## Darshan Nikam Date: 20/03/2024

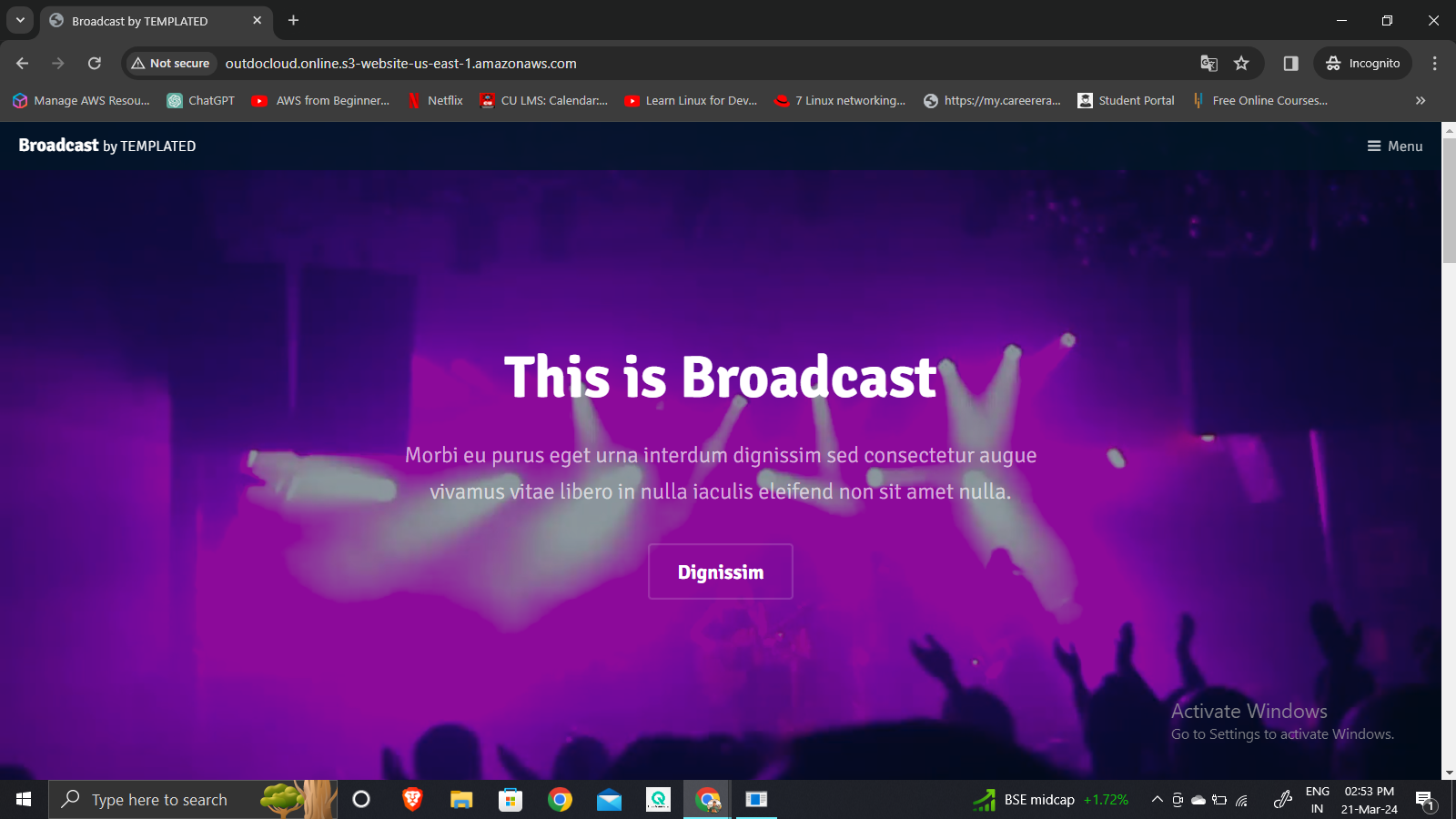
### **Route 53**

Practice: Hosting a Static Website on Amazon S3 and Configuring Route 53.

Hosting a static website on Amazon S3 (Simple Storage Service) and configuring Route 53 for DNS management provides a simple and cost-effective solution for serving web content. This practice involves utilizing Amazon S3 to store the website files and configuring Route 53 to map a custom domain name to the S3 bucket's endpoint using an alias record.

