Part 1: EC2 with ELB and ASG

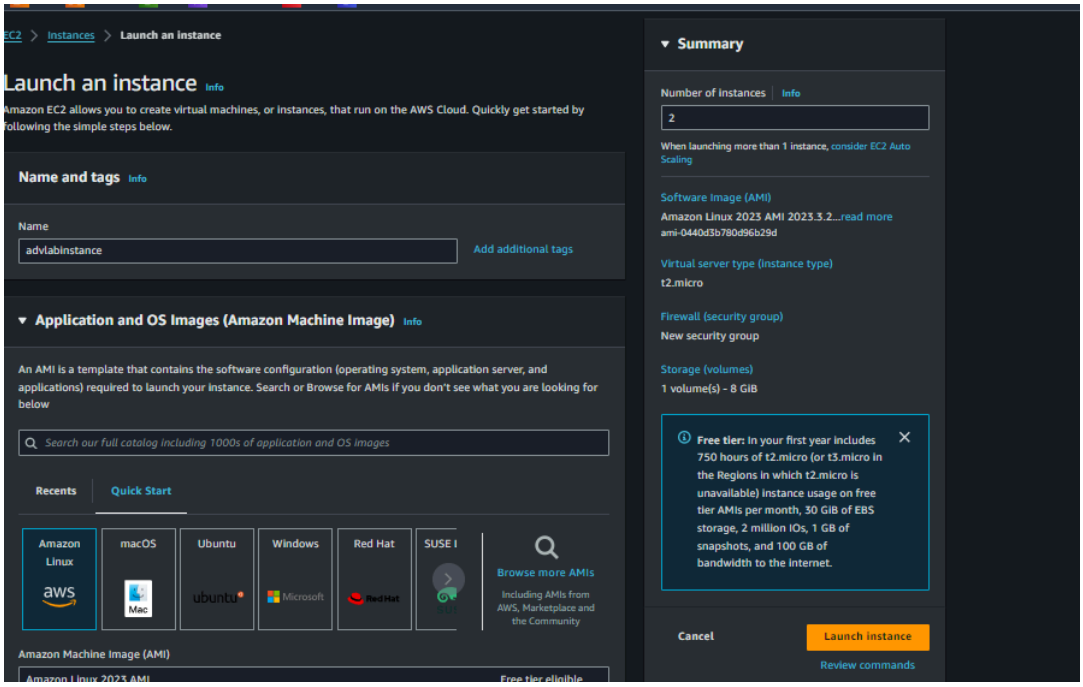
Objective: Learn how to create a scalable and highly available web application environment using Amazon EC2 instances, ELB, and ASG.

Approach:

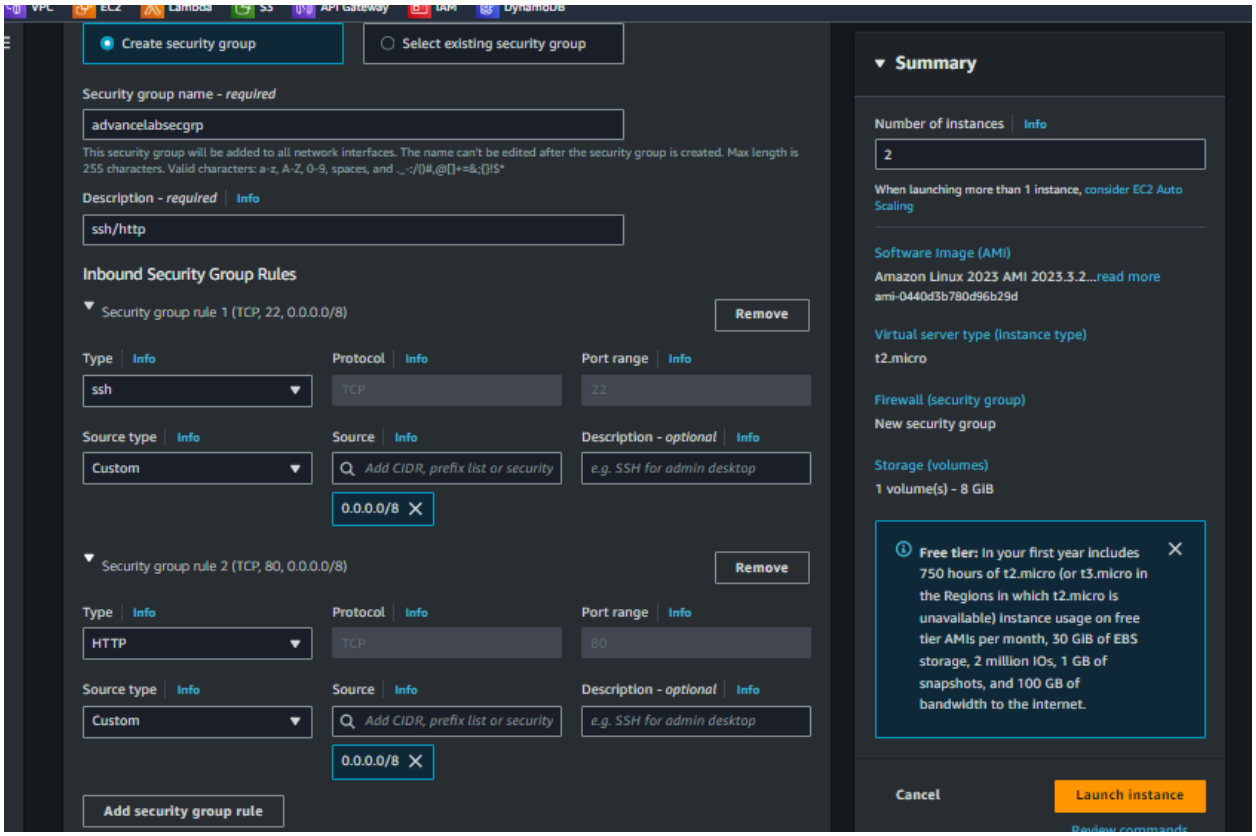
1. **Launch EC2 Instances**: Start by launching two or more EC2 instances. These instances will run a simple web application (e.g., a "Hello World" page or any basic web service).
2. **Configure Load Balancer**: Set up an Elastic Load Balancer (ELB) to distribute incoming web traffic across your EC2 instances. This step ensures high availability and fault tolerance.
3. **Set Up Auto Scaling Group (ASG)**: Create an ASG that uses the launched EC2 instances. Configure ASG policies to automatically scale the number of instances up or down based on criteria like CPU usage or network traffic.
4. **Test Your Setup**: Simulate traffic to test the scaling policies and the load balancer. Observe how ASG adds or removes instances and how ELB distributes traffic.
5. **Verify Website Functionality**: Ensure that the website hosted on EC2 instances remains accessible and functional during scaling operations.

Goal: By the end of this lab, students will have a hands-on understanding of setting up a load-balanced and auto-scaled web application using AWS services.

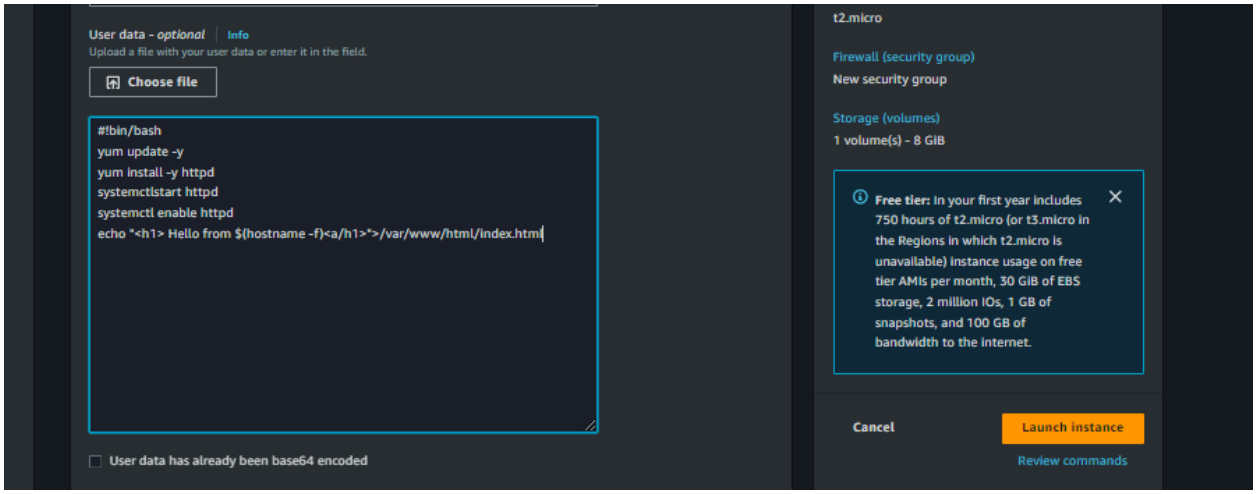
1. Launch 2 EC2 instance
Give name of instance and select application and OS Images



