AWS PROJECTS AND TASKS

1. Load balancing using spot instances in online shopping

Agenda:

- **Spot instance**
- **Launch Template**
- **Auto Scaling Group**
- **Load Balancer**
- **Target Groups**

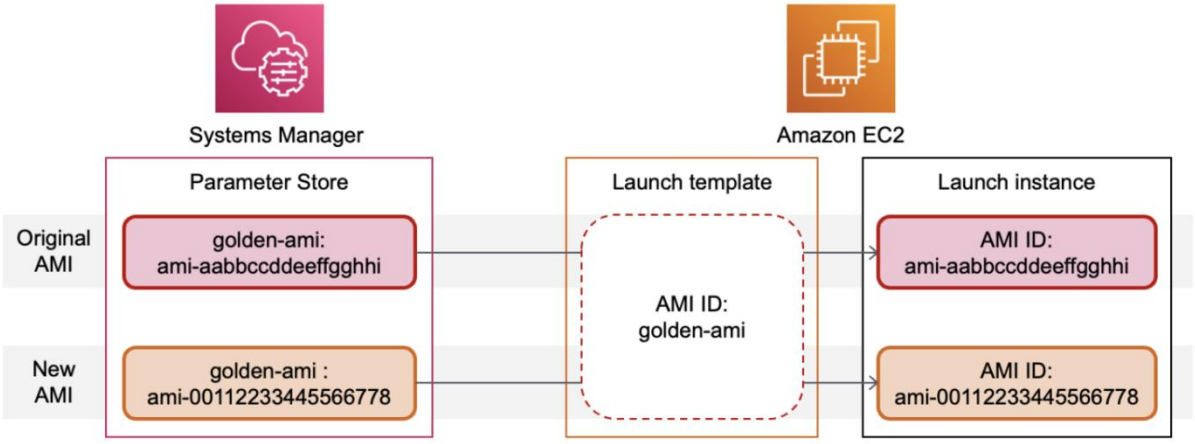
Real life examples of Online shopping.:

- **Spot instances:**

These instances are the cheapest one and good for small duration like few hours to few days only which works on bid pricing.If we bid for the instance then the instance should be available within the bid price or if bid pricing goes high then instance would be automatically terminated by aws.These instances are good for users who wants high processing power at low price and for minimum duration.

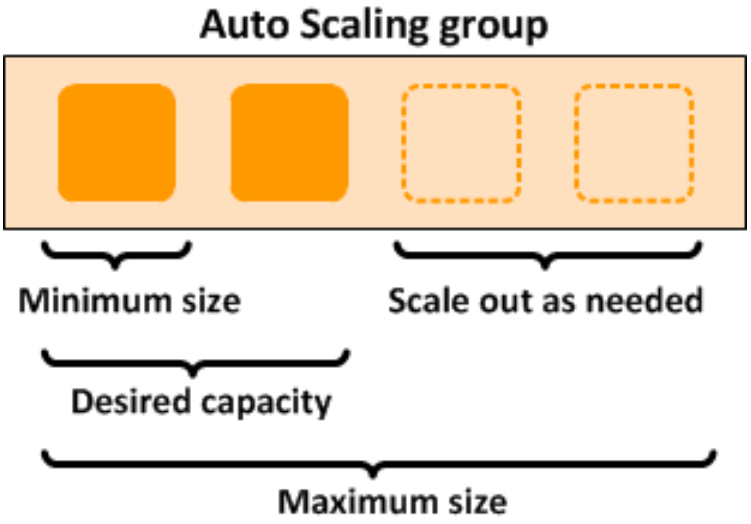
- Launch Templates:

Simplify and streamline the process of launching EC2 instances in AWS



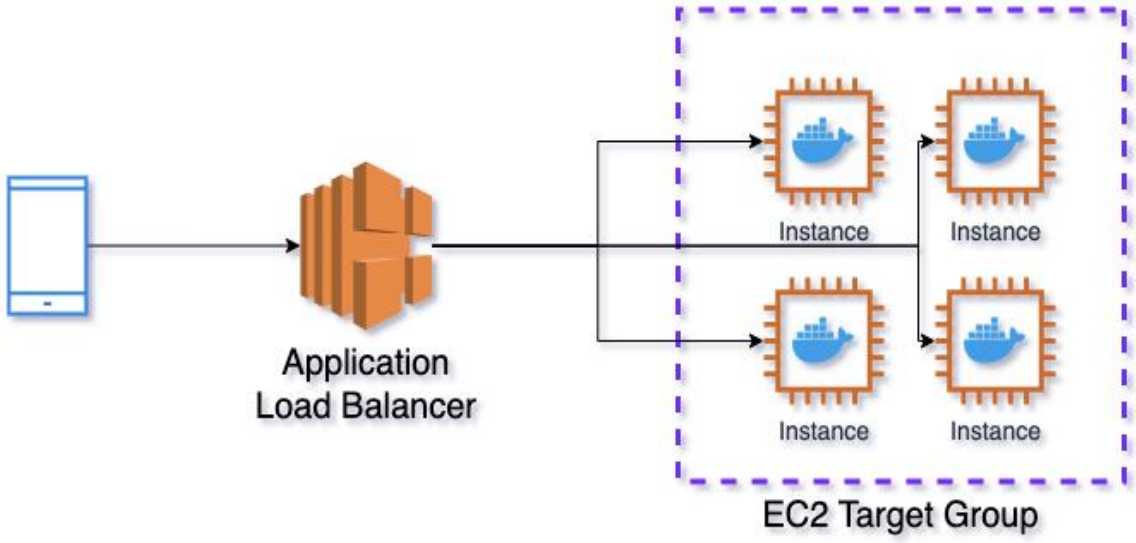
- Auto scaling group:

Auto scaling group can automatically add or remove instances based on traffic load and requirements.