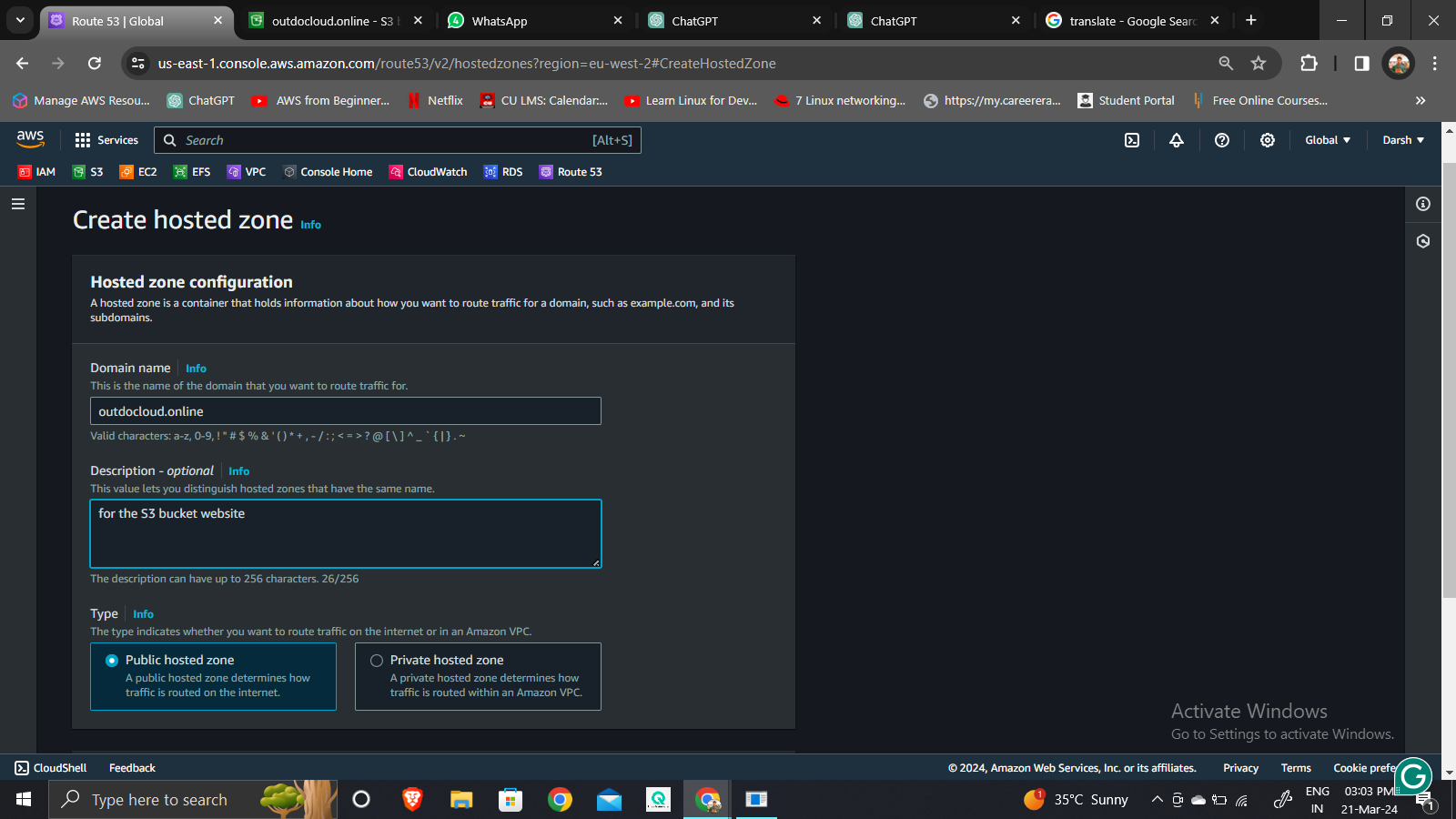
1. ***Create an S3 bucket with the same DNS name.***

* ***Upload the website file (Template) into the bucket.***
* ***Set Permission for public access.***
* ***Edit ACLs and allow “list and read” permissions for public access***
* ***In Bucket Properties, enable the static website hosting.***
* ***Browse the static URL of the bucket and verify website is accessible.***

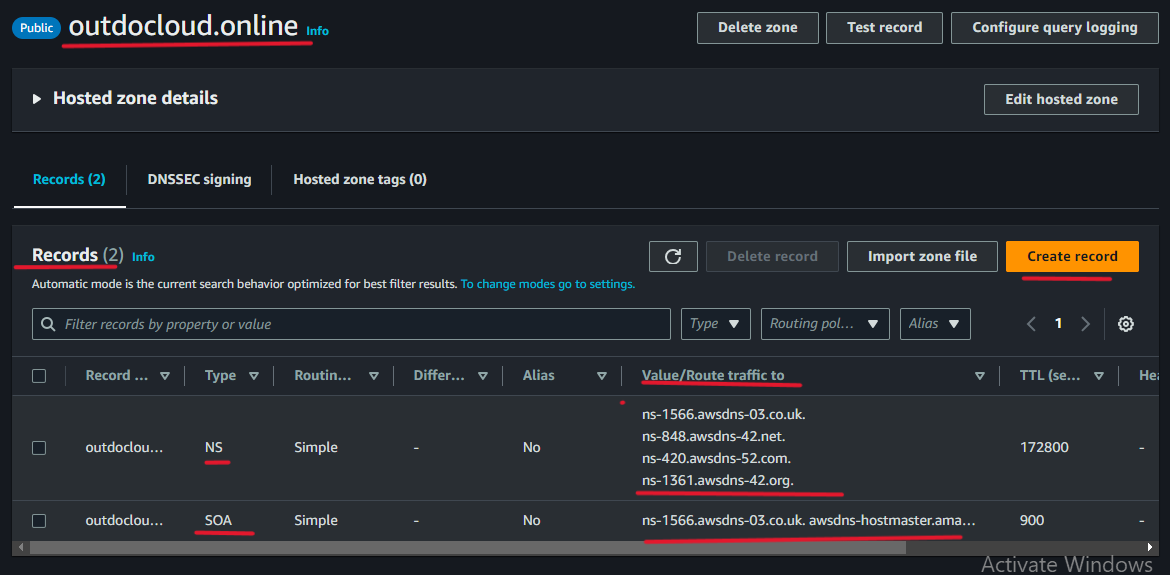


***Here my website is publically accessible using a bucket URL. Now we want to access this website using our domain name.***

