# Aws Cloud front & Route 53

**Objective:**

The objective of this assignment is to design and implement a complete end-to-end cloud infrastructure solution on AWS by configuring cross-region VPC peering, deploying a static website using Amazon S3, and enabling global content delivery with Amazon CloudFront. This includes purchasing and configuring a custom domain from GoDaddy, securing the website using an SSL certificate, and integrating the domain with AWS Route 53 for DNS management. The task further aims to validate successful deployment by updating website content and ensuring accessibility through the custom domain. The final deliverable will demonstrate the ability to deploy a publicly accessible, secure, and scalable web application using AWS services and third-party DNS management.

1. Configure VPC peering in cross regions.

* Created two vpc’s one in united states North Virginia. Other in EU Stockholm.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

* Created two subnets (public, private) in both regions

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

* Created two route tables (public, private) in both regions
* Added respective subnets to their route tables in subnet association option.

A screenshot of a computer

AI-generated content may be incorrect.

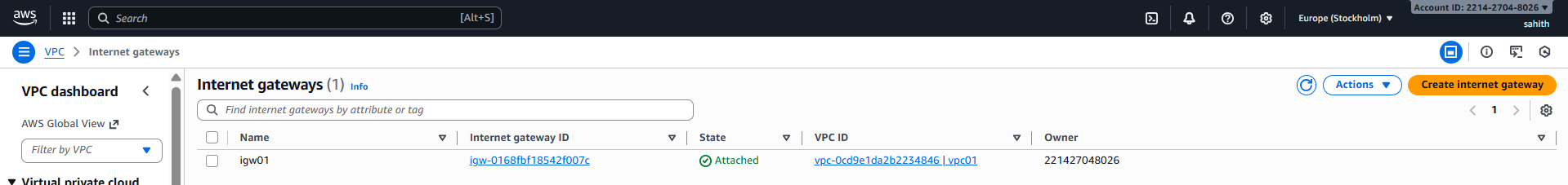
A screenshot of a computer

AI-generated content may be incorrect.

* Created one internet gateway in both regions. First igw to vpc .Attached their ID’s in public route tables.

A screen shot of a computer

AI-generated content may be incorrect.



* Created a vpc peering connection. Sent a request from North Virginia region to Europe Stockholm

A screenshot of a computer

AI-generated content may be incorrect.

* Accepted the request sent from north Virginia.

A screenshot of a computer

AI-generated content may be incorrect.

* Validation: as we can see below screenshot. It is connecting with private Ip of other ec2

A screenshot of a computer program

AI-generated content may be incorrect.

1. Purchase one domain from GoDaddy.

* Go to GoDaddy.com website > create an account > search for domain name > add to cart > click on buy now follow the steps.

A screenshot of a computer

AI-generated content may be incorrect.

* As we can see below, I have purchased domain name **shoppiex. store**

A screenshot of a computer

AI-generated content may be incorrect.

* Validation: Website launched.

A person looking at something

AI-generated content may be incorrect.

1. Deploy static website in S3.

* Created an s3 bucket with domain name

A screenshot of a computer

AI-generated content may be incorrect.

* Uploaded index.html and error.html from local machine. And made them public acls

A screenshot of a computer

AI-generated content may be incorrect.

* Create a static website hosting > gave domain name > gave both file names > click on create.

A screenshot of a computer

AI-generated content may be incorrect.

* Here we can see that it has been static website has been created

A screenshot of a computer

AI-generated content may be incorrect.

* Validation: we can check it in browser.

A screen shot of a computer

AI-generated content may be incorrect.

4.Create a CDN and attach one SSL certificate.

* Create a Route 53 hosted zone and map the domain with the CDN.
* Update the index.html in the S3 bucket and ensure the updated file is accessible using the domain name.
* Aws console > Route 53 > create hosted zone > in domain name gave shoppiex.store > remaining everything keep them default and click on create hosted zone.

A screenshot of a computer

AI-generated content may be incorrect.

* Hosted zone has been created successfully

A screen shot of a computer

AI-generated content may be incorrect.

* From the created hosted zone copy the value/route traffic, which is marked below.

A screenshot of a computer

AI-generated content may be incorrect.

* Redirect to GoDaddy website where we purchased the domain > select profile > DNS > domain name > nameservers.
* In name servers first remove old names add route traffic which we copied in before step by removing (.) at the end and save them.

A screenshot of a computer

AI-generated content may be incorrect.

* Go to Aws certificate manager > Create a certificate > request > select request a public certificate in certificate type > provide domain name here and keep remaining everything default > click on create certificate.

A screenshot of a computer

AI-generated content may be incorrect.

* As we can see below certificate has been created successfully.

A screenshot of a computer

AI-generated content may be incorrect.

* Go to cloud front > Distributions > give distribution name > select next

A screenshot of a computer

AI-generated content may be incorrect.

* keep everything default in this page, move to next page.

A screenshot of a computer

AI-generated content may be incorrect.

* keep everything default in this page, click on create distribution.

A screenshot of a computer

AI-generated content may be incorrect.

* as we can see below distribution has been created successfully.

A screenshot of a computer

AI-generated content may be incorrect.

* once distribution has been created, move to create origin > gave path name > selected public > click on create orgin

A screenshot of a computer

AI-generated content may be incorrect.

* as we can origin has been created successfully.

A screenshot of a computer

AI-generated content may be incorrect.

* Go to invalidation > create invalidation with /index.html

A screen shot of a computer

AI-generated content may be incorrect.

* Go to invalidation > create invalidation with /\*

A screen shot of a phone

AI-generated content may be incorrect.

* Navigate to s3 bucket and update and make them public from actions option.

A screenshot of a computer

AI-generated content may be incorrect.

* Website is working perfectly.

A screenshot of a website

AI-generated content may be incorrect.