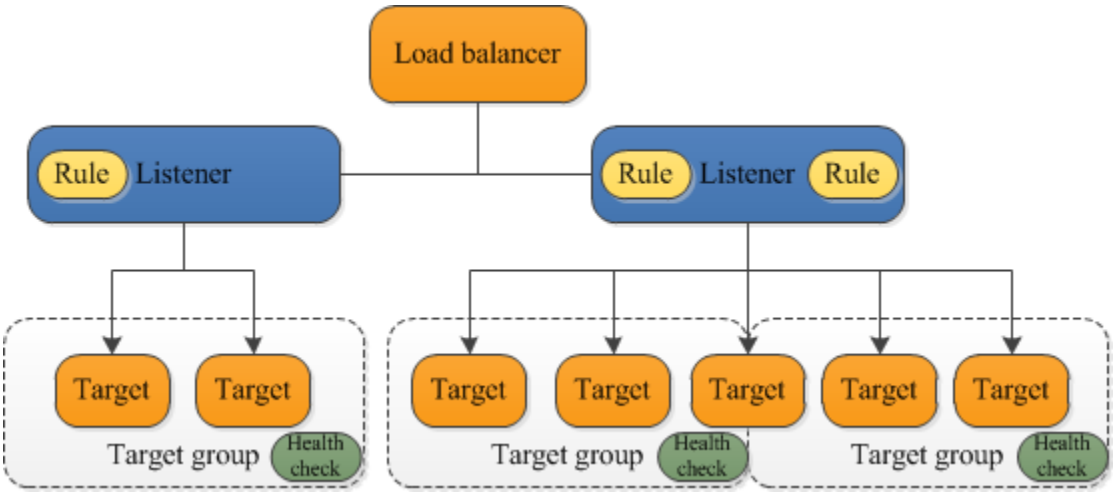
- Load Balancer:

