

# CLOUDFRONT & ROUTE53

## 1. Configure VPC peering in cross region

- Aws console
- Open vpc
- Create 2 vpc in 2 different regions
- N.virgina region as -1 vpc
- Ohio region – 2 vpc
- Creating vpc peering connection from n virgina region

us-east-1.console.aws.amazon.com/vpcconsole/home?region=us-east-1#CreatePeeringConnection

aws Search [Alt+S] United States (N. Virginia) Account ID: 4146-9191-2691 kamal

VPC > Peering connections > Create peering connection

### Create peering connection

A VPC peering connection is a networking connection between two VPCs that enables you to route traffic between them privately. [Info](#)

**Peering connection settings**

**Name - optional**  
Create a tag with a key of 'Name' and a value that you specify.

**Select a local VPC to peer with**

**VPC ID (Requester)**

**VPC CIDRs for vpc-0dc28a4aa37f88a38 (virginia)**

CIDR	Status	Status reason
172.31.0.0/16	Associated	-

**Select another VPC to peer with**

**Account**  
☒ My account  
☐ Another account

- Add requester vpc
- And acceptor vpc and id

us-east-1.console.aws.amazon.com/vpcconsole/home?region=us-east-1#PeeringConnectionDetails:VpcPeeringConnectionId=pcx-04c87be6a1e702fe9

aws Search [Alt+S] United States (N. Virginia) Account ID: 4146-9191-2691 kamal

VPC > Peering connections > pcx-04c87be6a1e702fe9

### VPC dashboard

AWS Global View [Filter by VPC](#)

**Virtual private cloud**

- Your VPCs
- Subnets
- Route tables
- Internet gateways
- Egress-only internet gateways
- Carrier gateways
- DHCP option sets

### pcx-04c87be6a1e702fe9 / peering-1.1

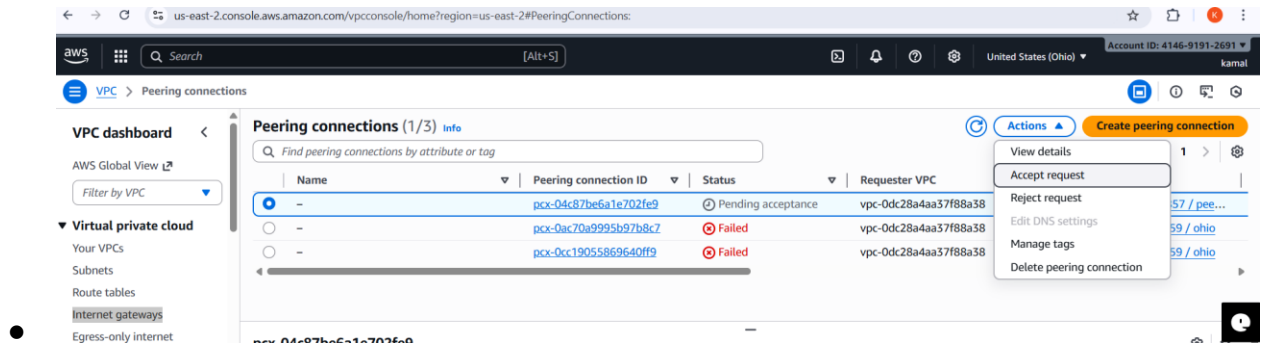
**Details**

<b>Requester owner ID</b> 414691912691	<b>Acceptor owner ID</b> 414691912691	<b>VPC Peering connection ARN</b> arn:aws:ec2:us-east-1:414691912691:vpc-peering-connection/pcx-04c87be6a1e702fe9
<b>Peering connection ID</b> pcx-04c87be6a1e702fe9	<b>Requester VPC</b> vpc-0dc28a4aa37f88a38 / virginia	<b>Acceptor VPC</b> vpc-0b36317faa0141857
<b>Status</b> Pending Acceptance by 414691912691	<b>Requester CIDRs</b> 172.31.0.0/16	<b>Acceptor CIDRs</b> -
<b>Expiration time</b> Wednesday, November 5, 2025 at 16:07:47 GMT+5:30	<b>Requester Region</b> N. Virginia (us-east-1)	<b>Acceptor Region</b> Ohio (us-east-2)

