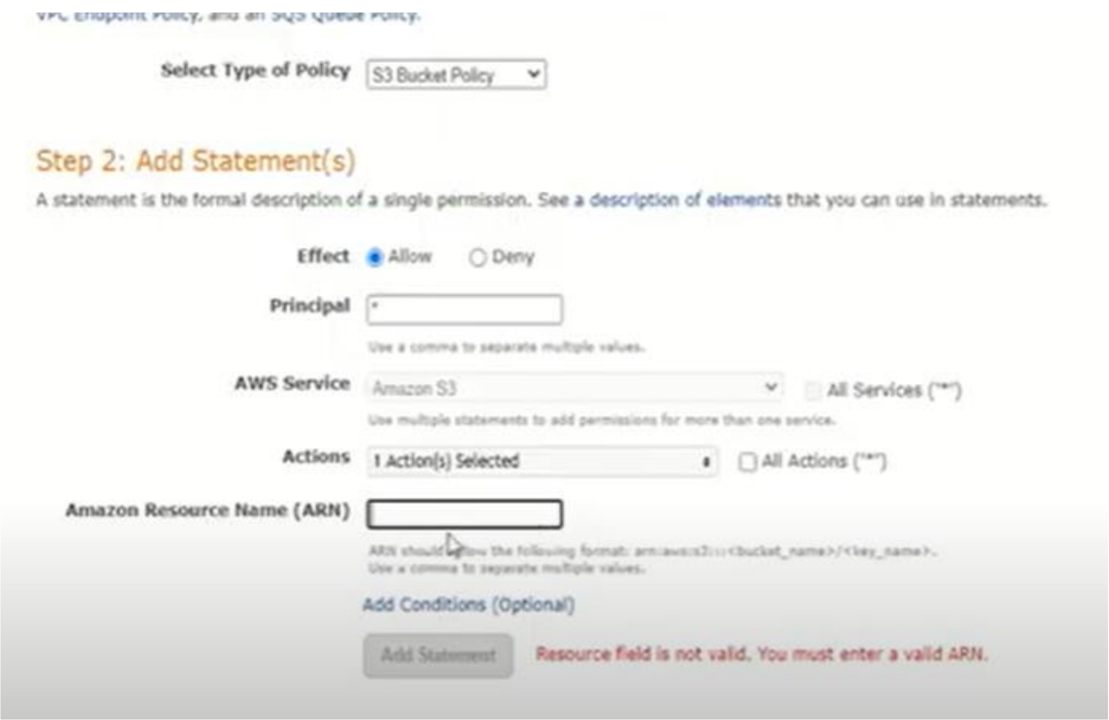
Use multiple statements to add permissions for more than one service.

Actions: -- Select Actions -- ☐ All Actions ('*')

Task 2: Static Web Hosting using S3.

Static Website is created using CSS and JavaScript and display content is same for everyone without any server-side processing or databases involved while Dynamic Website involves server-side processing and databases involved (EC2 & RDS).

1. Create a Bucket and upload files (index.html, yourfilename.html, error.html) in the bucket and click upload.
2. Make your bucket public for public access. It can be done through all three ways listed.
   - ✓ Set the permission at object level for all objects.
   - ✓ Set bucket public.
   - ✓ Set permission at account level.
3. For simplicity make the bucket public, make the object public by creating bucket policy. Paste the JSON of generated policy in policy of bucket.
   - **ARN or resource would be bucket ARN and append it with * for all objects.**



VPC Endpoint Policy, and an SQS 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.

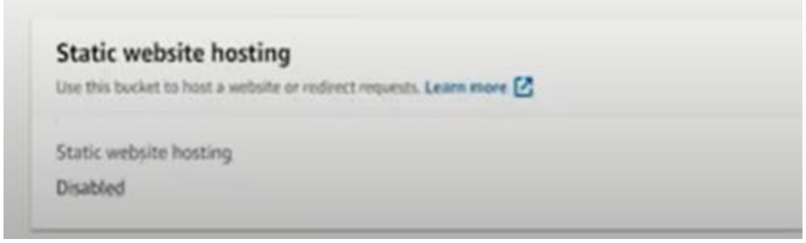
Actions: 1 Action(s) Selected □ All Actions ("*")

Amazon Resource Name (ARN):

ARN should follow the following format: arn:aws:s3:::<bucket_name>/<key_name>.
Use a comma to separate multiple values.

Resource field is not valid. You must enter a valid ARN.

4. The bucket and objects Access = PUBLIC.
5. Now create the website.
    a. Go to properties and there will be an option of static web hosting. Click edit and enable it.

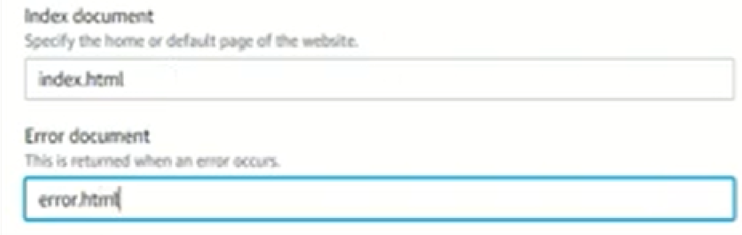


Static website hosting

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

Static website hosting: Disabled

    b. Specify the documents and save the settings.



Index document

Specify the home or default page of the website.

index.html

Error document

This is returned when an error occurs.