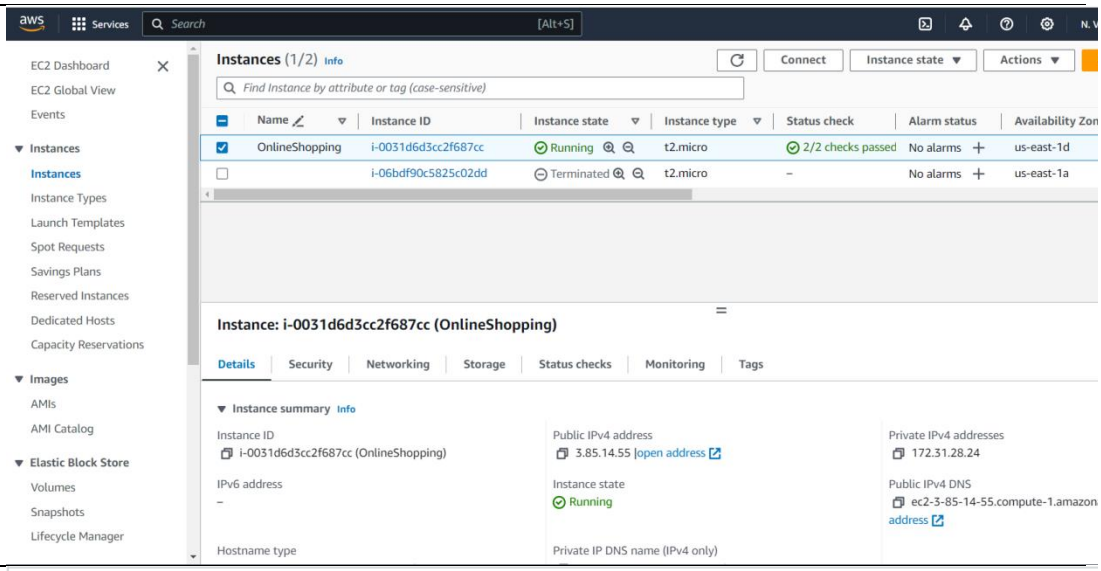
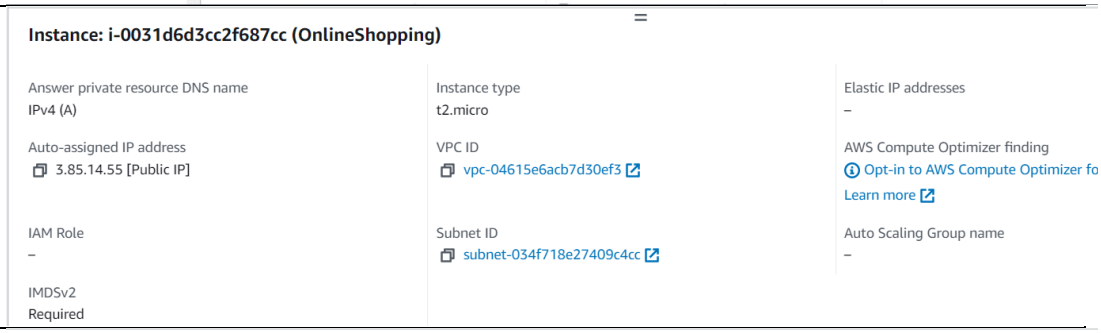
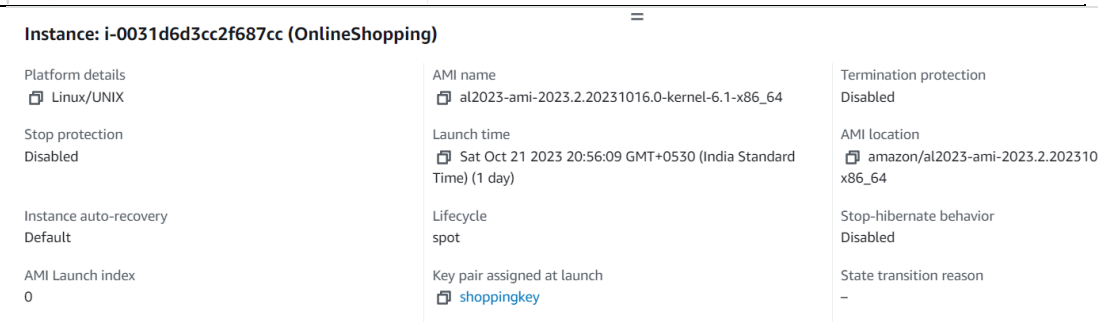
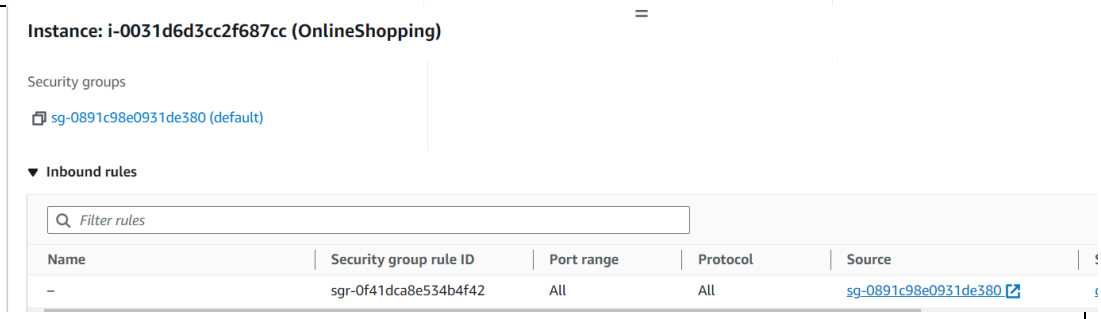
Network device or software application that distributes incoming network traffic across multiple servers or resources to ensure that no single server becomes overloaded.