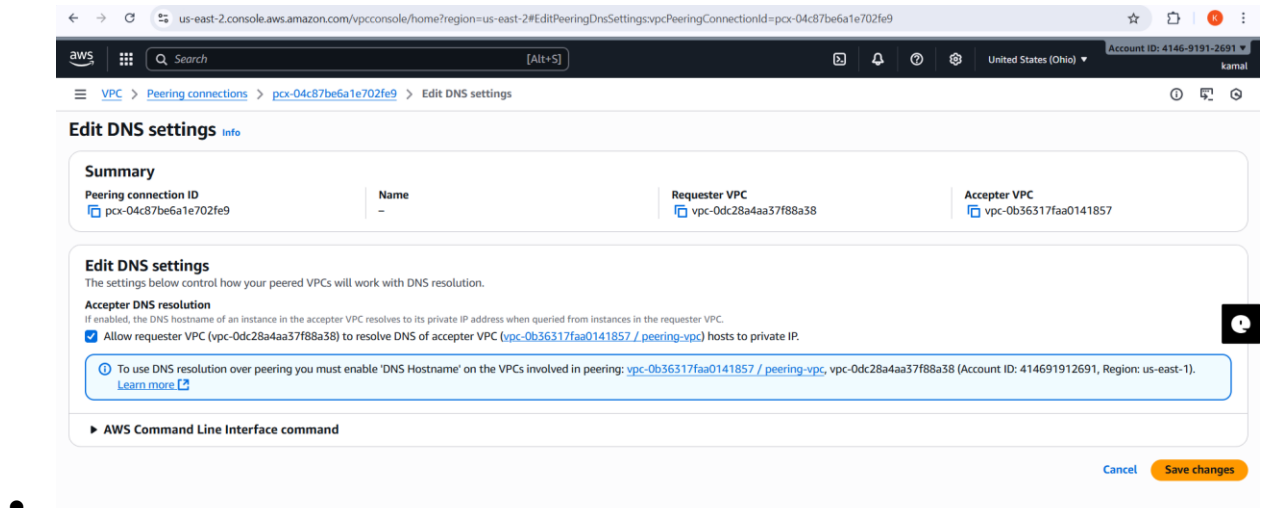
[DNS](#) [Route tables](#) [Tags](#)