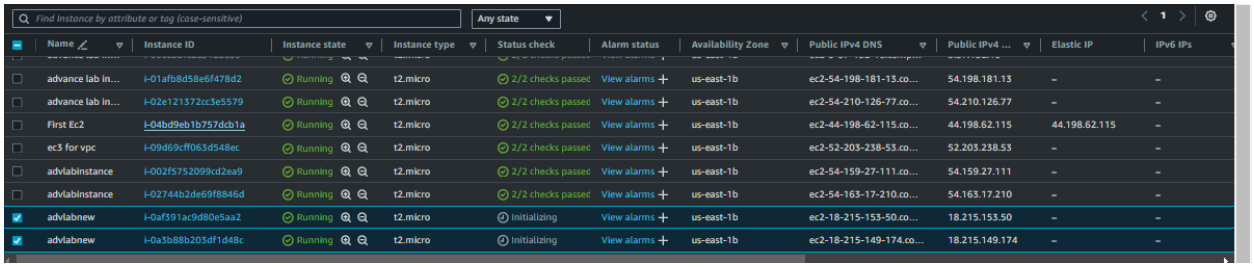
2.Create security grp:



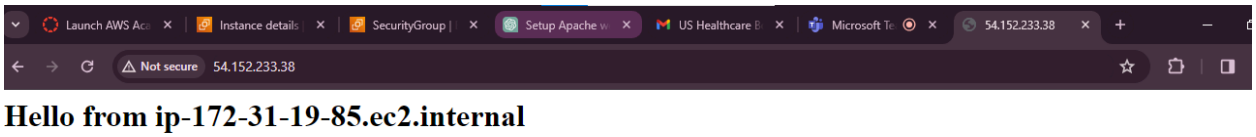
3.At Advanced Detail section below script is aTTached and rest configuration are le[as it is.



4. Instance successfully created:

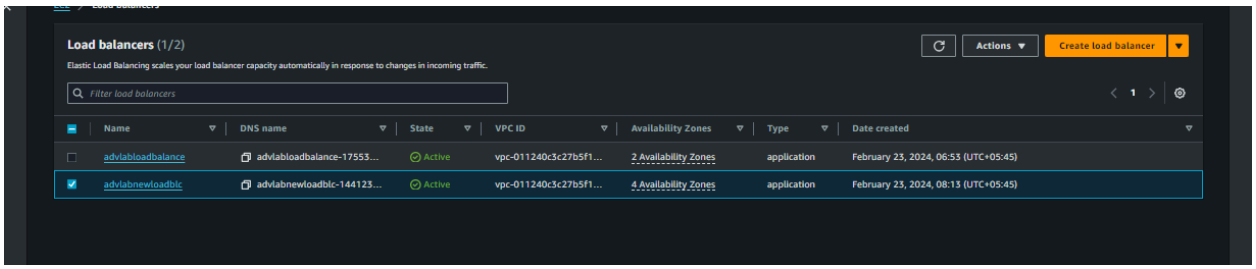


the public IPv4 address of both ec2 instances worked in web browser

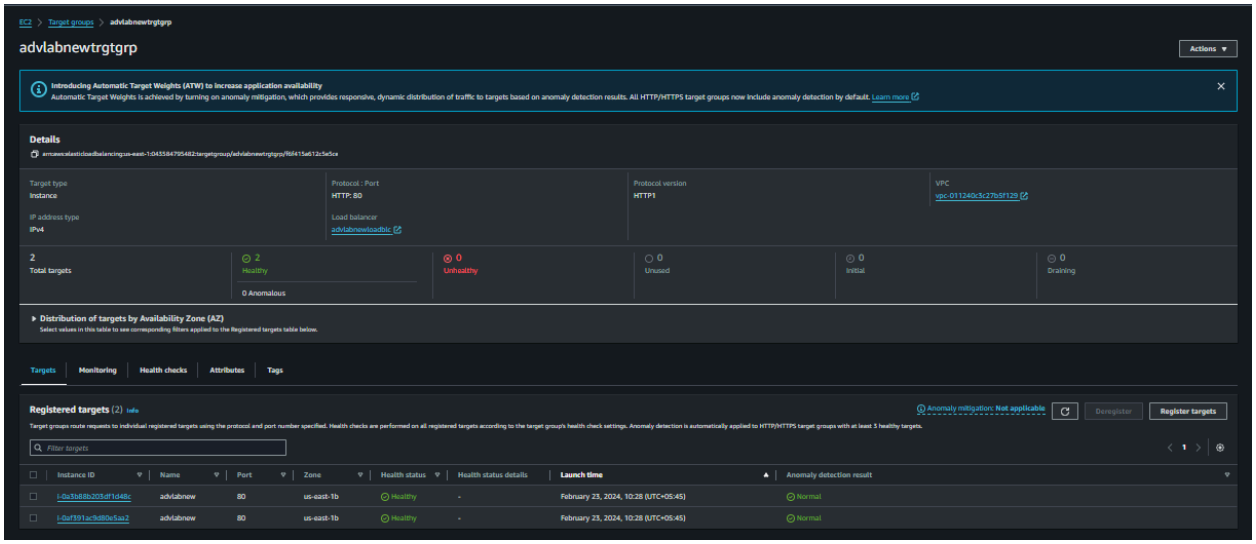


Hello from ip-172-31-19-85.ec2.internal

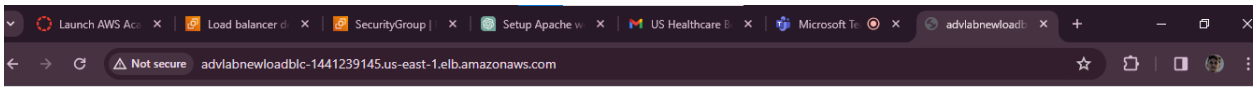
5.Navigate to load balancers screen and click on “Create load balancer” and create a new load balancer



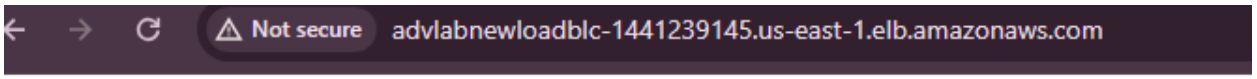
6.Basic configuration for target group is provided. Here, target group instances are chosen and for rest default configurations are kept as it is.



7.A[er that, the DNS name of load balancer is copied and accessed in web browser. When the URL is refreshed, multiple 'mes it runs different instance of the specified target group to maintain the load