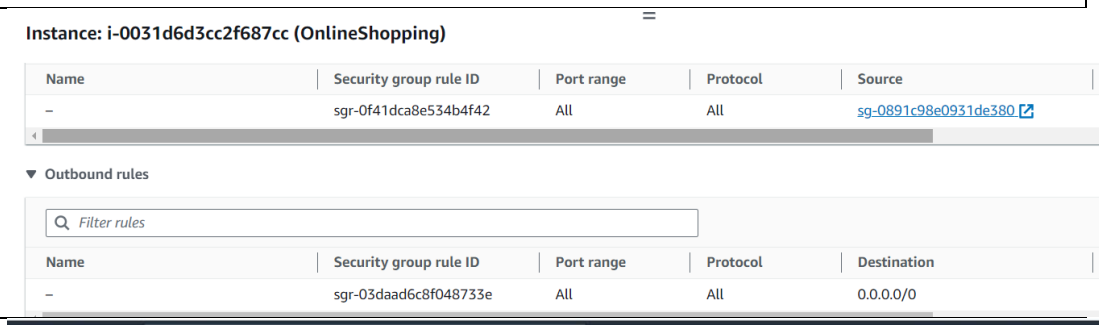
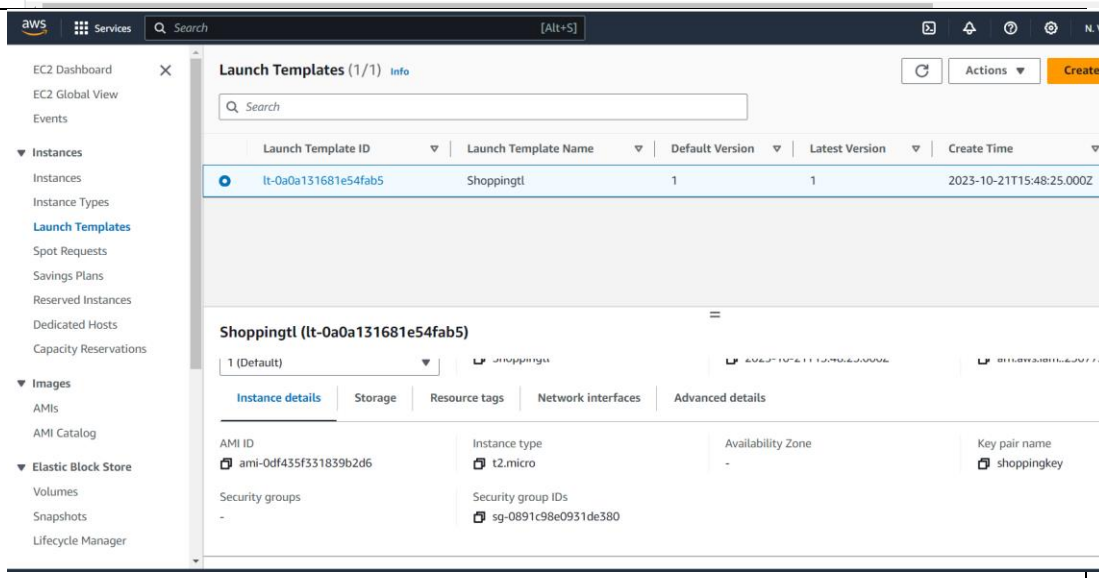
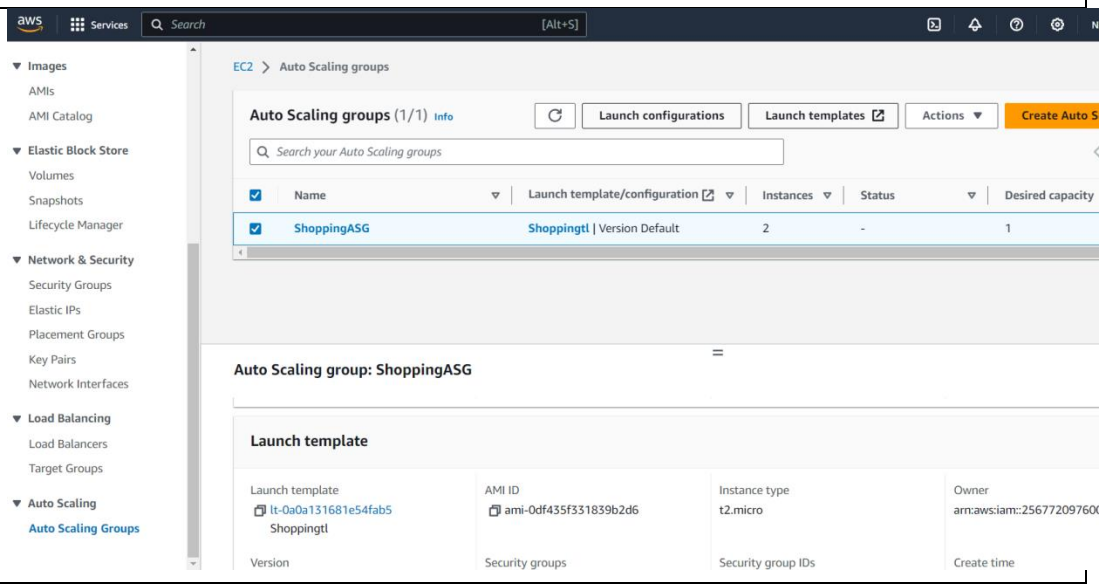


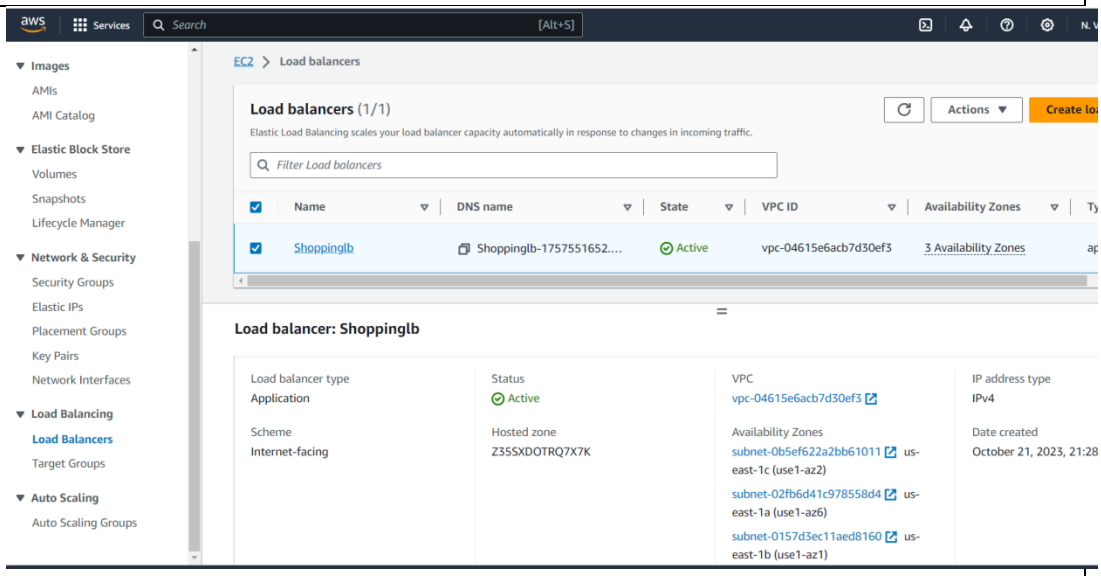
- Target groups:

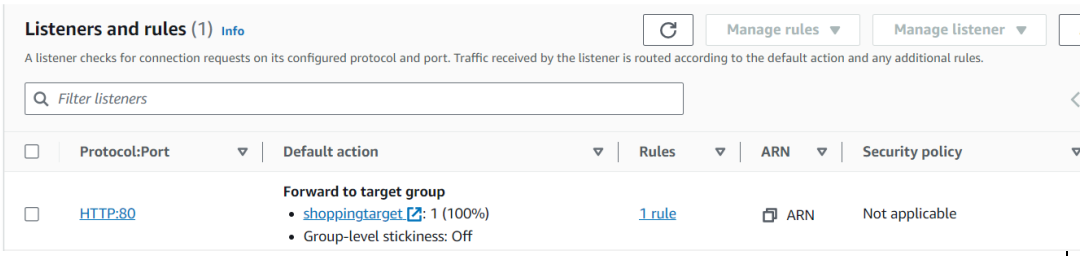
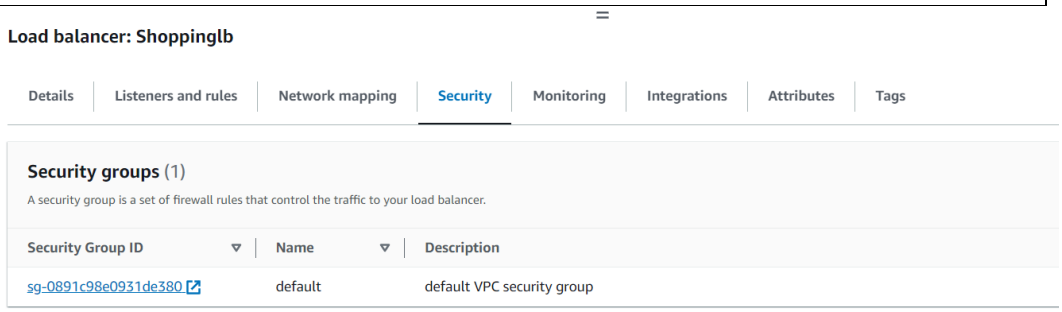
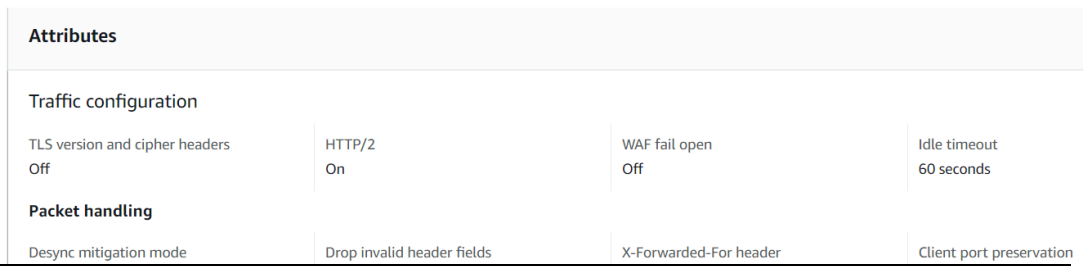
are the instances which are part of the load balancing pool.
It routes traffic from the load balancer to set of registered targets