- Go to acceptor vpc > peering connections and accept connection

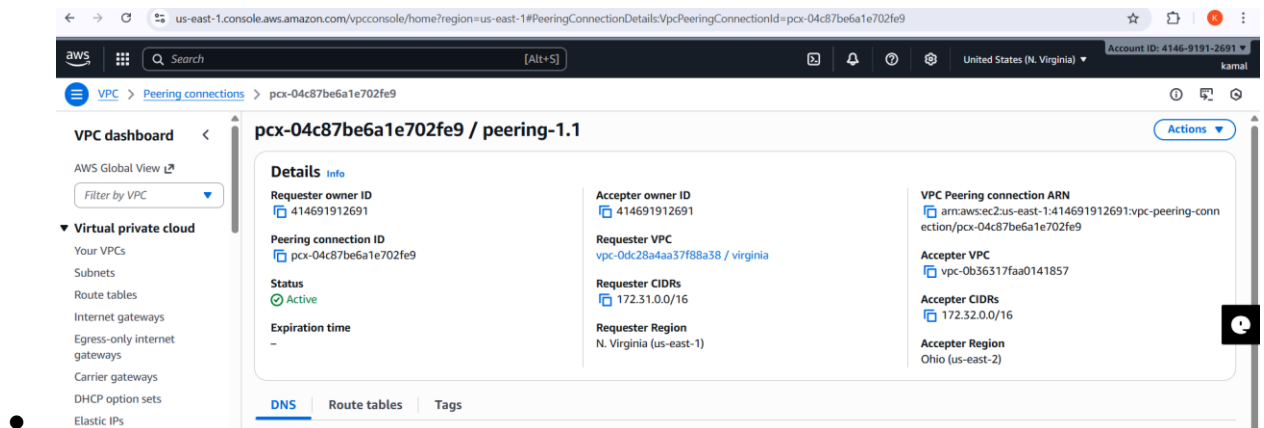
- 
- 
- 
-



## • Enable acceptor dns setting



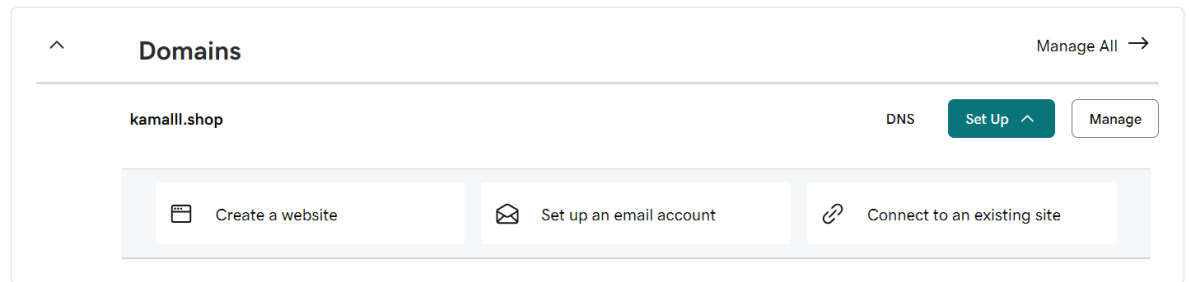
## • Peering successful



## 2 . Purchase one domain from GoDaddy.

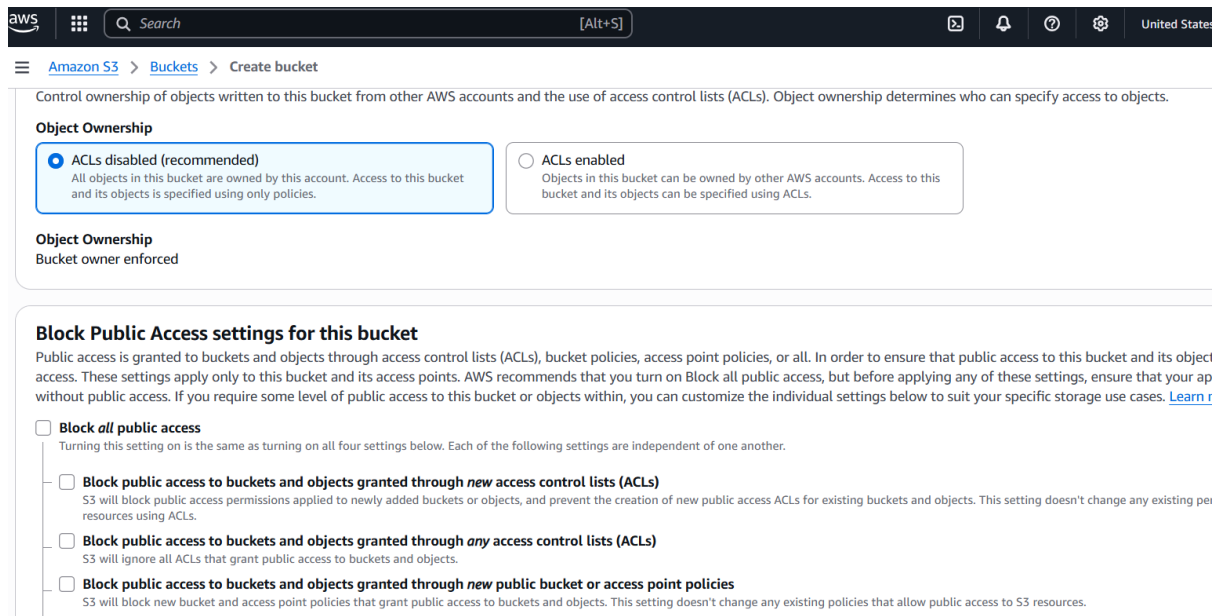
- Open godaddy
- Search for domains and buy
-