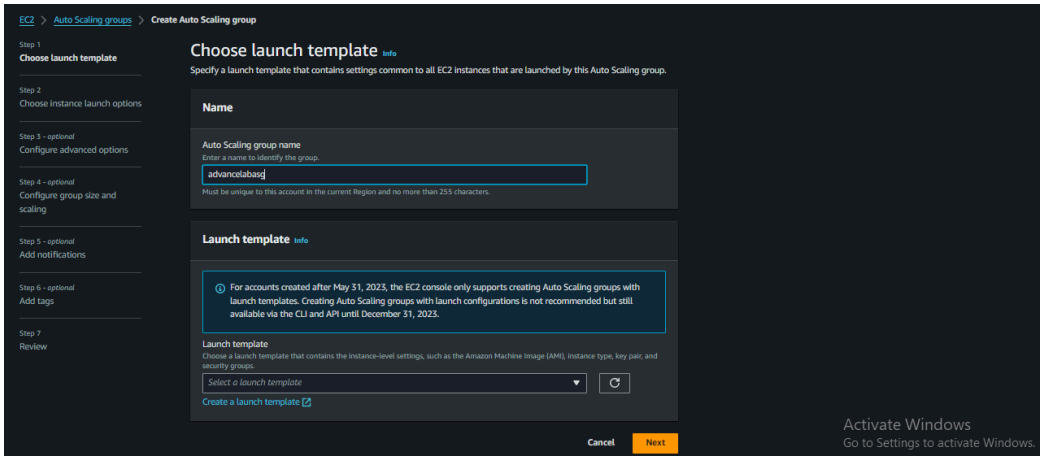


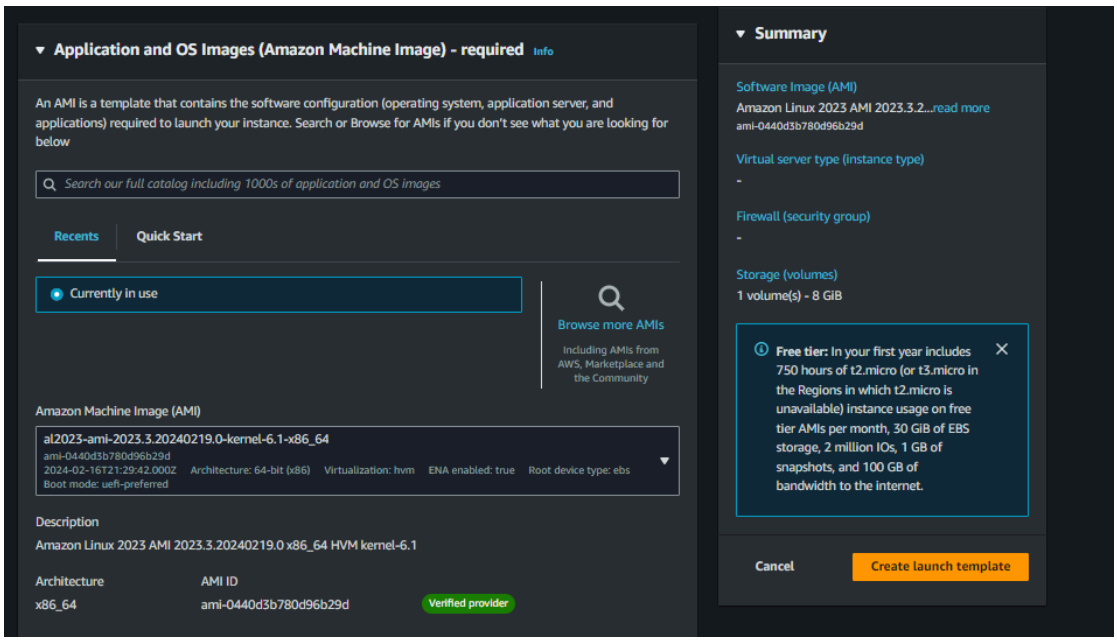
Hello from ip-172-31-21-71.ec2.internal

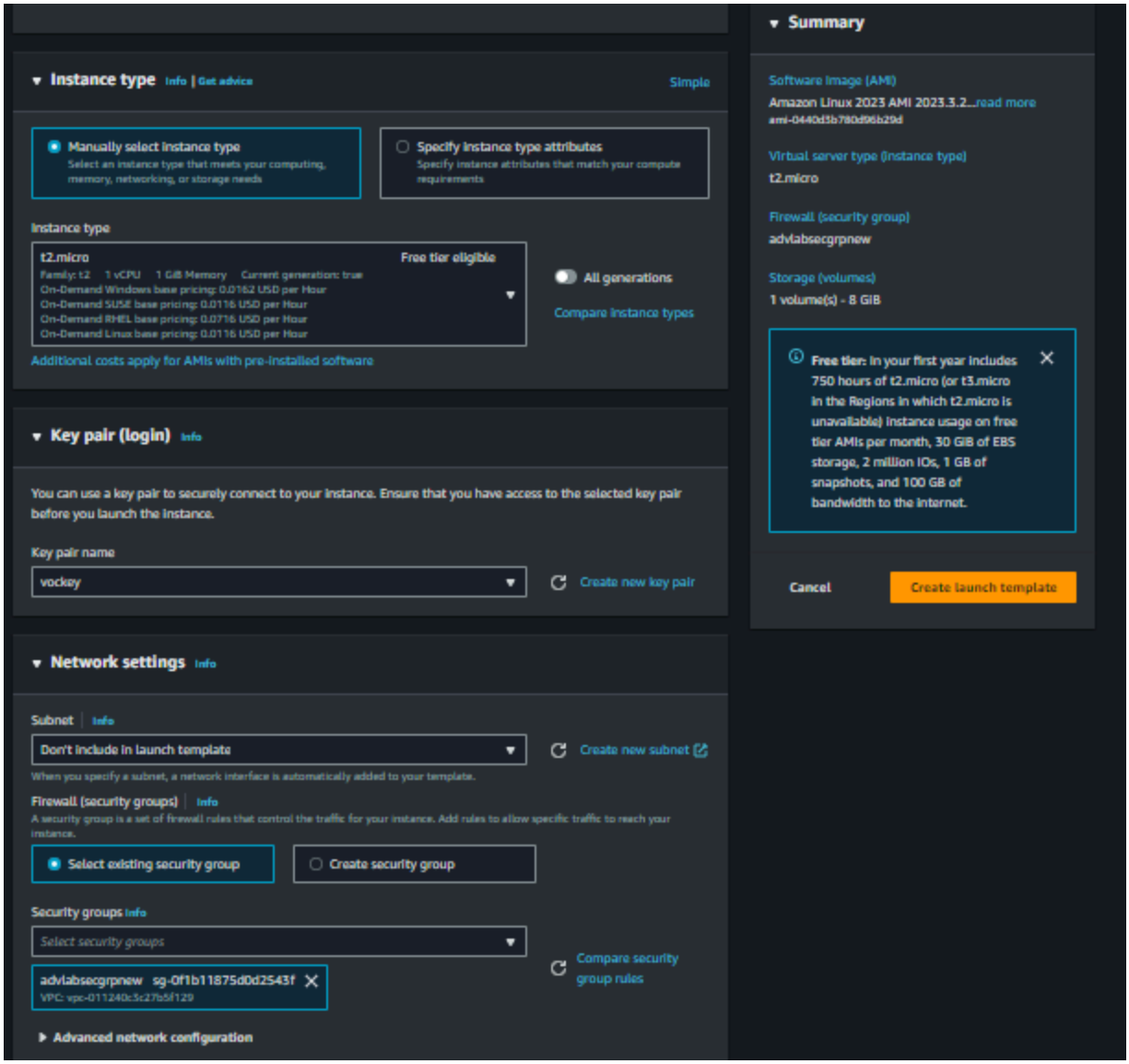


Hello from ip-172-31-19-85.ec2.internal

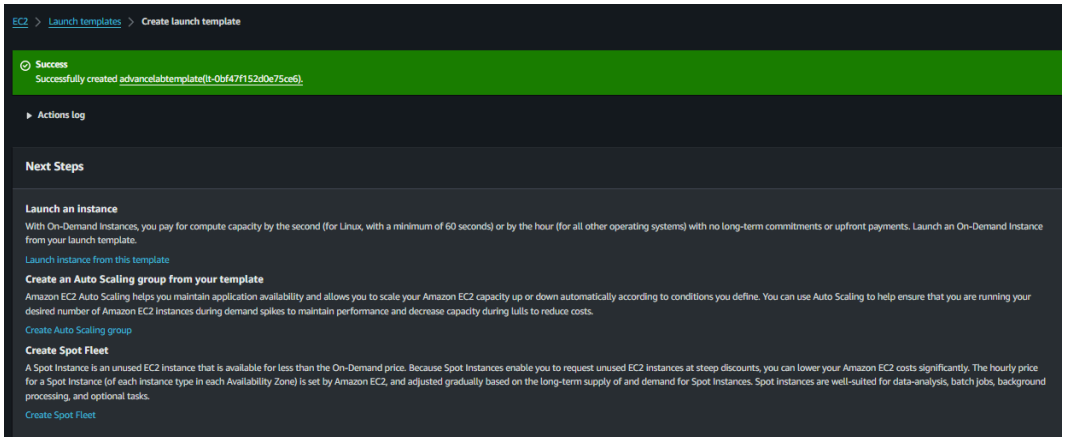
8.Create new auto scaling group.
In launch template section, “Create a launch template” is selected and new template is created


EC2 > Auto Scaling groups > Create Auto Scaling group
Step 1
Choose launch template
Step 2
Choose instance launch options
Step 3 - optional
Configure advanced options
Step 4 - optional
Configure group size and scaling
Step 5 - optional
Add notifications
Step 6 - optional
Add tags
Step 7
Review
Choose launch template Info
Specify a launch template that contains settings common to all EC2 instances that are launched by this Auto Scaling group.
Name
Auto Scaling group name
Enter a name to identify this group.
advancelabasg
Must be unique to this account in the current Region and no more than 255 characters.
Launch template Info
For accounts created after May 31, 2023, the EC2 console only supports creating Auto Scaling groups with launch templates. Creating Auto Scaling groups with launch configurations is not recommended but still available via the CLI and API until December 31, 2023.
Launch template
Choose a launch template that contains the instance-level settings, such as the Amazon Machine Image (AMI), instance type, key pair, and security groups.
Select a launch template
Create a launch template
Cancel
Next
Activate Windows
Go to Settings to activate Windows.