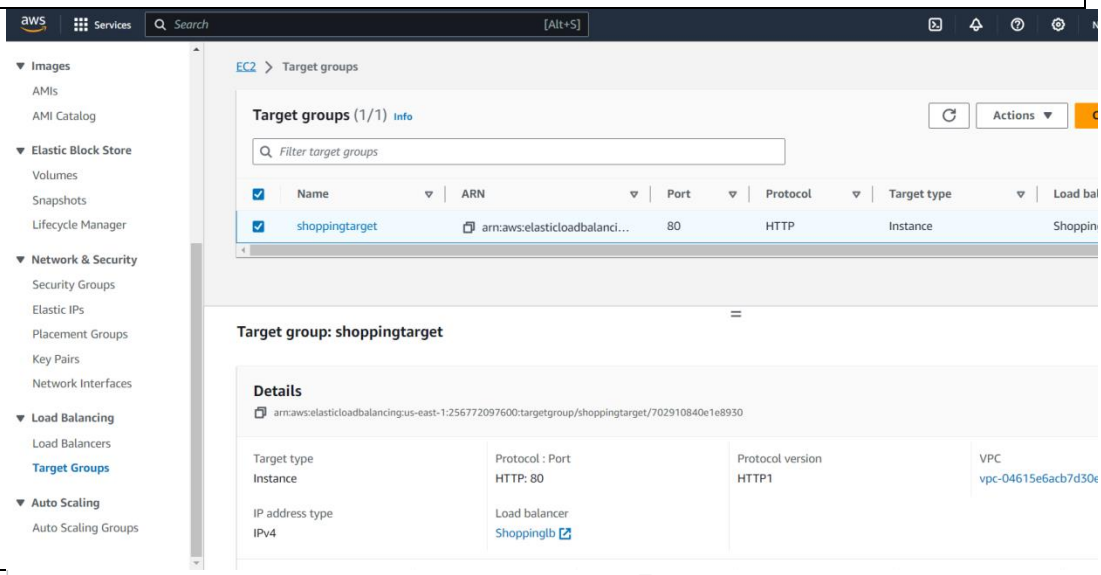
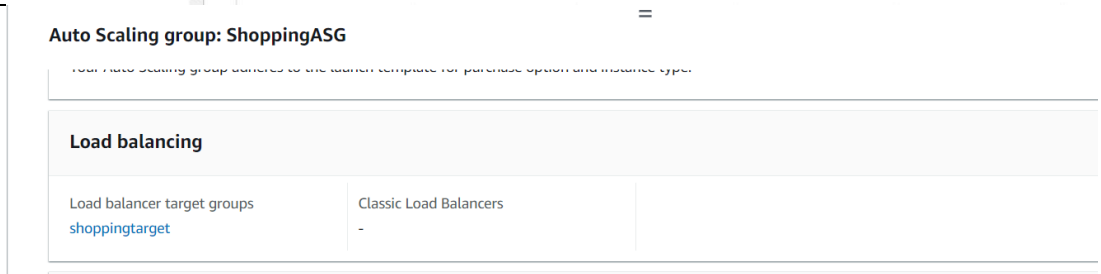
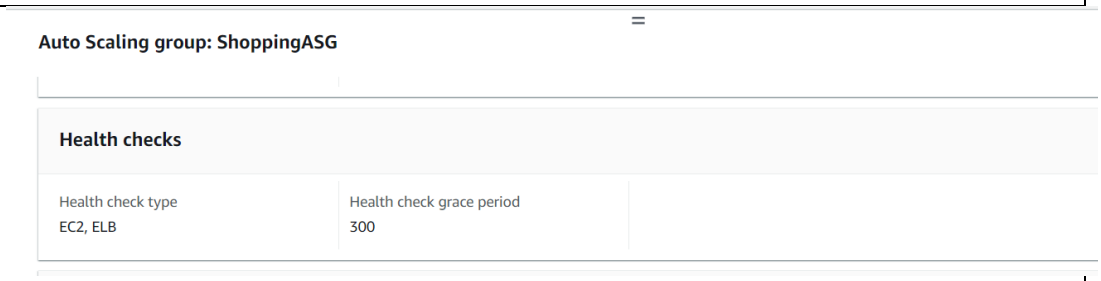