## All Products and Services



### 3 . Deploy static website in S3.

- Aws console
- Open s3
- Create a bucket
- Give unique bucket name globally
- Select acl disable option which is recommended
- Uncheck the block ip option



- Open bucket
- Navigate to properties
- Select host static website
- Now , open terminal and create 2 files
- Index.html -file 1
- Error.html -file2

- Now , enter the index.html file and error.html file in properties for static website

≡ [Amazon S3](#) > [Buckets](#) > [static-website-1.1](#) > Edit static website hosting

☒ 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 [Using Amazon S3 Block Public Access](#)

**Index document**  
Specify the home or default page of the website.

**Error document - optional**  
This is returned when an error occurs.

**Redirection rules – optional**  
Redirection rules, written in JSON, automatically redirect webpage requests for specific content. [Learn more](#)

- Upload 2 files in s3 bucket

Drag and drop files and folders you want to upload here, or choose **Add files** or **Add folder**.

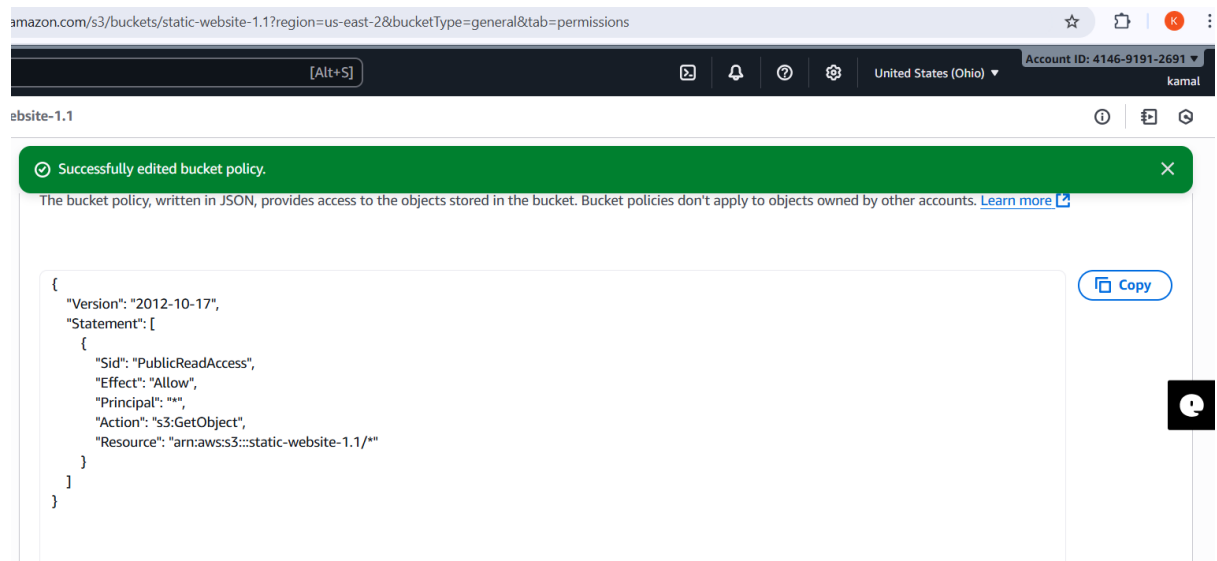
**Files and folders** (2 total, 41.0 B)

All files and folders in this table will be uploaded.

<input type="checkbox"/>	Name	Folder	Type	Size
<input type="checkbox"/>	error.html	-	text/html	22.0 B
<input type="checkbox"/>	index.html	-	text/html	19.0 B

- **Destination** [Info](#)
- - .. ..
- Go to bucket permission add a bucket policy
- Add json format policy
- {
- "Version": "2012-10-17",
- "Statement": [
- {
- "Sid": "PublicReadAccess",
- "Effect": "Allow",
- "Principal": "\*",
- "Action": "s3:GetObject",
- "Resource": "arn:aws:s3:::static-website-1.1/\*"
- }
- ]

- }



- 
- Now click static website link , you can able to see the page working



- 

4 . Create a CDN and attach one SSL certificate

5 . Create a Route 53 hosted zone and map the domain with the CDN.