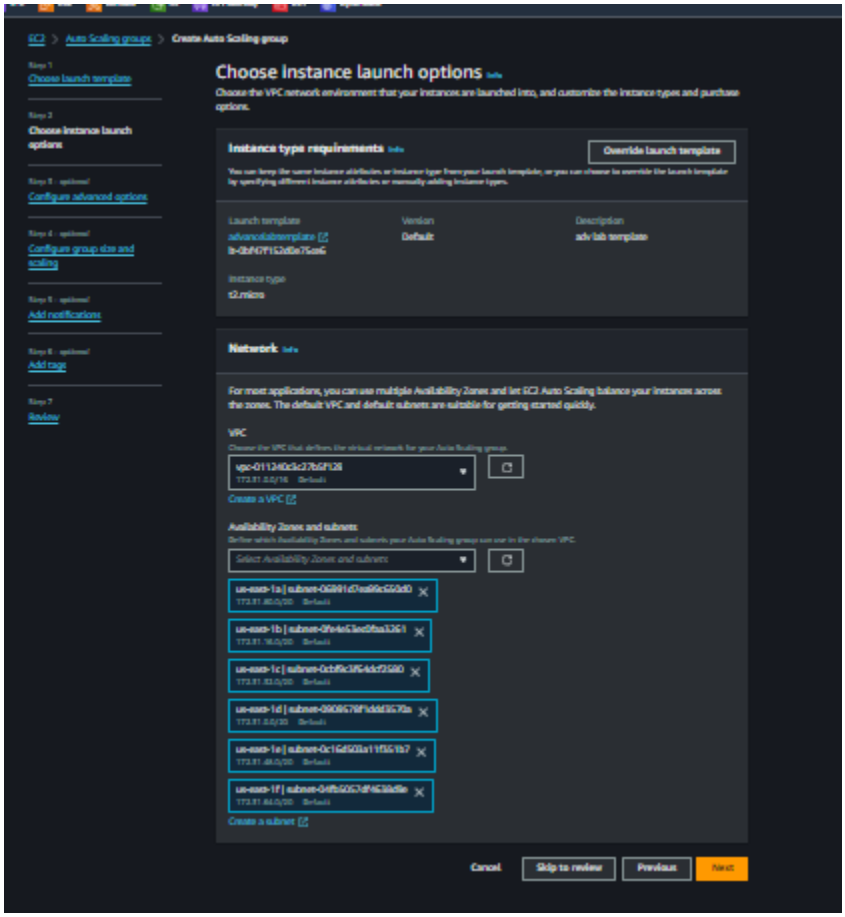

Application and OS Images (Amazon Machine Image) - required Info
An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below
Search our full catalog including 1000s of application and OS images
Recents
Quick Start
Currently in use
Browse more AMIs
Including AMIs from AWS, Marketplace and the Community
Amazon Machine Image (AMI)
al2023-ami-2023.3.20240219.0-kernel-6.1-x86_64
ami-0440d3b780d96b29d
2024-02-16T21:29:42.000Z Architecture: 64-bit (x86) Virtualization: hvm ENA enabled: true Root device type: ebs
Boot mode: uefi-preferred
Description
Amazon Linux 2023 AMI 2023.3.20240219.0 x86_64 HVM kernel-6.1
Architecture
x86_64
AMI ID
ami-0440d3b780d96b29d
Verified provider
Summary
Software Image (AMI)
Amazon Linux 2023 AMI 2023.3.2...read more
ami-0440d3b780d96b29d
Virtual server type (instance type)
-
Firewall (security group)
-
Storage (volumes)
1 volume(s) - 8 GiB
Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth to the internet.
Cancel
Create launch template



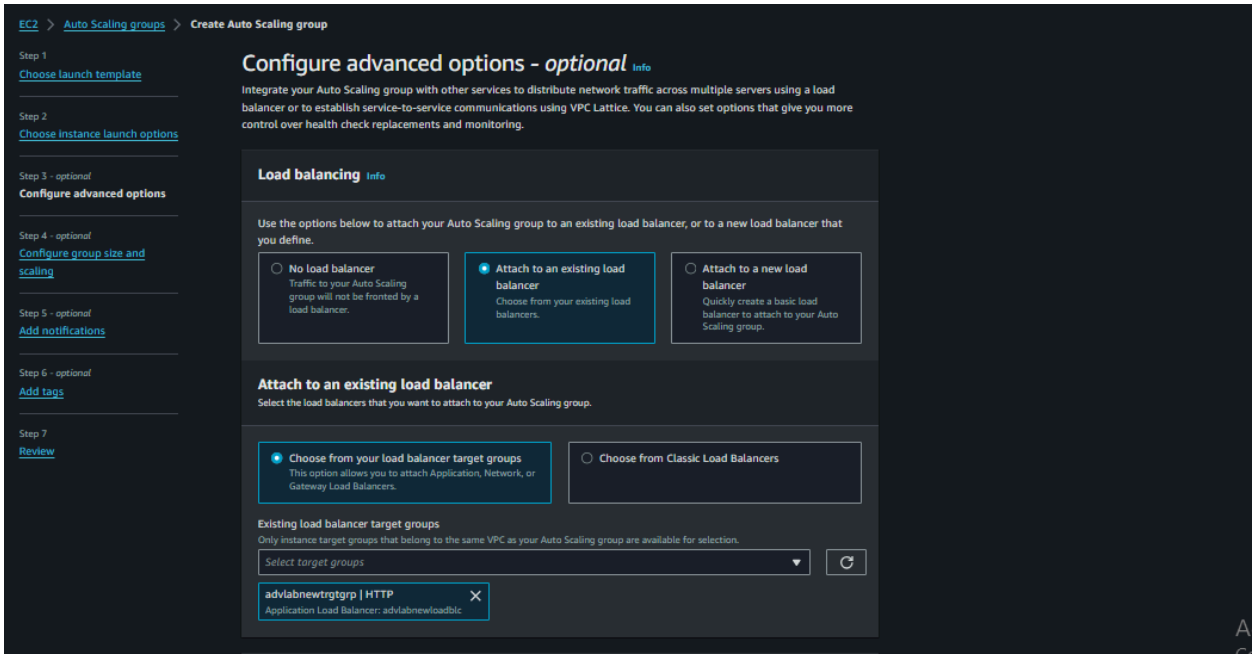
9.Afer confirming the configura'on, template is launched .