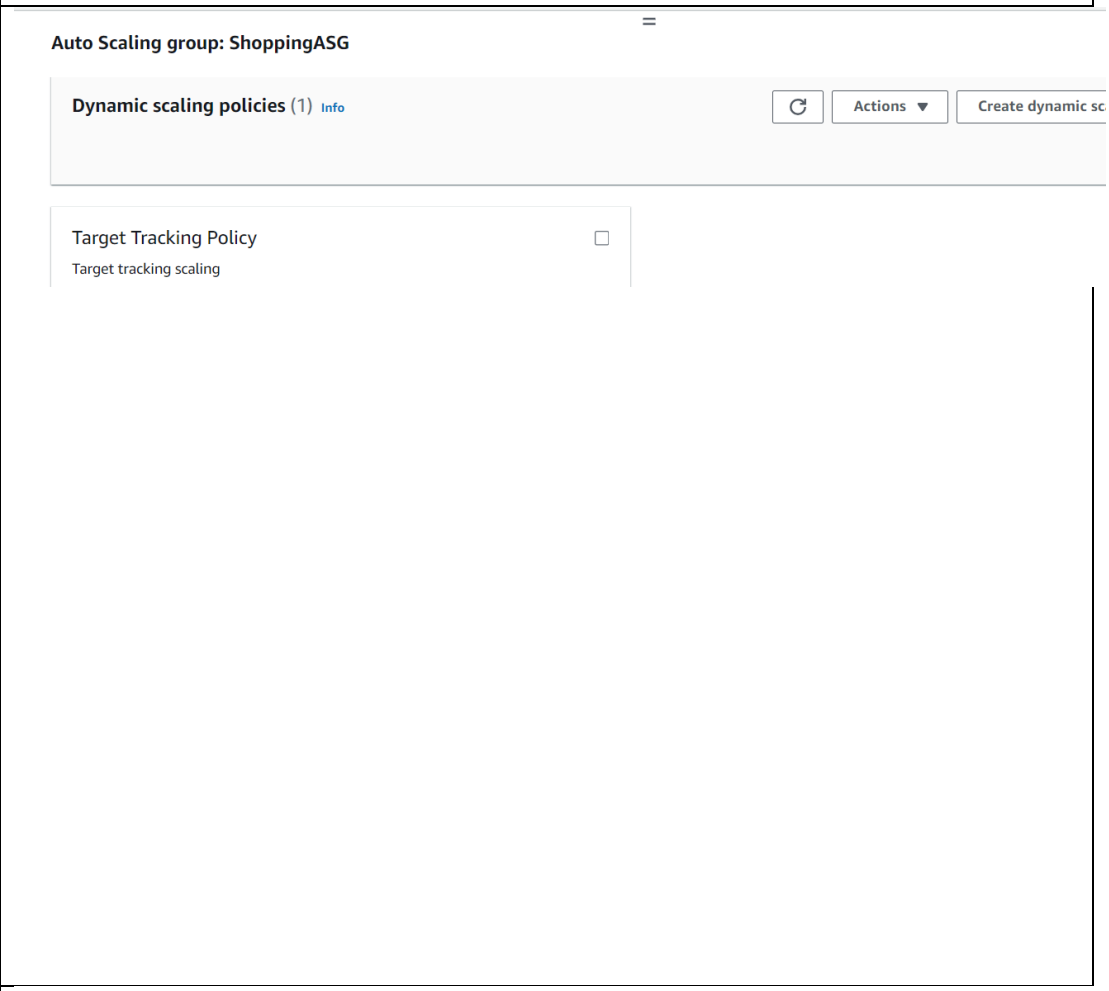
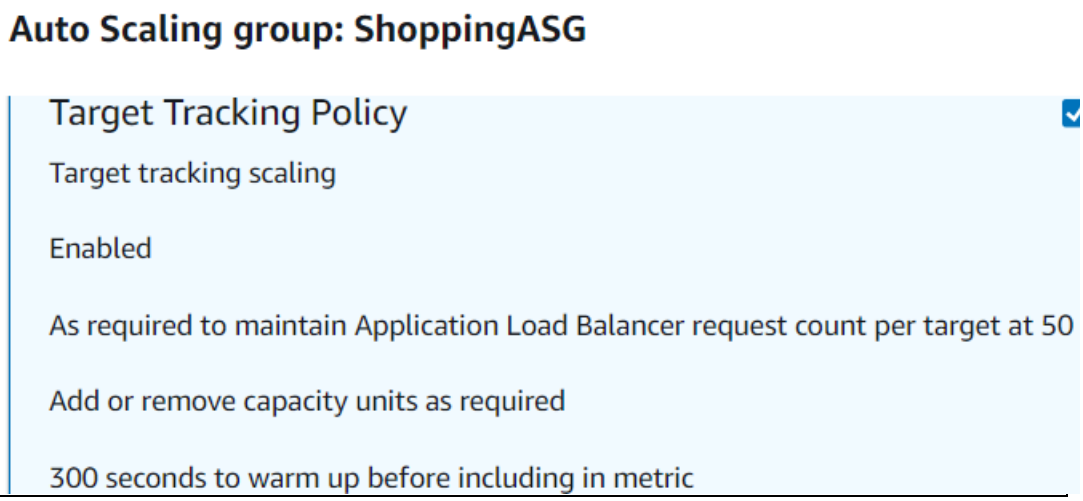
*Create the spot instances for load balancing in an online store can help reduce costs while ensuring high availability	
*Spot Instance VPC and Subnet configurations	
*Spot instance created and the following key pair attached with it	
*Inbound and outbound rules are typically associated with security groups and network access control lists (ACLs) in	