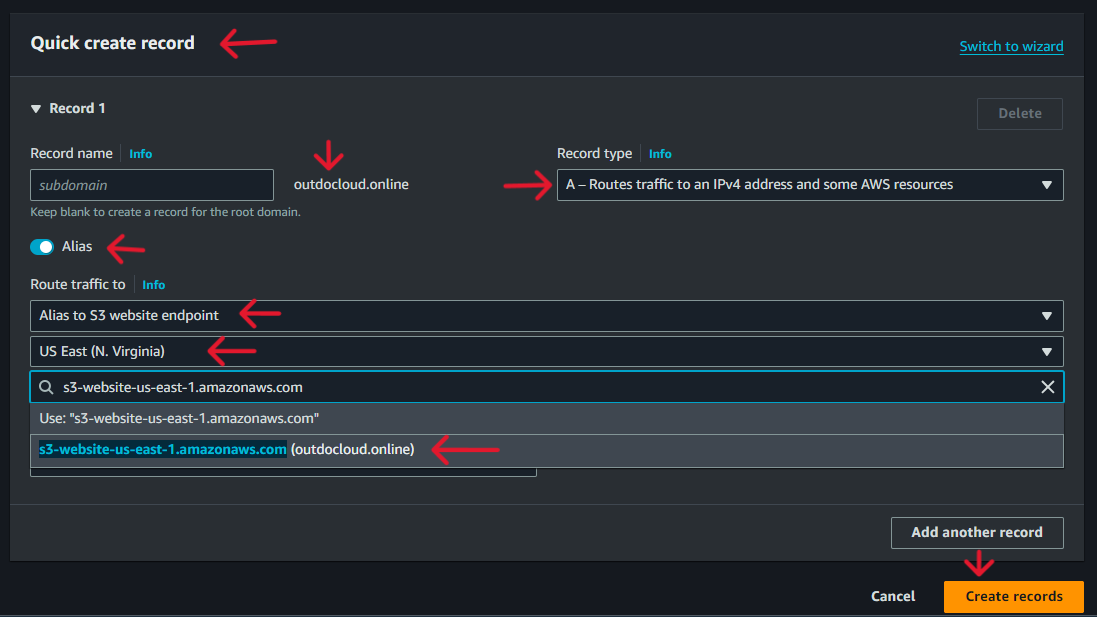
1. ***Go to the Route 53 service, select the Hosted zone, and click on the Create Hosted zone button.***
2. ***Enter the domain name that you have. Select type as public hosted zone to route traffic on the internet, and click on create hosted zone button.***



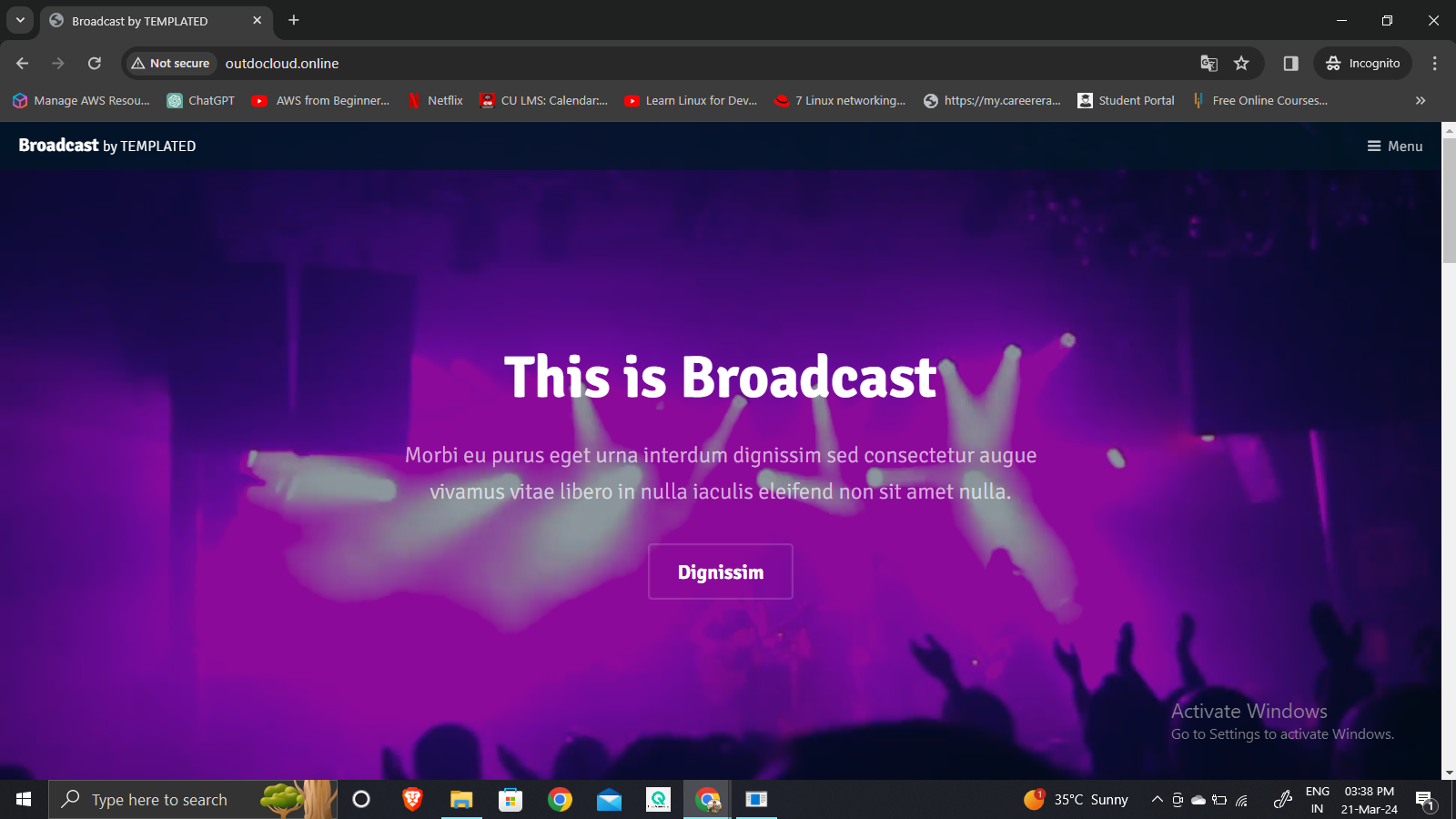
1. ***By default, Route 53 creates two records: NS (Name Server) and SOA (Start of Authority).***