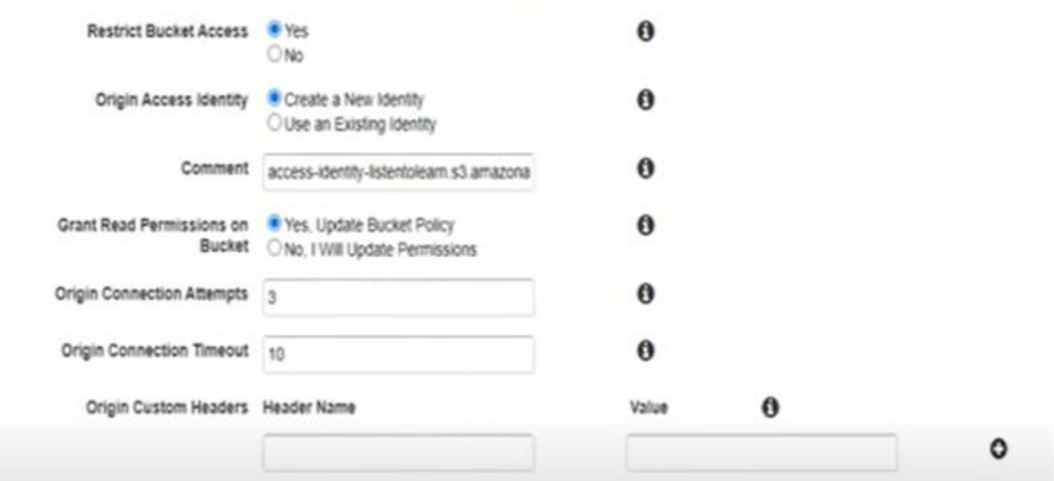
error.html

    c. After saving the changes the URL is generated for your static website.

Task 3: Web Hosting using Cloud Front and S3 as origin.

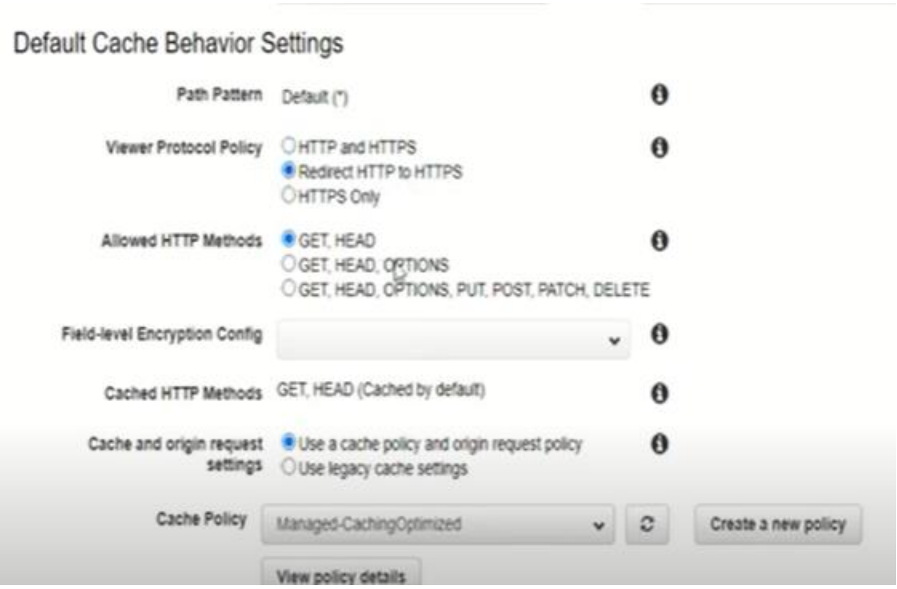
CDN is used to deliver content to the edge users from origin (Origin from where content originates can be S3, EC2 or Load Balancer). CDN provides cost effective and secure global edge network for caching the content that reduces latency for user base making the application readily available.

1. Create the bucket with some objects in any region, enable the versioning and let the other parameters as default. Objects of the bucket are not public. Add the html files in bucket that are going to be served through CDN. The file should not be accessible as permissions are not set.
2. Create a distribution for hosting the website through CloudFront.
    a. Origin Domain = S3
    b. Origin Path is blank as the content is placed on root.
    c. Sheild can be enabled if you want to add one more layer to access the data from S3.
    d. Restrict bucket access so that buckets will not be accessed directly but through CloudFront. An access identity will be created.
    e. Ask CloudFront to create bucket policy.
    f. Keep the rest defaults.



Restrict Bucket Access	● Yes ○ No	
Origin Access Identity	● Create a New Identity ○ Use an Existing Identity	
Comment	access-identity-listentolearn.s3.amazona	
Grant Read Permissions on Bucket	● Yes, Update Bucket Policy ○ No, I Will Update Permissions	
Origin Connection Attempts	3	
Origin Connection Timeout	10	
Origin Custom Headers	Header Name	Value

    g. Set the viewer protocol = http to https (recommended)



Default Cache Behavior Settings

Path Pattern: Default (*)