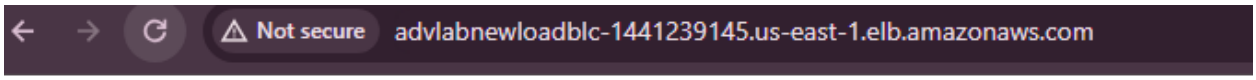
10.Now, the configuration of auto scaling is continued. Here newly created launch template is selected
In network section default VPC and all the subnets are selected



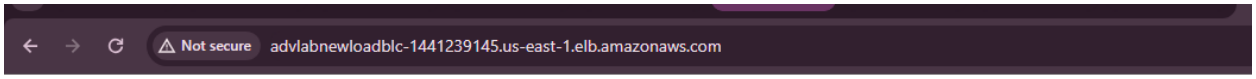
11.Autoscaling group created:



12. As we have defined 2 desired capacity, 2 EC2 instance of same target group is always maintained based on the created launch template.
When DNS of load balancer is refreshed even for multiple 'mes the, it can be observed that EC2 instance is working



Hello from ip-172-31-19-85.ec2.internal



Hello from ip-172-31-21-71.ec2.internal

Part 2: Hosting a Static Portfolio Website on S3

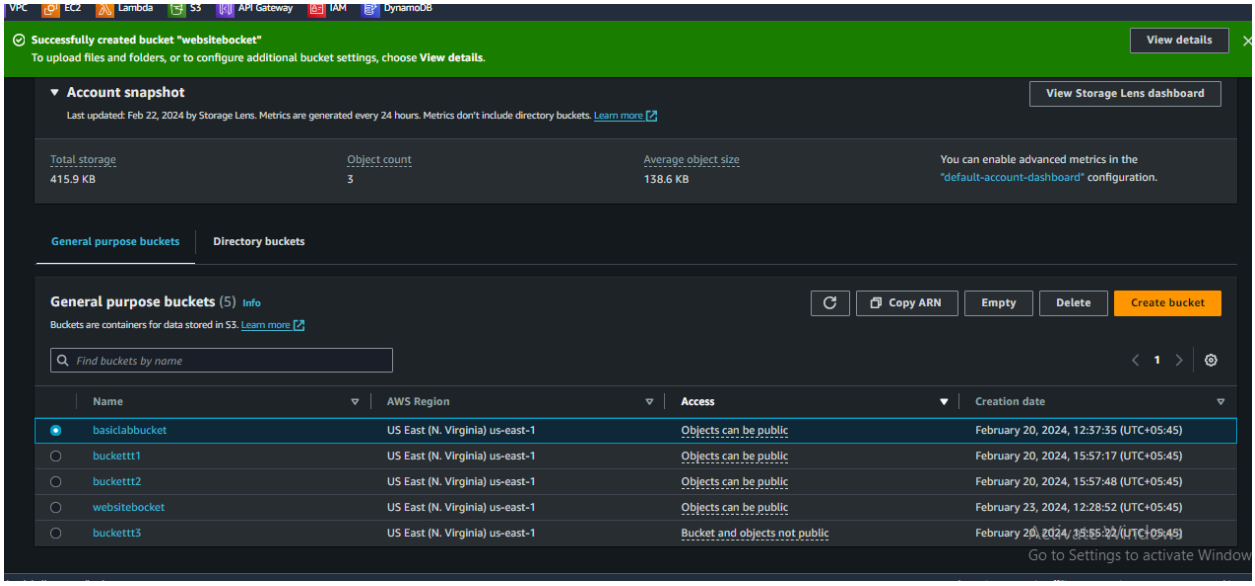
Objective: Learn to host a static website (such as a personal portfolio) on Amazon S3.

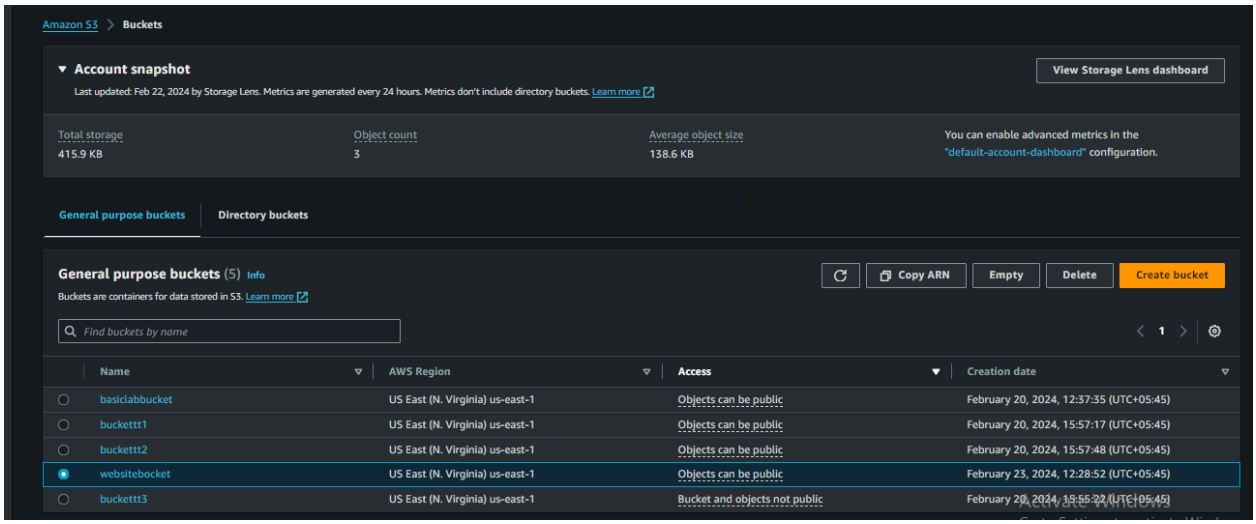
Approach:

1. **Create an S3 Bucket**: Start by creating a new S3 bucket. Configure the bucket for website hosting, which includes setting permissions to make the content publicly accessible.
2. **Upload Website Files**: Upload the static files of your portfolio website (HTML, CSS, JavaScript, images) to the S3 bucket.
3. **Configure DNS**: Use Amazon Route 53 or another DNS service to point a domain name to the S3 bucket. This makes the website accessible via a user-friendly URL.
4. **Enable Additional Features** (Optional): Implement features like HTTPS for secure access and CloudFront for content delivery optimization.

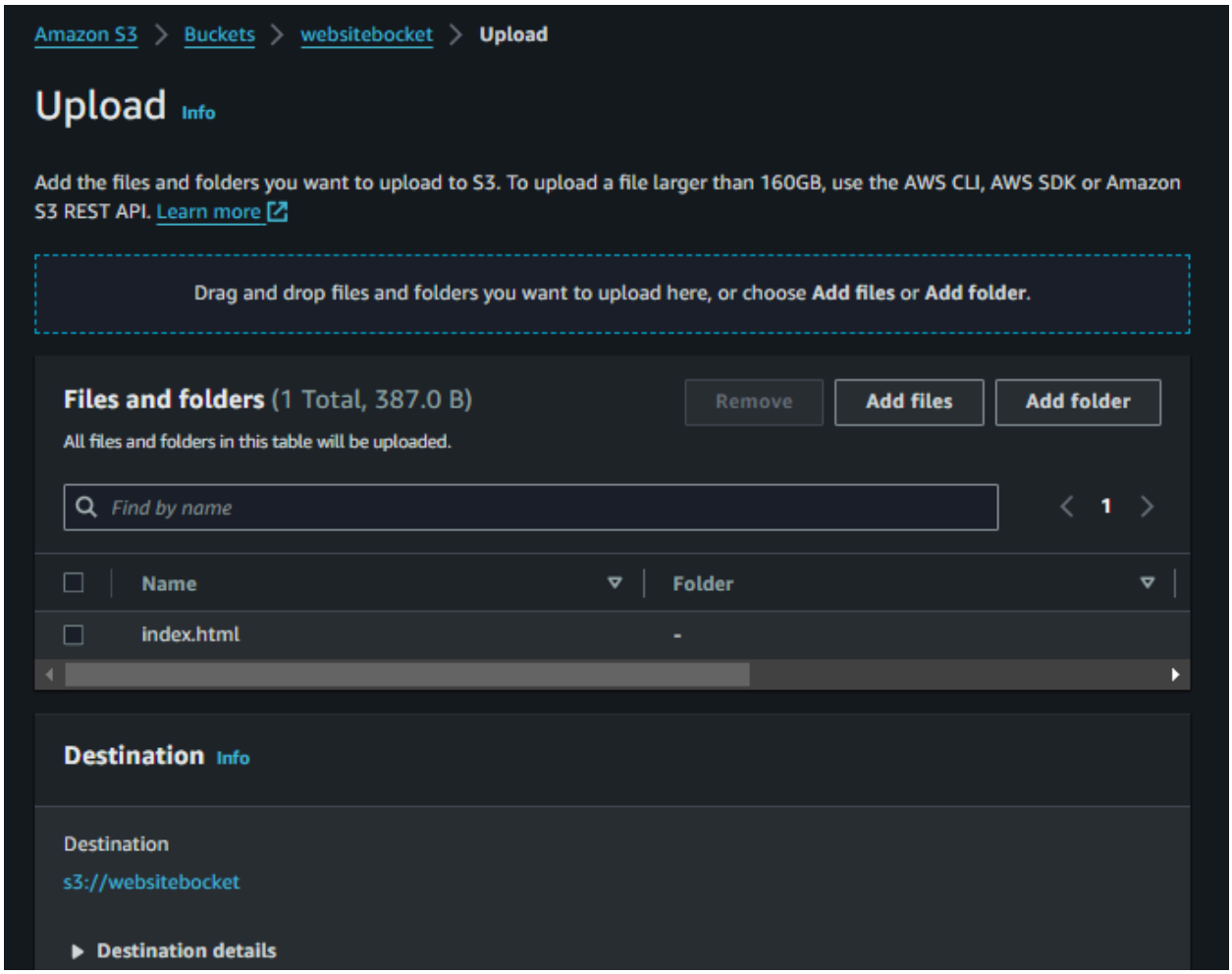
Goal: Students will understand how to use S3 for hosting static websites, manage bucket permissions, and integrate with other AWS services for a complete web hosting solution.