* ***The NS record specifies the name servers for your domain. Configure this NS record with your domain registrar.***
* ***The SOA record contains administrative information about the zone.***
* ***Now we need to create a new record for the bucket.***
* ***Click on Create Record, record name box keeps blank to create a record for the root domain.***
* ***Select the record type as A, and click on the Alias button to select it.***
* ***Select Alias to the S3 website endpoint to route traffic.***
* ***Choose a region where the s3 bucket is created, select the bucket, and then click on Create Record.***



1. ***Now go to the web browser and search your domain name.***



***Here my website is accessible using the domain name.***

### **Route 53 – Health Checks**

***Route 53 Health Checks is a service provided by Amazon Web Services (AWS) as part of its Route 53 DNS (Domain Name System) service. It allows you to monitor the health and performance of your web applications and resources by periodically sending requests to your endpoints, such as websites, web servers, or other resources, and checking their responsiveness. Here's how it works and why it's considered global:***

* ***Monitoring Endpoints: Route 53 Health Checks regularly monitors the health of your endpoints by sending requests (HTTP, HTTPS, TCP, or UDP) to these endpoints from multiple locations around the world.***
* ***Configurable Parameters: You can configure various parameters for these health checks, such as the frequency of checks, the protocol used, the endpoint to check, and the expected response. This flexibility allows you to tailor the health checks according to the specific needs of your applications.***
* ***Health Check Status: Based on the responses received from the endpoints, Route 53 Health Checks determines the health status of each endpoint. If an endpoint fails to respond or responds with an unexpected status code, it's considered unhealthy.***
* ***DNS Failover: Route 53 Health Checks can be used in conjunction with DNS failover to route traffic away from unhealthy endpoints to healthy ones automatically. This ensures high availability and reliability for your applications.***
* ***Global Reach: One of the key features of Route 53 Health Checks is its global reach. AWS has multiple monitoring locations around the world, allowing it to perform health checks from various geographic locations. This global coverage ensures that your applications are monitored from diverse locations, giving you a comprehensive view of their health and performance.***
* ***Highly Available: Route 53 Health Checks itself is designed to be highly available and reliable. It operates on the same infrastructure that powers Amazon's global network of data centers, ensuring minimal downtime and robust performance.***
* ***Integration with Other AWS Services: Route 53 Health Checks seamlessly integrates with other AWS services like Route 53 DNS, Amazon Cloudwatch, and AWS Lambda, allowing you to automate responses to health check events and gain deeper insights into the health and performance of your applications.***

***Overall, Route 53 Health Checks is a powerful tool for monitoring the health and performance of your applications globally, ensuring that they remain available and responsive to users around the world.***

***Practice: Monitoring Multi-Region Websites with Route 53 Health Checks***

***In this practice, we'll see how to use Route 53 Health Checks to keep an eye on websites hosted in different regions. We’ll set up these checks to make sure your websites are working well and can handle visitors from anywhere.***

1. ***Launch 2 Ec2 Instances in two different regions.***

* ***Allow SSH and HTTP rules.***
* ***Add script in user data of both Instances:***