Viewer Protocol Policy:
○ HTTP and HTTPS
● Redirect HTTP to HTTPS
○ HTTPS Only

Allowed HTTP Methods:
● GET, HEAD
○ GET, HEAD, OPTIONS
○ GET, HEAD, OPTIONS, PUT, POST, PATCH, DELETE

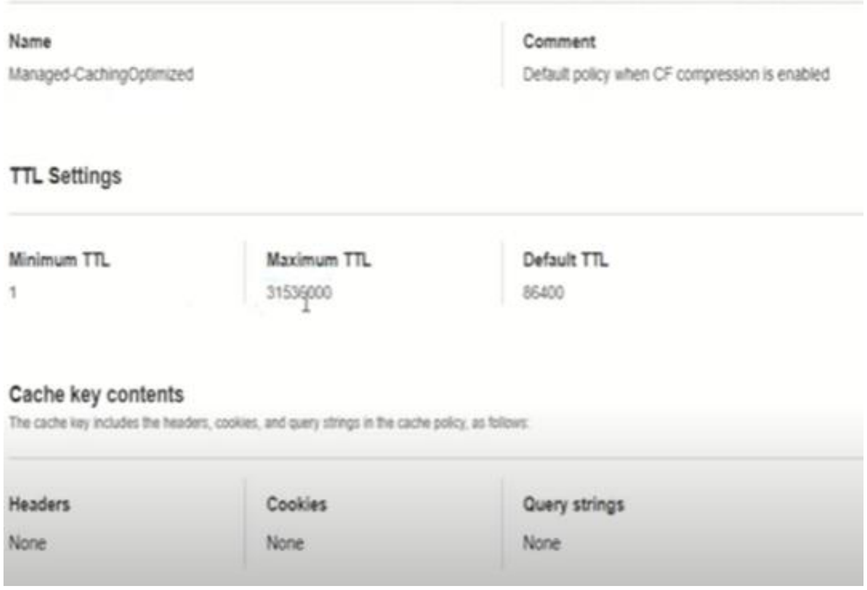
Field-level Encryption Config:

Cached HTTP Methods: GET, HEAD (Cached by default)

Cache and origin request settings:
● Use a cache policy and origin request policy
○ Use legacy cache settings

Cache Policy: Managed-CachingOptimized

h. Attach Cache policies. AWS provides default cache policies that mainly manage TTL settings. (How long the cache will hold the content before considering it as a cache miss and redirecting the request to the origin for actual content.) You can create your own policies.



Name	Comment
Managed-CachingOptimized	Default policy when CF compression is enabled

TTL Settings

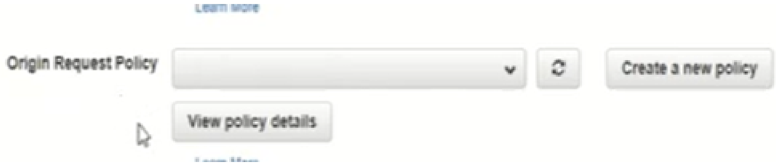
Minimum TTL	Maximum TTL	Default TTL
1	31536000	86400

Cache key contents

The cache key includes the headers, cookies, and query strings in the cache policy, as follows:

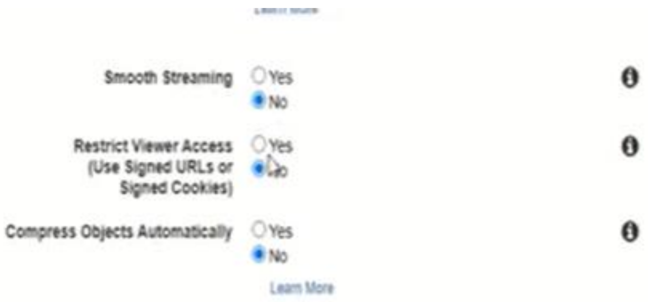
Headers	Cookies	Query strings
None	None	None

i. Origin request policies will allow you to add headers or query string to the parameters that are required to be passed to origin.



Origin Request Policy:

j. Smooth streaming options are for video content to be distributed with any restrictions if applied to viewers or URL.

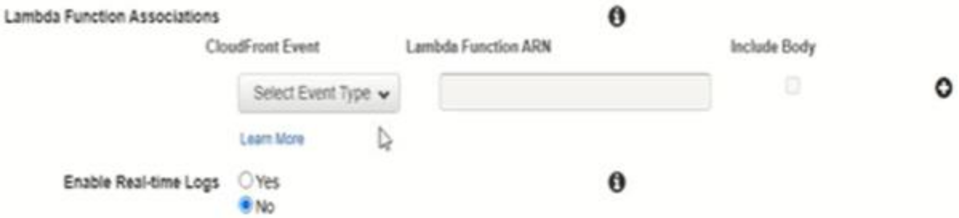


Smooth Streaming ○ Yes ● No

Restrict Viewer Access (Use Signed URLs or Signed Cookies) ○ Yes ● No

Compress Objects Automatically ○ Yes ● No

k. Lambda function can be attached

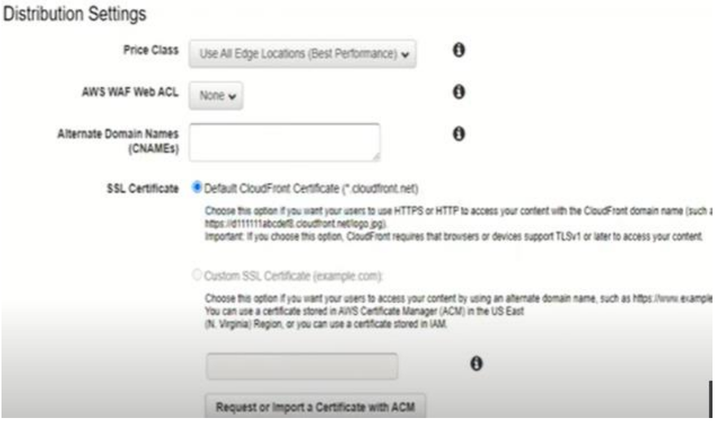


Lambda Function Associations

CloudFront Event	Lambda Function ARN	Include Body
Select Event Type		□

Enable Real-time Logs ○ Yes ● No

l. You can specify the edge location depending upon the user base.