1. Create S3 bucket:

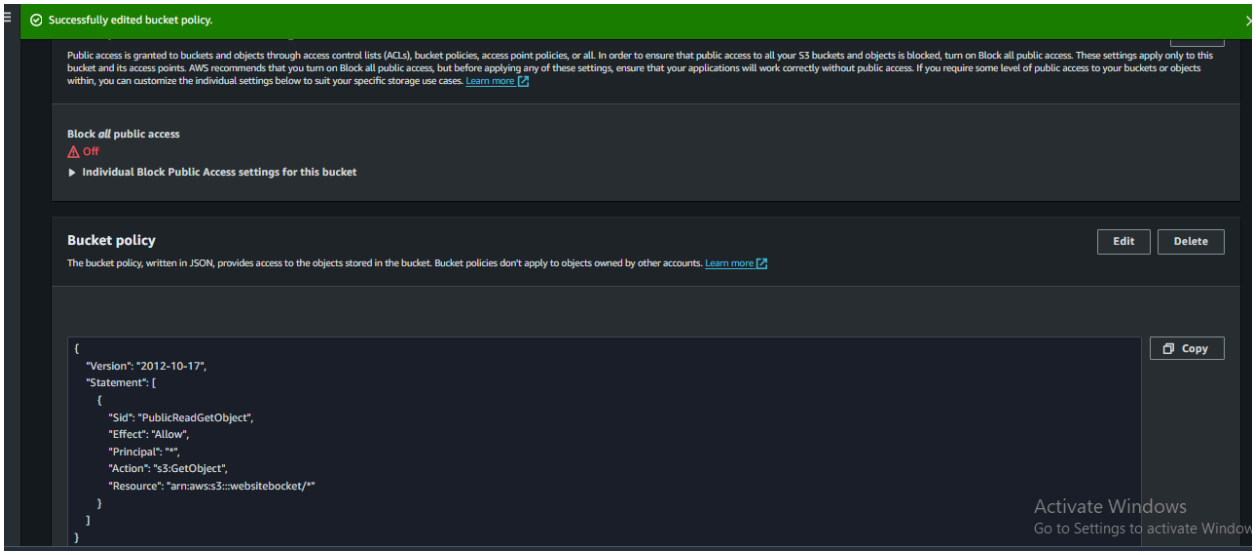




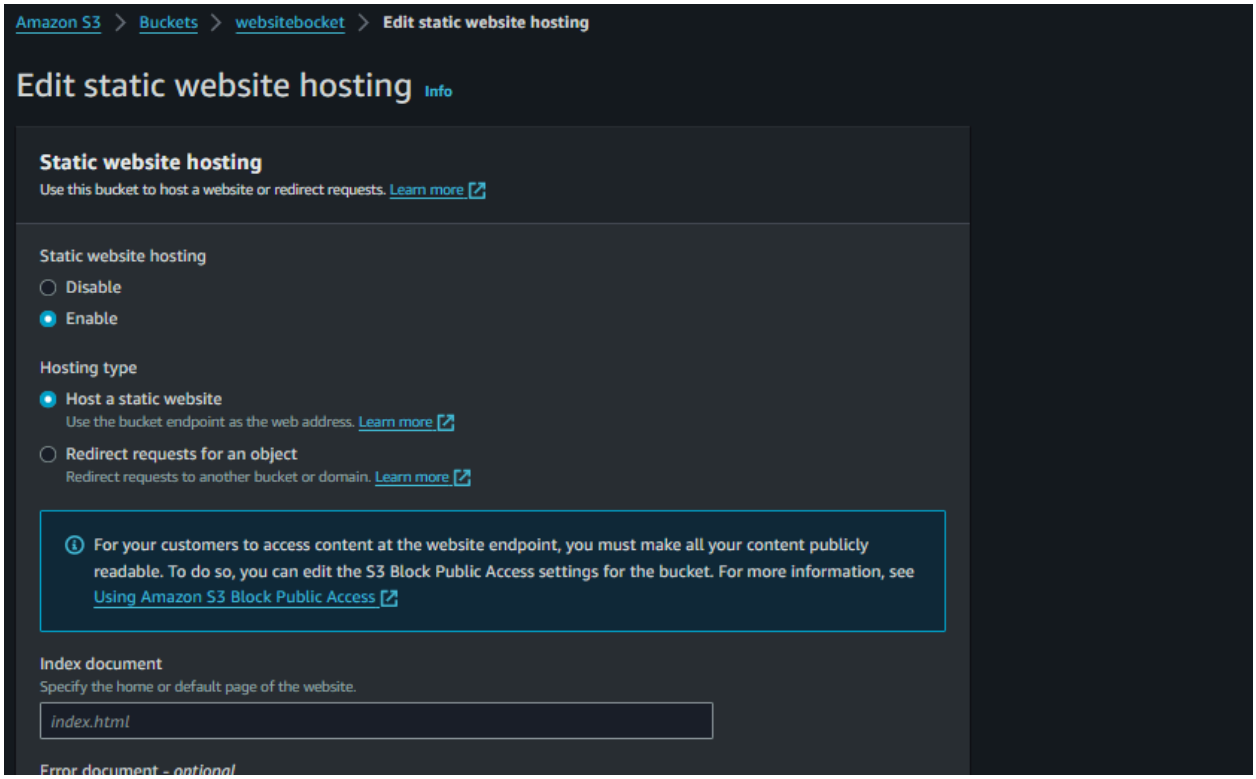
2. Upload index.html



3. Edit bucket policy

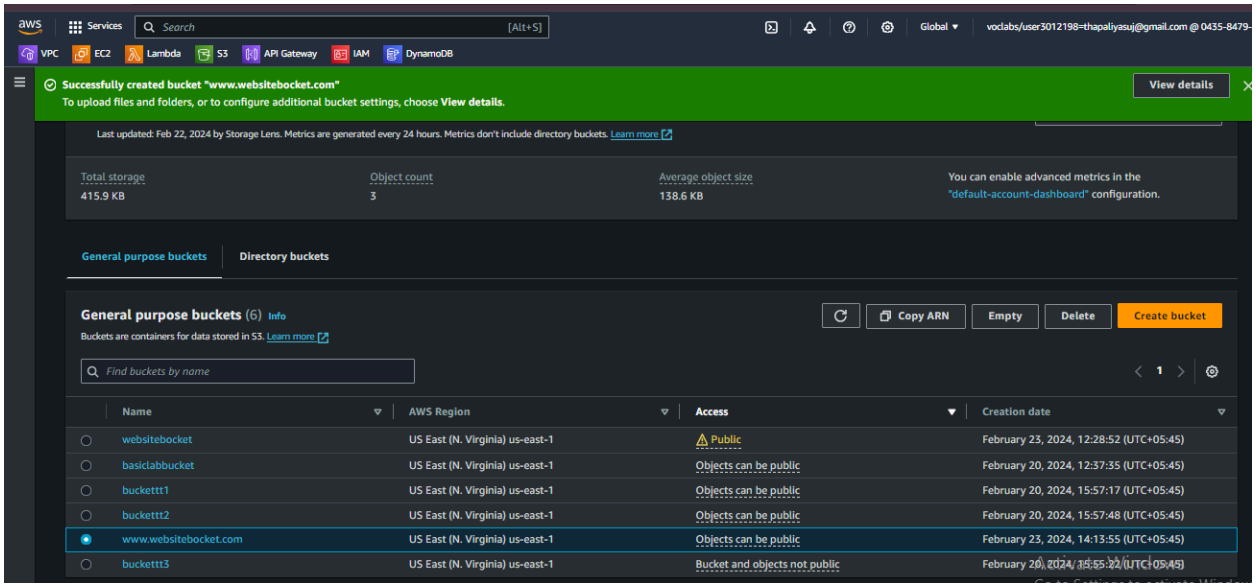


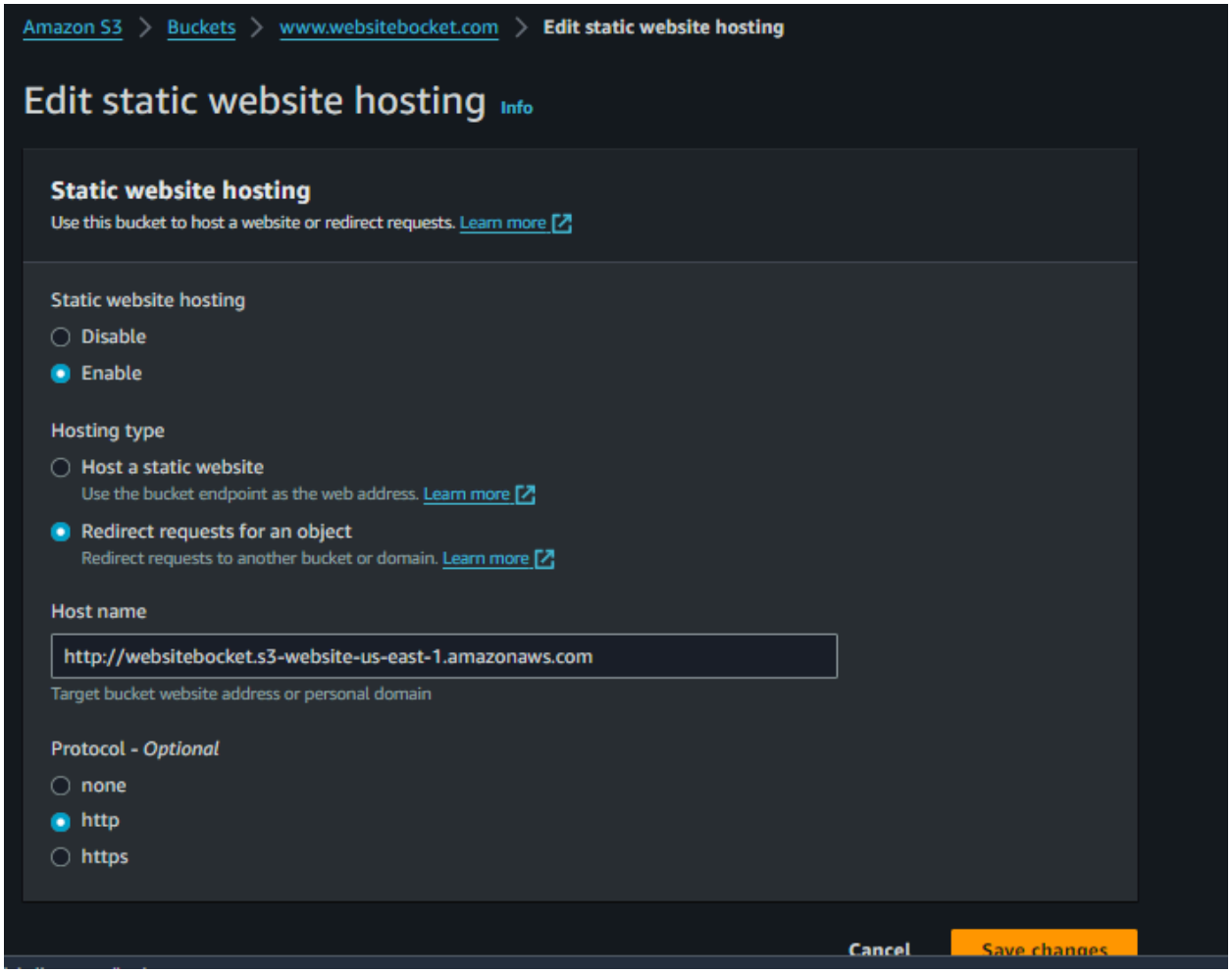
4. Edit static web hosting:

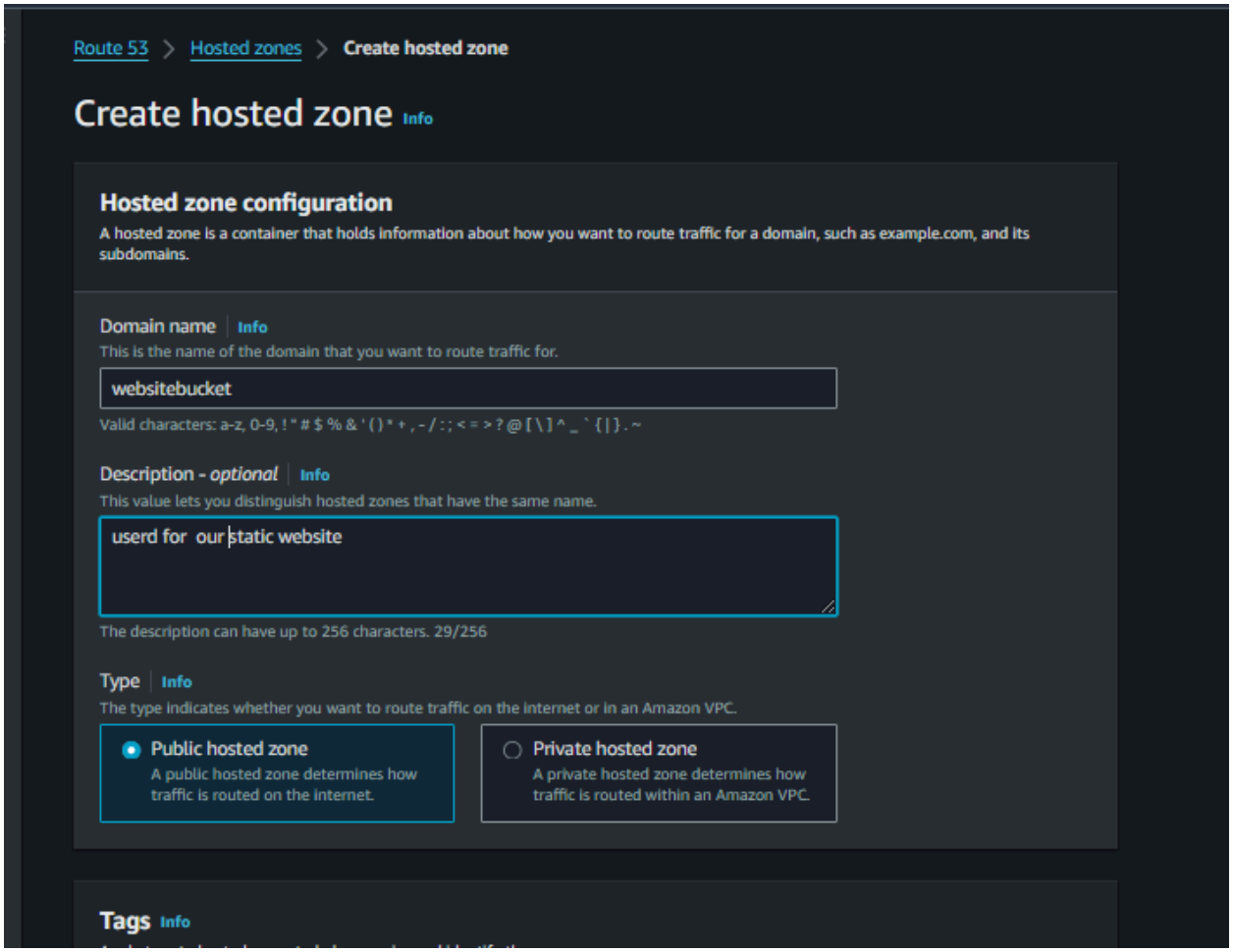


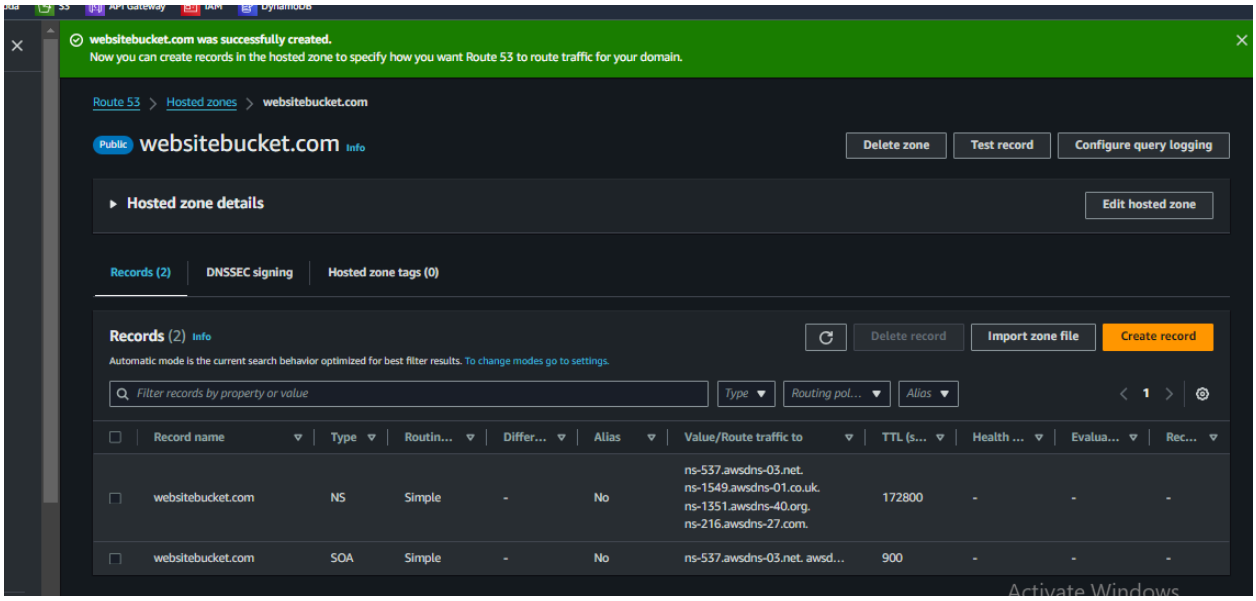
Not secure websitebocket.s3-website-us-east-1.amazonaws.com

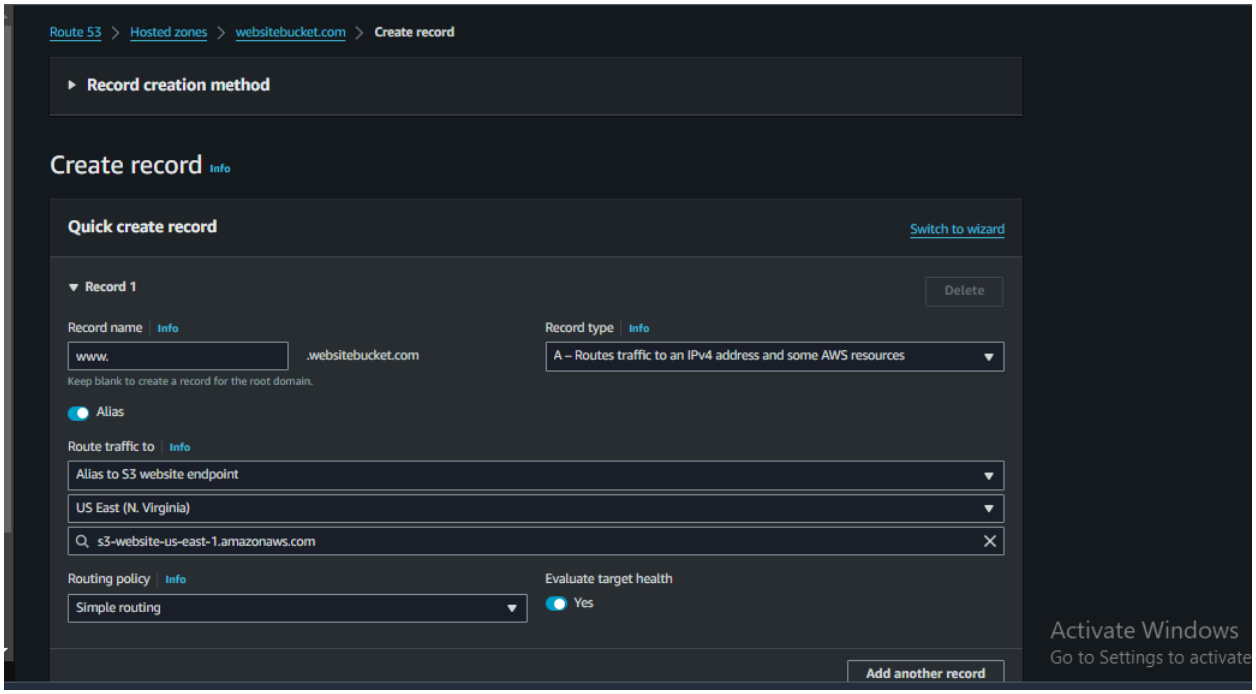
This is a simple HTML + CSS template!




Amazon S3 > Buckets > www.websitebocket.com > Edit static website hosting
Edit static website hosting Info
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
http://websitebocket.s3-website-us-east-1.amazonaws.com
Target bucket website address or personal domain
Protocol - Optional
none
http
https
Cancel
Save changes


Route 53 > Hosted zones > Create hosted zone
Create hosted zone Info
Hosted zone configuration
A hosted zone is a container that holds information about how you want to route traffic for a domain, such as example.com, and its subdomains.
Domain name Info
This is the name of the domain that you want to route traffic for.
websitebucket
Valid characters: a-z, 0-9, ! " # $ % & ' ( ) * + , - / : ; < = > ? @ [\ ] ^ _ ` { | } . ~
Description - optional Info
This value lets you distinguish hosted zones that have the same name.
userd for our static website
The description can have up to 256 characters. 29/256
Type Info
The type indicates whether you want to route traffic on the internet or in an Amazon VPC.
Public hosted zone
A public hosted zone determines how traffic is routed on the internet.
Private hosted zone
A private hosted zone determines how traffic is routed within an Amazon VPC.