#!/bin/bash

sudo -i

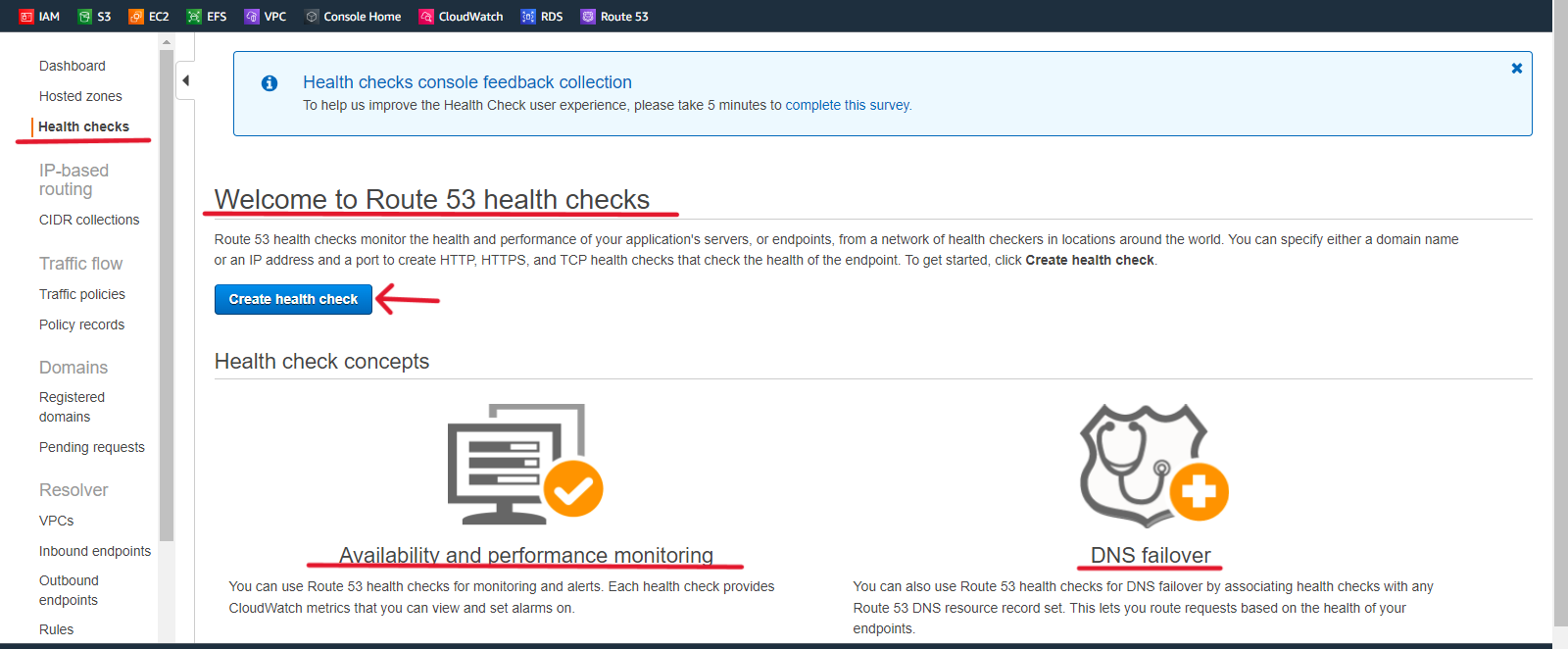
yum install httpd -y

systemctl start httpd

systemctl enable httpd

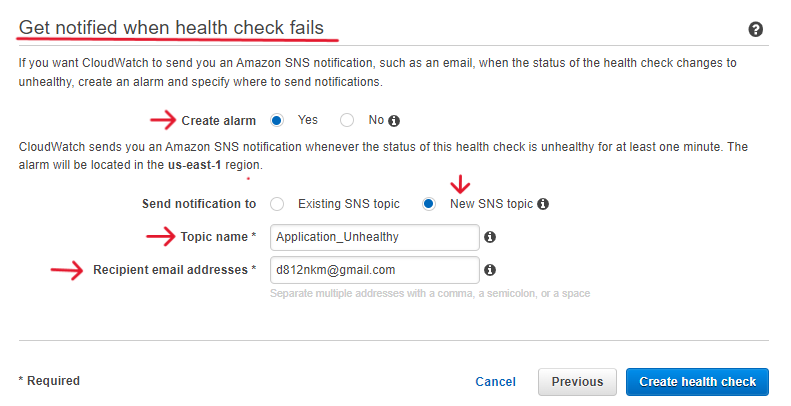
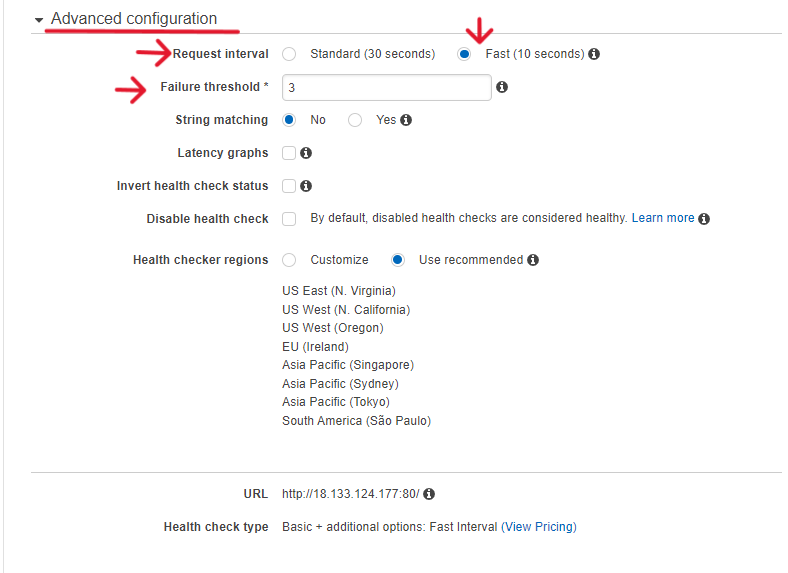
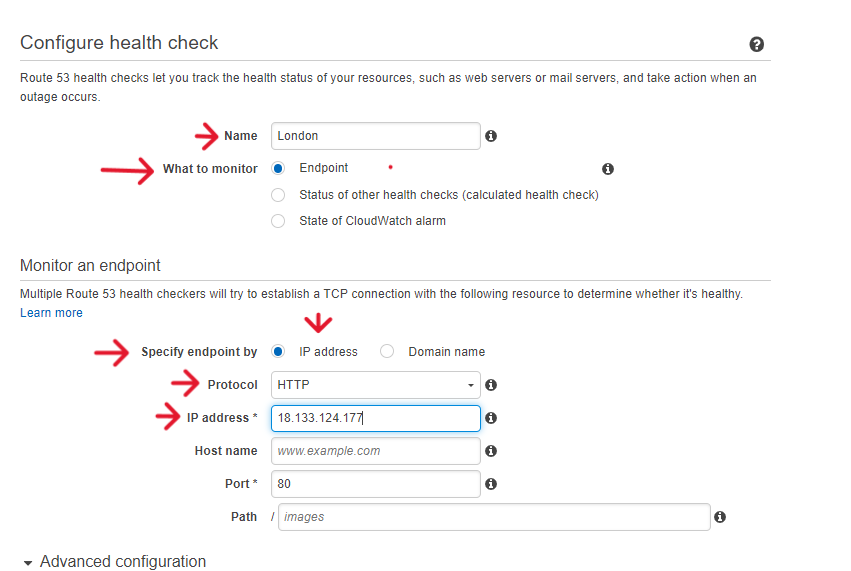
echo “Hello everyone, this “Region Name” > /var/www/html/index.html