Distribution Settings

Price Class: Use All Edge Locations (Best Performance)

AWS WAF Web ACL: None

Alternate Domain Names (CNAMEs):

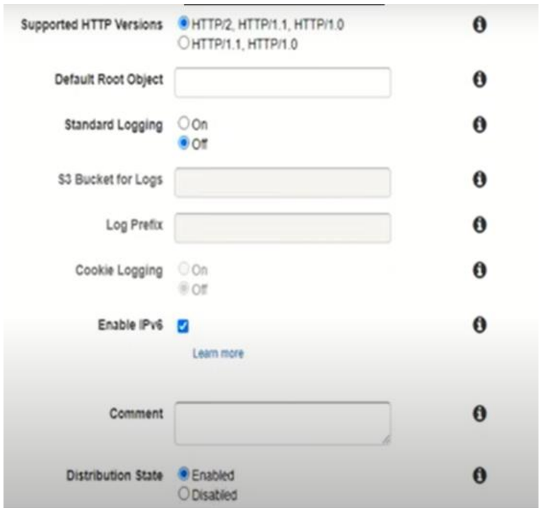
SSL Certificate: ● Default CloudFront Certificate (*.cloudfront.net)

Choose this option if you want your users to use HTTPS or HTTP to access your content with the CloudFront domain name (such a
https://d111111abcdef8.cloudfront.net/logo.jpg).
Important: If you choose this option, CloudFront requires that browsers or devices support TLSv1 or later to access your content.

○ Custom SSL Certificate (example.com):

Choose this option if you want your users to access your content by using an alternate domain name, such as https://www.example
You can use a certificate stored in AWS Certificate Manager (ACM) in the US East
(N. Virginia) Region, or you can use a certificate stored in IAM.

m. Alternate domain names can also be applied.



Supported HTTP Versions ● HTTP/2, HTTP/1.1, HTTP/1.0 ○ HTTP/1.1, HTTP/1.0

Default Root Object:

Standard Logging ○ On ● Off

S3 Bucket for Logs:

Log Prefix:

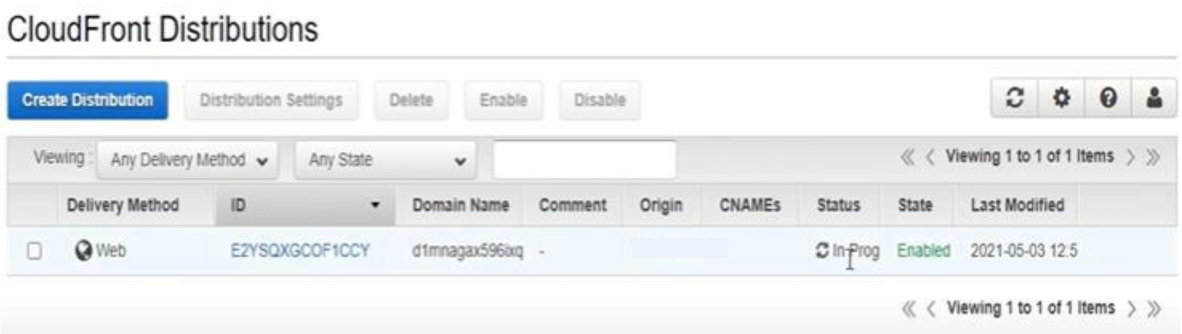
Cookie Logging ○ On ● Off

Enable IPv6 ☑

Comment:

Distribution State ● Enabled ○ Disabled

n. Create the distribution and see the distribution when created.



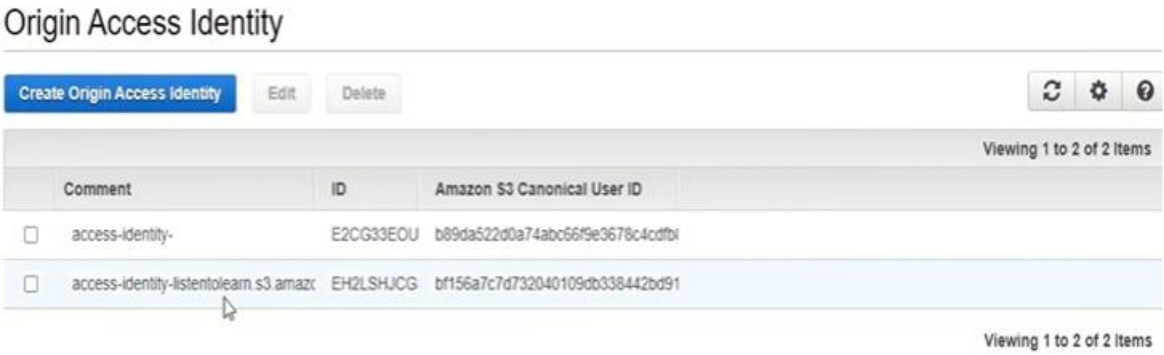
CloudFront Distributions

Create Distribution
Distribution Settings
Delete
Enable
Disable
Viewing : Any Delivery Method
Any State
Viewing 1 to 1 of 1 Items

	Delivery Method	ID	Domain Name	Comment	Origin	CNAMEs	Status	State	Last Modified
□	Web	E2YSQXGCOF1CCY	d1mnagax596ixq	-			In Prog	Enabled	2021-05-03 12:5