- Go to **AWS Console** → **Route 53** → **Hosted zones**.
- Click **Create hosted zone**.
- **Domain name:** kamalll.shop
- **Type:** Public hosted zone
- Click **Create**.
- After creation, you'll see **four NS (Name Server)** records and one **SOA** record.

[Alt+S]

## Hosted zones (1)

View details
Edit
Delete
Create hosted zone

Automatic mode is the current search behavior optimized for best filter results. [To change modes go to settings.](#)

< 1 >

	Hosted zone name	Type	Created by	Record count
<input type="radio"/>	<a href="#">kamalll.shop</a>	Public	Route 53	4

Public
kamalll.shop
Info

Delete zone
Test record
Configure query logging

Hosted zone details
Edit hosted zone

Records (4)
DNSSEC signing
Hosted zone tags (0)

### Records (4) Info

Type
Routing p...
Alias

< 1 >

<input type="checkbox"/>	Record ...	Type	Routin...	Differ...	Alias	Value/Route traffic to
<input type="checkbox"/>	kamalll.sh...	NS	Simple	-	No	ns-285.awsdns-35.com. ns-1436.awsdns-51.org. ns-601.awsdns-11.net. ns-1767.awsdns-28.co.uk
<input type="checkbox"/>	kamalll.sh...	SOA	Simple	-	No	ns-285.awsdns-35.com. a

- **Change Nameservers in GoDaddy**
- Go to **GoDaddy** → **Domains** → **kamalll.shop** → **Manage DNS** → **Nameservers**.
- Click **Change** → **Enter my own nameservers (advanced)**.
- Delete the existing ones.
- Paste the **4 Route 53 nameservers** from Step 1.
- Click **Save / Apply Changes**.

Nameservers determine where your DNS is hosted and where you add, edit or delete your DNS records.

Using custom nameservers

Nameservers ?

ns-285.awsdns-35.com

ns-1436.awsdns-51.org

ns-601.awsdns-11.net

ns-1767.awsdns-28.co.uk

- 
- **Add ACM Validation Records (in Route 53)**
- Go to **AWS Console → Certificate Manager (ACM) →** open your certificate (kamalll.shop).
- Copy each **CNAME Name** and **CNAME Value** shown under “Domain validation”.
- Go back to **Route 53 → your hosted zone → Create record**.
- Create a record, with CNAME AND CNAME VALUE

<input type="checkbox"/>	_200aac1...	CNAME	Simple	-	No	_011d5447ff43275efa01
--------------------------	-------------	-------	--------	---	----	-----------------------

- 
- Wait till certificate is issued

The screenshot shows the AWS Certificate Manager console. At the top, there's a navigation bar with the AWS logo, a search bar, and various icons. Below the navigation bar, the breadcrumb trail shows 'ates > bb3f802e-8fcc-4ef3-93a6-0661c7cfe480'. The main heading is 'bb3f802e-8fcc-4ef3-93a6-0661c7cfe480' with a 'Delete' button. The 'Certificate status' section displays the following information:

Identifier	Status
bb3f802e-8fcc-4ef3-93a6-0661c7cfe480	Issued

Below the status table, the 'ARN' is shown as 'arn:aws:acm:us-east-1:414691912691:certificate/bb3f802e-8fcc-4ef3-93a6-0661c7cfe480'. The 'Type' is listed as 'Amazon Issued'.

- 
- Now go CLOUDFRONT
- Create a distribution
- Select s3 bucket which contains Index.html and error.html
- Add the ACM Certificate

Get TLS certificate

**TLS certificate** [Info](#) [Refresh certificates](#)

Transport layer security (TLS) encrypts communication to and from your domain. You must have a TLS certificate with AWS Certificate Manager (ACM) to use CloudFront.

**Available certificates** [Info](#)

These certificates cover the domains that will be served by this distribution.