1. ***Go to Route 53 Dashboard and select Health Checks service.***

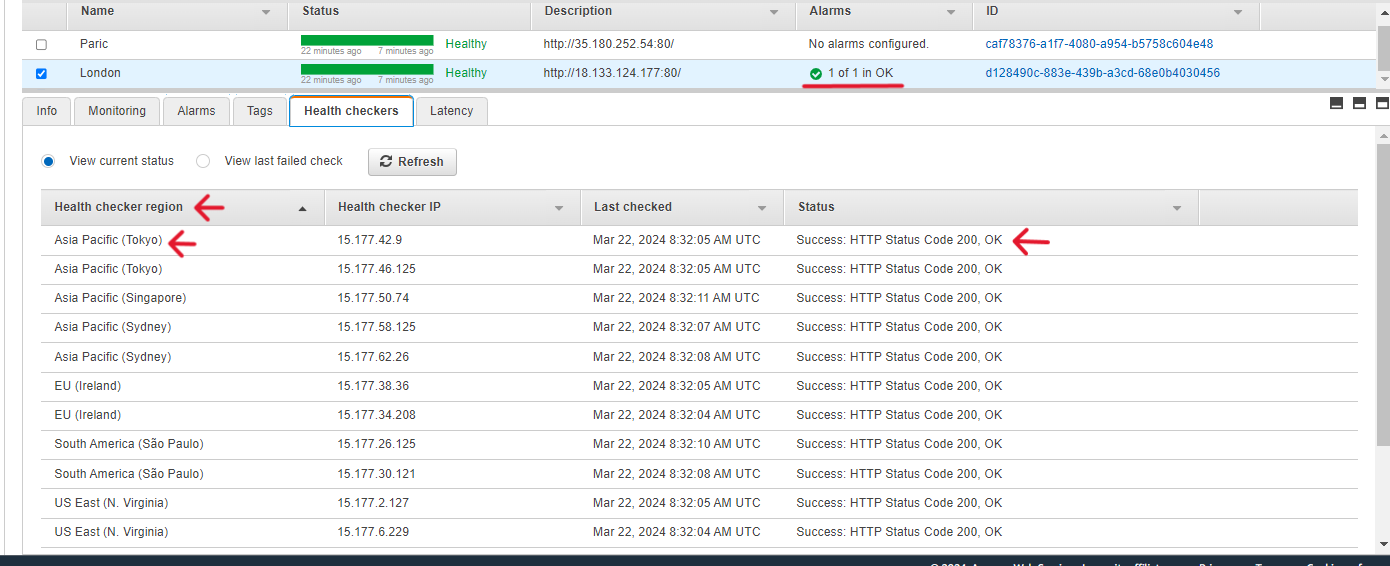
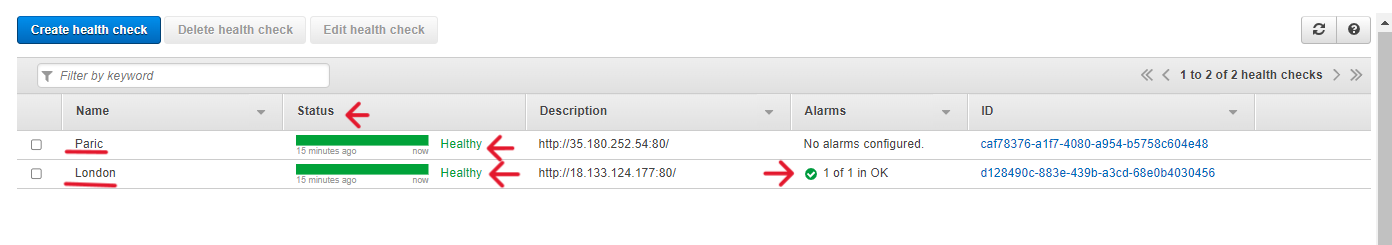