Tags Info


websitebucket.com was successfully created.
Now you can create records in the hosted zone to specify how you want Route 53 to route traffic for your domain.
Route 53 > Hosted zones > websitebucket.com
Public websitebucket.com Info
Delete zone
Test record
Configure query logging
Hosted zone details
Edit hosted zone
Records (2)
DNSSEC signing
Hosted zone tags (0)
Records (2) Info
Delete record
Import zone file
Create record
Automatic mode is the current search behavior optimized for best filter results. To change modes go to settings.
Filter records by property or value
Type
Routing pol...
Alias
Record name
Type
Routin...
Differ...
Alias
Value/Route traffic to
TTL (s...
Health ...
Evalua...
Rec...
websitebucket.com
NS
Simple
-
No
ns-537.awsdns-03.net.
ns-1549.awsdns-01.co.uk.
ns-1351.awsdns-40.org.
ns-216.awsdns-27.com.
172800
websitebucket.com
SOA
Simple
-
No
ns-537.awsdns-03.net. awsd...
900
Activate Windows


Route 53 > Hosted zones > websitebucket.com > Create record
Record creation method
Create record Info
Quick create record
Switch to wizard
Record 1
Delete
Record name Info
www.
.websitebucket.com
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 S3 website endpoint
US East (N. Virginia)
s3-website-us-east-1.amazonaws.com
Routing policy Info
Simple routing
Evaluate target health
Yes
Add another record
Activate Windows
Go to Settings to activate