Viewing 1 to 1 of 1 Items

3. Go to identity access you will se new identity created there.



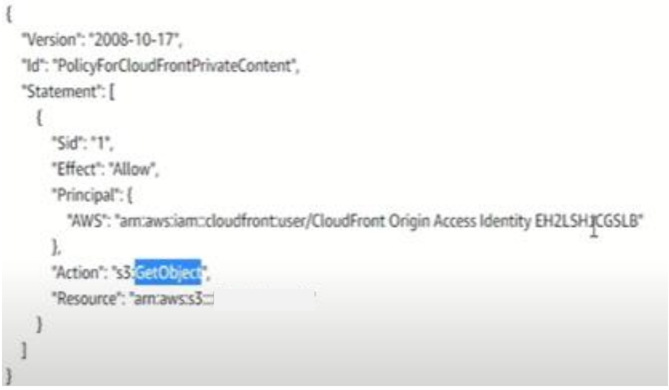
Origin Access Identity

Create Origin Access Identity
Edit
Delete
Viewing 1 to 2 of 2 Items

	Comment	ID	Amazon S3 Canonical User ID
□	access-identity-	E2CG33EOU	b89da522d0a74abc66f9e3678c4cdfb(
□	access-identity-listentolearn.s3.amaz(	EH2LSHJCG	bf156a7c7d732040109db338442bd91

Viewing 1 to 2 of 2 Items

4. See the bucket policy created by CloudFront. Identity ID will match. Copy the domain name, append the index file with this in URL and access the website through this.

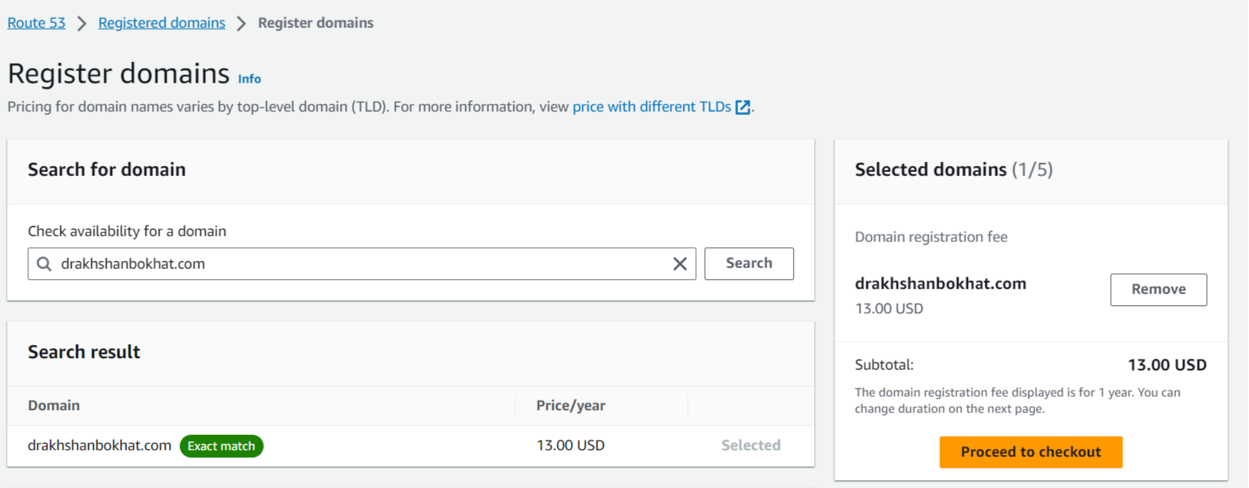


```
{
  "Version": "2008-10-17",
  "Id": "PolicyForCloudFrontPrivateContent",
  "Statement": [
    {
      "Sid": "1",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::cloudfront:user/CloudFront Origin Access Identity EH2LSHJCGSLB"
      },
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::
    }
  ]
}
```

Task 4: (ON YOUR CHOICE) Static Web Hosting using custom Route53 and S3.

- You would not be charged if you deleted or released the domain within 12 hours.
- Domain purchase will take up to three days for your domain to become active and available for use.
- Domain will come with the hosted zones attached to it with two records NS and SOA.

1. Search for route53 and register your domain (it can be a new domain, or you can use any existing domain). Go through the steps ann Review and submit if you want to buy a 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

drakhshanbokhat.com

Search result

Domain	Price/year	
drakhshanbokhat.com Exact match	13.00 USD	Selected

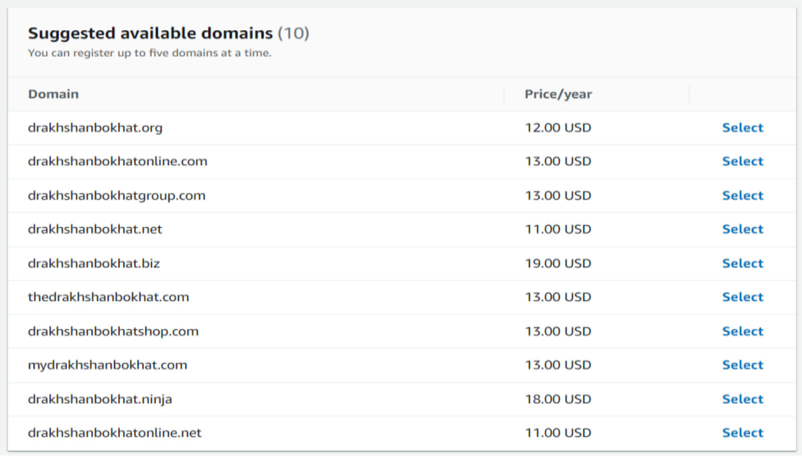
Selected domains (1/5)