* ***Click on the Create Health Check button.***
* ***Name the health check the same as your Instance to identify.***
* ***Select Endpoint: (You can enter the domain name or IP address of the endpoint you want to monitor. This could be a website, an API endpoint, a load balancer, an EC2 instance, or any other resource accessible over the internet.)***
* ***Enter Ec2 Instance IP Address.***
* ***Check Advance Configuration and Specify as required.***
* ***If you want CloudWatch to send you an Amazon SNS notification whenever the status of this health check is unhealthy for at least one minute then click on Create alarm.***
* ***Click on the Create Health Check button.***
* ***Create a new health check for the second instance following the steps outlined above.***

***Images reference for the above steps***

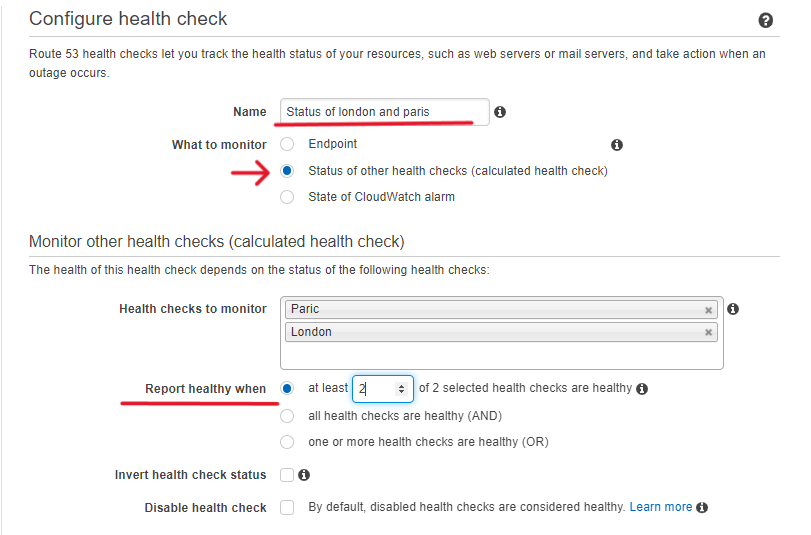


***In the below image, we can see our both instances are healthy.***

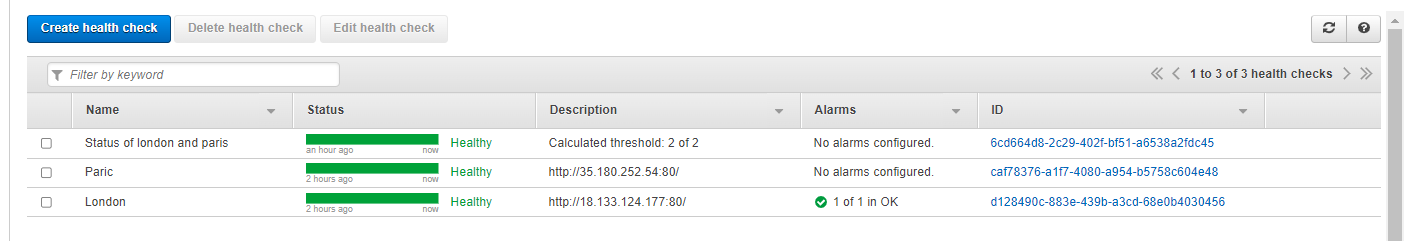


1. ***Now we will create a Health Check for the status of other Health Checks.***

* ***Click on Create a Health check***
* ***Select the option Status of other Health Checks.***
* ***Select both health checks in the “Health Checks to Monitor” option.***
* ***Determine the method when this status reports healthy status, here we determining the report healthy, when both health checks are healthy, which means one of the health checks is the unhealthy status will unhealthy.***
* ***Click on the next button, select the alarm option yes or no, and click on the Create Health Check button.***