- ☒ kamalll.shop (bb3f802e-8fcc-4ef3-93a6-0661c7cfe480)
- ☐ Create a new certificate

**Certificate details** [View in AWS Certificate Manager](#)

<b>ARN</b> arn:aws:acm:us-east-1:414691912691:certificate/bb3f802e-8fcc-4ef3-93a6-0661c7cfe480	<b>Covered domains</b> kamalll.shop	<b>Source</b> Amazon
---	--	-------------------------

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

- After successfully created , wait until it get deployed

amazon.com/cloudfront/v4/home?region=us-east-1#/distributions

**Distributions (1)** [Info](#) [Refresh](#) [Enable](#) [Disable](#) [Delete](#) [Create distribution](#)

Search all distributions Filter type All distributions

ID	Status	Description	Type	Domain name	Alternate domain names	Origins	Last modified
E3E1L3KJLF7UIU	Enabled	-	Standard	d13f61e...	kamalll.shop	static-website-1.1.:	Deploying

- Once deployed ,
- Copy distributiou domain name ,
- Navigate to R53
- Selected hostezone add records
- FOR ALIAS WE NEED TO ADD 2 RECORDS
- Create record -1 > name empty > type A > ON ALIAS
- > SELECT ALIAS TO CLOUDfront distribution
- CHOOSEDISTRIBUTION ALSO
- DONE

Route 53 > Hosted zones > kamalll.shop > Create record

**Record 1** [Delete](#)

**Record name** [Info](#)  
subdomain kamalll.shop

**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).

Q d13f61ehaf9m0m.cloudfront.net

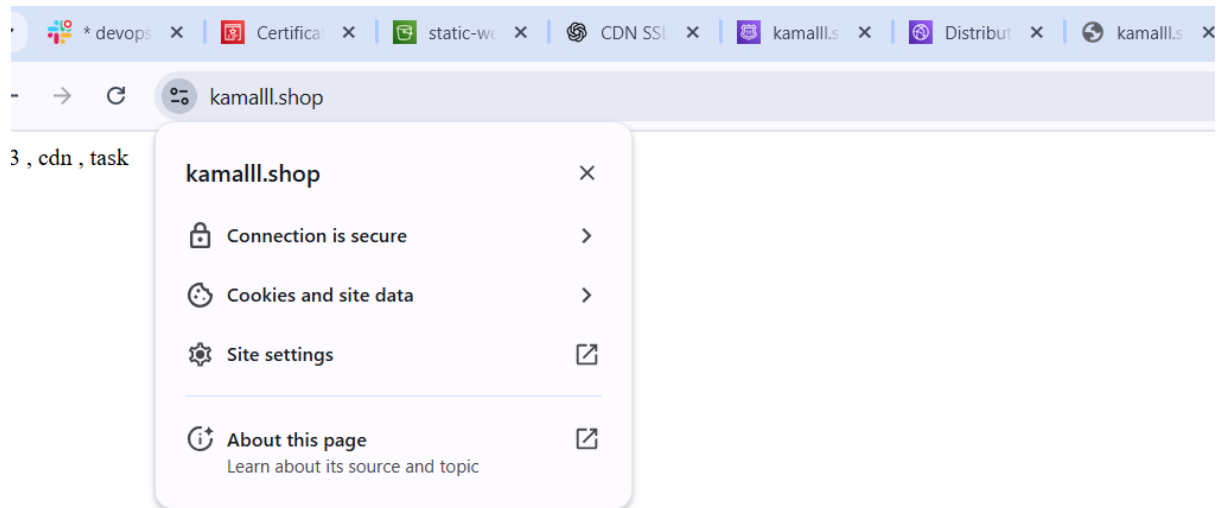
Use: "d13f61ehaf9m0m.cloudfront.net"

kamalll.shop (d13f61ehaf9m0m.cloudfront.net)

kamalll.shop (d13f61ehaf9m0m.cloudfront.net)



- kamalll.shop → main/root domain (FOR ALIAS)
- www.kamalll.shop → the www version (FOR ALIAS)
- Now go to , distribution , and open DISTRIBUTIONS
- SELECT INVALIDATIONS AND GIVE PATH AS ( /\* )
- CHECK POLICY OF BUCKET IS PUBLIC S3
- S3 BUCKET SHOULD CONTAIN INDEX.HTML AND ERROR.HTML FILES



- 
- Connection is secure