Domain registration fee

drakhshanbokhat.com
13.00 USD

Subtotal: **13.00 USD**

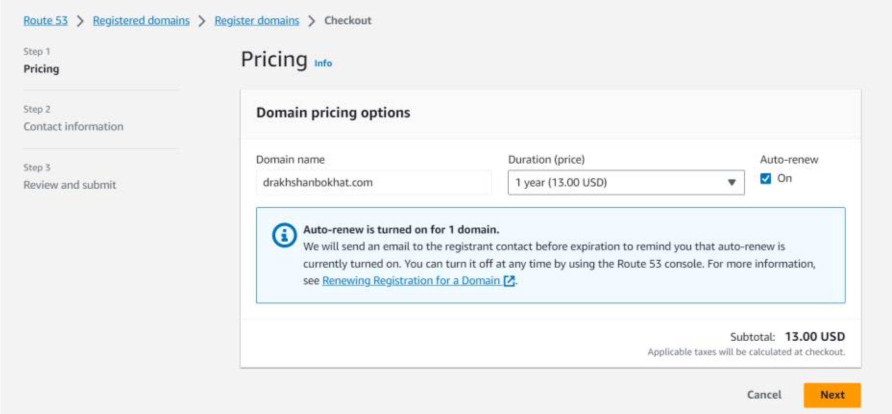
The domain registration fee displayed is for 1 year. You can change duration on the next page.



Suggested available domains (10)

You can register up to five domains at a time.

Domain	Price/year	
drakhshanbokhat.org	12.00 USD	Select
drakhshanbokhatonline.com	13.00 USD	Select
drakhshanbokhatgroup.com	13.00 USD	Select
drakhshanbokhat.net	11.00 USD	Select
drakhshanbokhat.biz	19.00 USD	Select
thedrakhshanbokhat.com	13.00 USD	Select
drakhshanbokhatshop.com	13.00 USD	Select
mydrakhshanbokhat.com	13.00 USD	Select
drakhshanbokhat.ninja	18.00 USD	Select
drakhshanbokhatonline.net	11.00 USD	Select



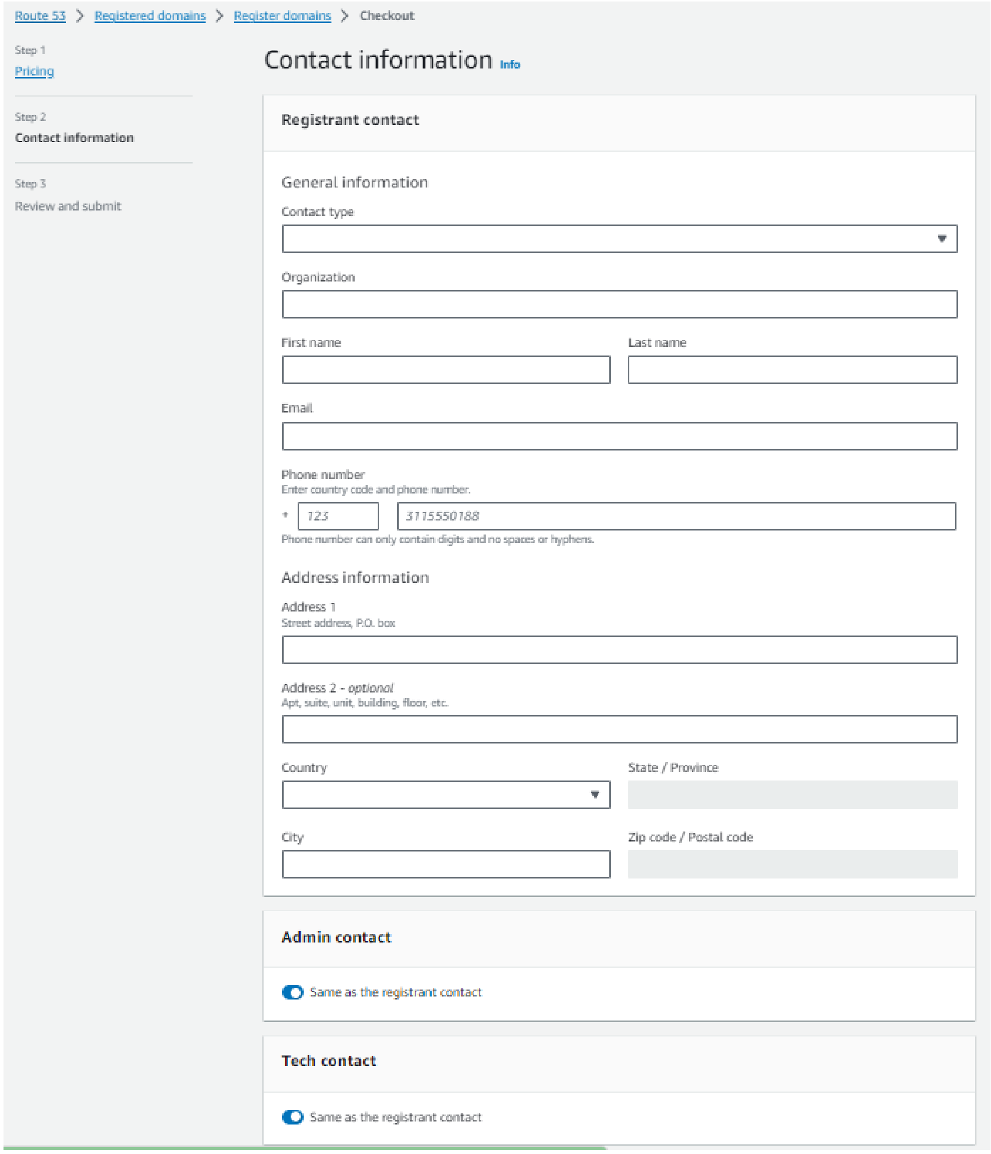
Pricing Info

Domain pricing options

Domain name	Duration (price)	Auto-renew
drakhshanbokhat.com	1 year (13.00 USD)	☑ 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: **13.00 USD**
Applicable taxes will be calculated at checkout.