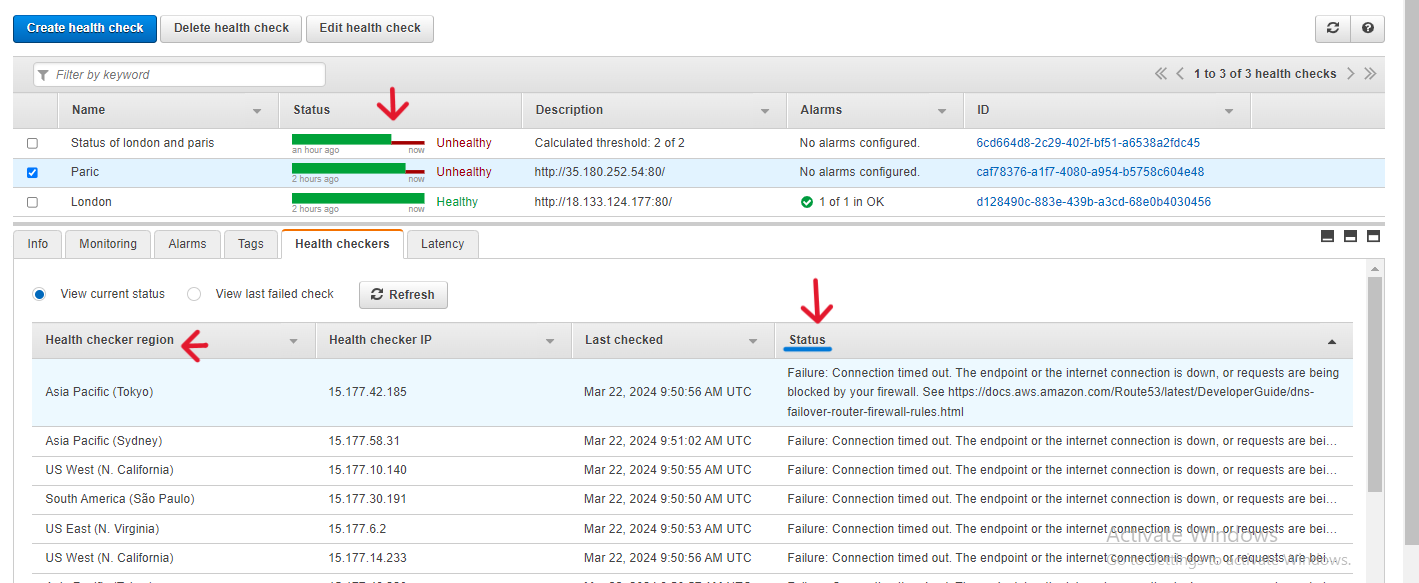


1. ***Now check the status of health checks***



***We can see here both region's Instances are healthy, that’s why the “Status of London and Paris” health check is also healthy.***

1. ***Now stop the Paris Instance and then again check the health Status.***



***Here we can see the health status is unhealthy after stopping the Instance.***

***“””””””””””””””””””””””””””””””””””””””””””””””””””””””””””””””””””””””””””””””””””””””””””””””””””””””””””””””””””””””””***

***Route 53 Hosted Zone Routing Policy***

***A hosted zone routing policy is a configuration option within Amazon Route 53, which is a highly available and scalable cloud Domain Name System (DNS) web service provided by Amazon Web Services (AWS). Route 53 allows users to manage DNS routing for their domain names.***

***When you create a hosted zone in Route 53, you have the option to specify how DNS queries for your domain name are directed to your resources. The routing policy determines how Route 53 handles DNS queries based on factors like health checks, latency, geographic location, and weighted distribution.***

***types of routing policies available in Amazon Route 53:***

1. ***Simple Routing: Simple routing associates a single resource (like a server) with a DNS record. It's straightforward and directs all traffic to the specified resource.***

***Example: You have a website hosted on a single server with the IP address 203.0.113.1. You create a DNS record that maps your domain name (example.com) to this IP address.***

1. ***Weighted Routing: Weighted routing allows you to distribute traffic among different resources based on assigned weights. This is useful for load balancing or A/B testing.***

***Example: You have two servers: Server A (203.0.113.1) and Server B (198.51.100.1). You assign 70% of the traffic to Server A and 30% to Server B.***

1. ***Latency-Based Routing: This routing policy directs users to the server with the lowest latency or shortest network path, improving website performance.***

***Example: Users in Asia are directed to a server located in Singapore, while users in North America are directed to a server located in the United States.***

1. ***Failover Routing: Failover routing sets up primary and secondary resources. If the primary resource becomes unavailable, traffic is automatically routed to the standby resource.***

***Example: Your primary website server is in Region A. You set up a backup server in Region B. If the primary server goes down, traffic is redirected to the backup server in Region B.***

1. ***Geolocation Routing: Geolocation routing directs users to different resources based on their geographic location. It's useful for serving region-specific content.***

***Example: Users in Europe are directed to a server in Frankfurt, while users in South America are directed to a server in São Paulo.***

1. ***Geoproximity Routing: Geoproximity routing considers both user location and resource location to direct traffic. It helps optimize routing based on proximity.***

***Example: Users in India are directed to the nearest server location, which might be in Mumbai or Chennai, depending on their actual location.***

1. ***IP-Based Routing: IP-based routing directs traffic based on the IP address of the DNS resolver. It's useful for directing traffic from specific networks or regions.***

***Example: Traffic from corporate offices with IP range 192.0.2.0/24 is directed to an internal server, while traffic from the public internet is directed to an external server.***

1. ***Multivalue Answer Routing: This routing policy returns multiple IP addresses in response to DNS queries, allowing for basic load balancing across multiple resources.***

***Example: You have three identical web servers hosting your website. With Multivalue Answer Routing, Route 53 can return the IP addresses of all three servers to DNS queries. When users access your website, their DNS resolver will randomly select one of these IP addresses, distributing the incoming traffic evenly across your servers and improving overall performance and reliability.***