Route 53 > Registered domains > Register domains > Checkout

Step 1
Pricing

Step 2
Contact information

Step 3
Review and submit

Contact information Info

Registrant contact

General information

Contact type

Organization

First name

Last name

Email

Phone number
Enter country code and phone number.

+ 123 3115550188

Phone number can only contain digits and no spaces or hyphens.

Address information

Address 1
Street address, P.O. box

Address 2 - *optional*
Apt, suite, unit, building, floor, etc.

Country

State / Province

City

Zip code / Postal code

Admin contact

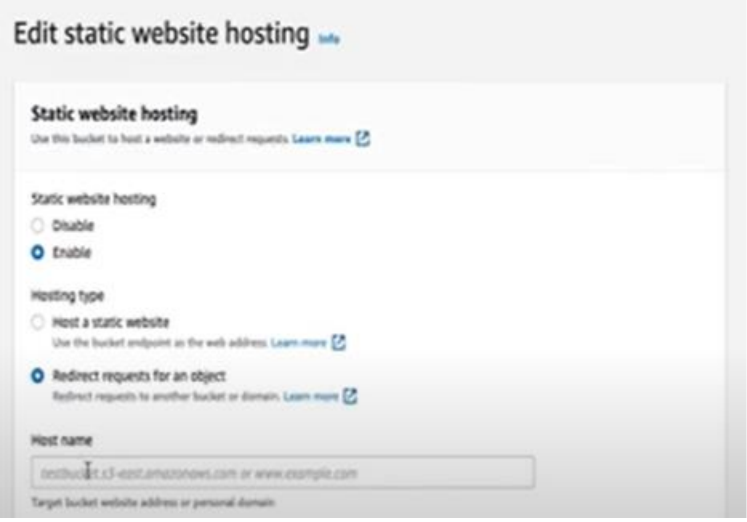
Same as the registrant contact

Tech contact

Same as the registrant contact

2. Setup the S3 buckets. You have to setup two buckets.
    a. First bucket is for root domain and its name must be same as custom domain (purchased /used).
    b. Second bucket is for subdomain and its name must be similar to first bucket with www attached to it.
    c. Both buckets must be in same location and select the location close to the clients/customers.
3. Setup the static website configuration. In your root bucket setup the files: index and error.
4. Setup the subdomain bucket and enable static website hosting. Add redirect request to an object to root domain bucket (set as hostname).

    * Use http as protocol. In order to use https cloud front is required.



Edit static website hosting

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

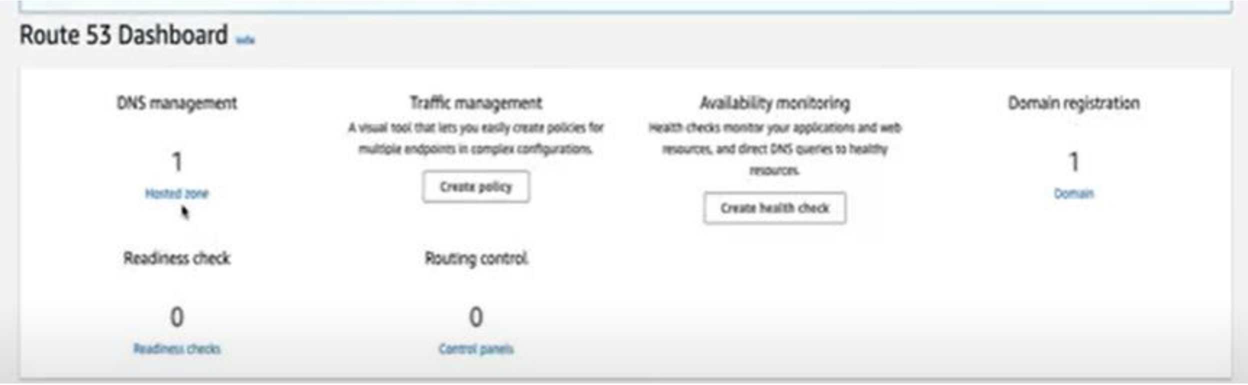
Redirect requests to another bucket or domain. Learn more

Host name

testbucket.s3-east.amazonaws.com or www.example.com

Target bucket website address or personal domain

5. Enable public access to bucket and set the policy. *Allow all the users to access the objects with in the bucket. For all use * (for action and objects)*
6. Access the website through S3 URL.
7. Go to route53 hosted zones and select the domain and add two records.



Route 53 Dashboard

DNS management: 1 Hosted zone

Traffic management: A visual tool that lets you easily create policies for multiple endpoints in complex configurations. Create policy

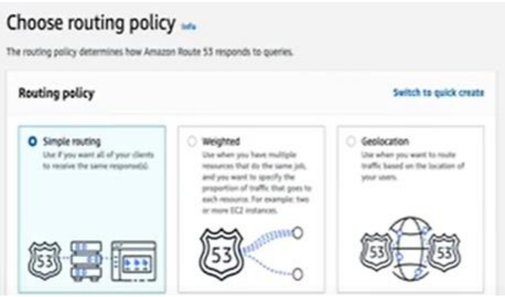
Availability monitoring: Health checks monitor your applications and web resources, and direct DNS queries to healthy resources. Create health check

Domain registration: 1 Domain

Readiness check: 0 Readiness checks

Routing control: 0 Control panels

8. Select simple routing. Add Simple Record (Type-A)
   a. Select S3 (traffic route to)
   b. Select the region of S3 bucket.
   c. URL will be gotten automatically.



Choose routing policy

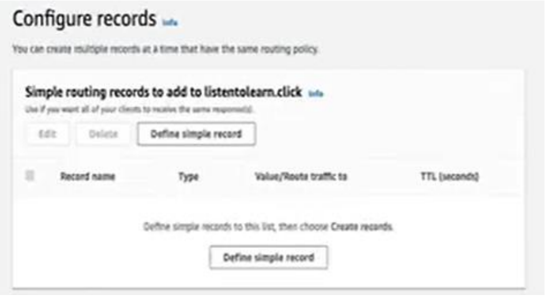
The routing policy determines how Amazon Route 53 responds to queries.

Routing policy — Switch to quick create

● Simple routing — Use if you want all of your clients to receive the same response(s).

○ Weighted — Use when you have multiple resources that do the same job, and you want to specify the proportion of traffic that goes to each resource. For example: two or more EC2 instances.

○ Geolocation — Use when you want to route traffic based on the location of your users.



Configure records

You can create multiple records at a time that have the same routing policy.

Simple routing records to add to listentolearn.click

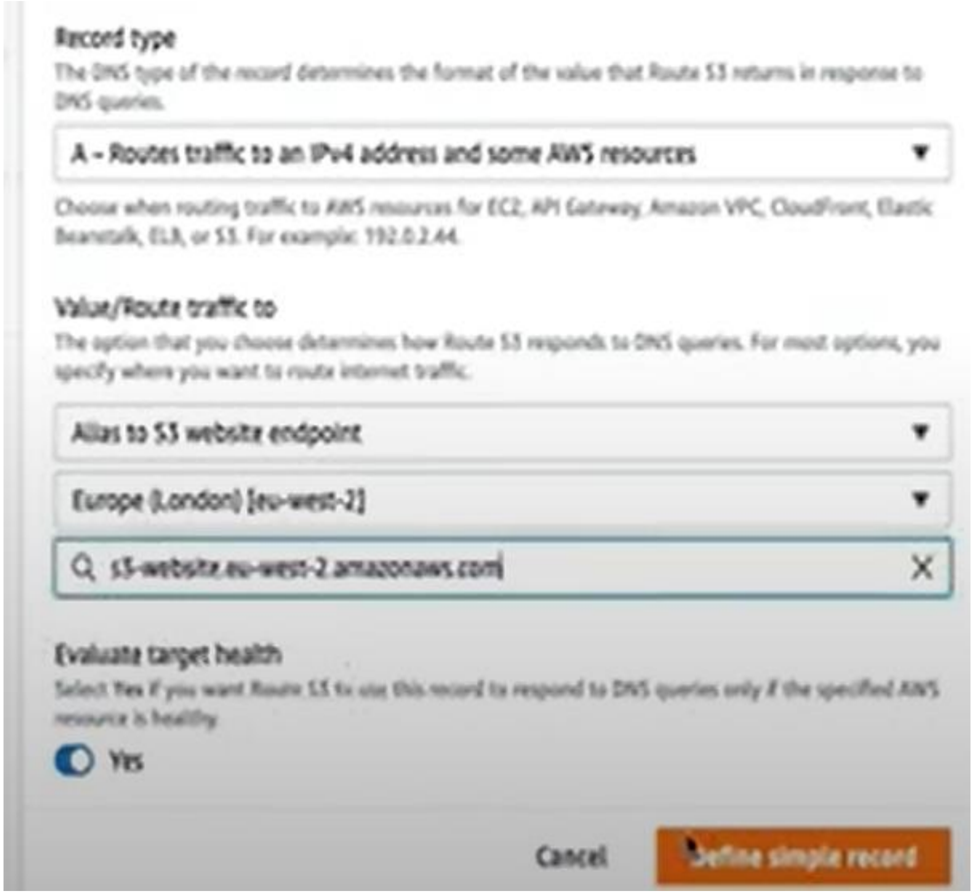
Use if you want all of your clients to receive the same response(s).

Edit | Delete | Define simple record

Record name	Type	Value/Route traffic to	TTL (seconds)

Define simple records to this list, then choose Create records.

Define simple record



Record type

The DNS type of the record determines the format of the value that Route 53 returns in response to DNS queries.

A - Routes traffic to an IPv4 address and some AWS resources ▼

Choose when routing traffic to AWS resources for EC2, API Gateway, Amazon VPC, CloudFront, Elastic Beanstalk, ELB, or S3. For example: 192.0.2.44.

Value/Route traffic to

The option that you choose determines how Route 53 responds to DNS queries. For most options, you specify where you want to route internet traffic.

Alias to S3 website endpoint ▼

Europe (London) [eu-west-2] ▼

s3-website.eu-west-2.amazonaws.com

Evaluate target health

Select Yes if you want Route 53 to use this record to respond to DNS queries only if the specified AWS resource is healthy.

● Yes

9. Repeat the same steps to define the record for second bucked add www at *blog* option.
10. Now both records are added. Access the website using your custom domain.