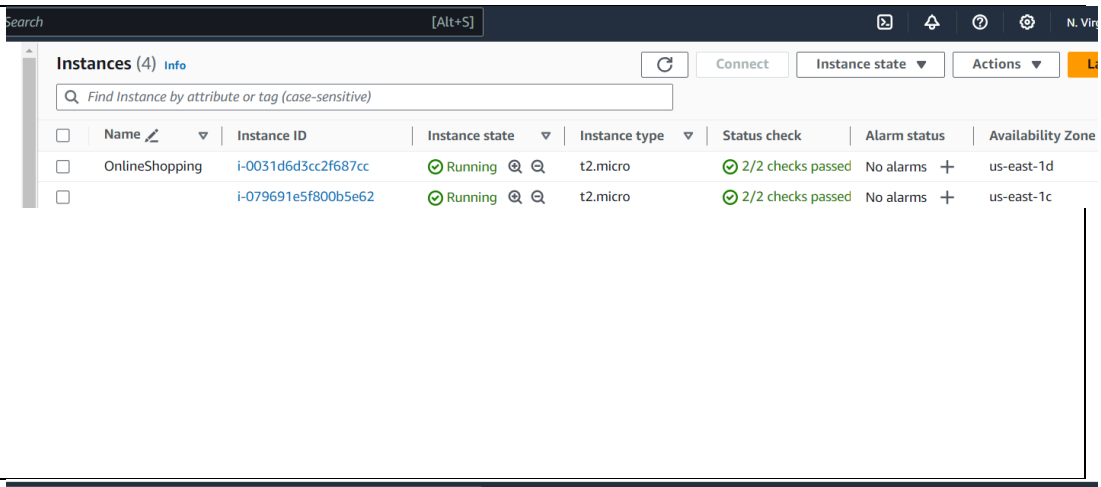
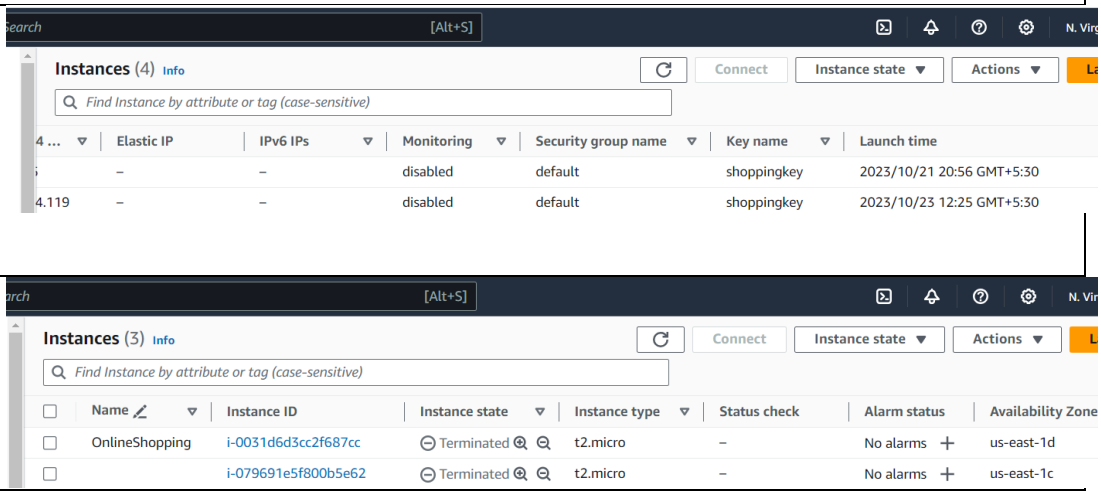
cloud computing environment	
*Spot instance inbound rules and outbound rules	
*Create an AWS Launch Template that specifies the configuration for your spot instances. *You can define the instance type, security groups, key pair, and other settings.	
*Create an Auto Scaling group on your cloud provider (e.g., AWS, Google Cloud) that utilizes spot instances. *This group will automatically adjust the number of	

instances based on traffic.	
*The practice of distributing subnets across Availability Zones is a best practice for achieving high availability, fault tolerance, and efficient load balancing in cloud-based applications for online stores using spot instances in AWS.	
*Deploy a load balancer, such as an Application Load Balancer (ALB) on AWS to distribute incoming traffic efficiently across your spot instances. *This ensures even load distribution and fault	

tolerance.	
*Load balancers perform health checks on the spot instances in the associated target group. * If a spot instance becomes unhealthy or is terminated, the load balancer will stop routing traffic to it. *This ensures that only healthy instances serve customer requests	**Load balancer: Shoppinglb**  **Listeners and rules** (1) Info Manage rules ▼ Manage listener ▼ A listener checks for connection requests on its configured protocol and port. Traffic received by the listener is routed according to the default action and any additional rules. Filter listeners Protocol:Port | Default action | Rules | ARN | Security policy HTTP:80 | Forward to target group • shoppingtarget: 1 (100%) • Group-level stickiness: Off | 1 rule | ARN | Not applicable
*Load balancer security groups	**Load balancer: Shoppinglb**  Details | Listeners and rules | Network mapping | Security | Monitoring | Integrations | Attributes | Tags **Security groups** (1) A security group is a set of firewall rules that control the traffic to your load balancer. Security Group ID | Name | Description sg-0891c98e0931de380 | default | default VPC security group
*The HTTP port is essential for efficiently routing web traffic and ensuring web	**Load balancer: Shoppinglb**  **Attributes** Traffic configuration TLS version and cipher headers: Off HTTP/2: On WAF fail open: Off Idle timeout: 60 seconds **Packet handling** Desync mitigation mode | Drop invalid header fields | X-Forwarded-For header | Client port preservation

application support	
*When traffic arrives at the load balancer, it distributes incoming requests across the spot instances in the target group	
*Configure your load balancer to work in conjunction with the Auto Scaling group. * This enables automatic addition or removal of spot instances as per the traffic demand	
*Set up health checks to monitor the status of your spot instances. *The load balancer can route traffic only	

to healthy instances.	
*Create auto scaling policies that define when and how instances should be added or removed based on metrics like CPU utilization or request count. *Dynamic scaling policies are crucial for managing the workload and costs in an online store that uses spot instances	
*Target tracking scaling policy adjust resource capacity based on actual demand and handle fluctuations in traffic.	

*The spot instance below online shopping launched based on auto scaling policies given to auto scaling group	
*The spot instance have been created with same configuration	
*The spot instances has been terminated	

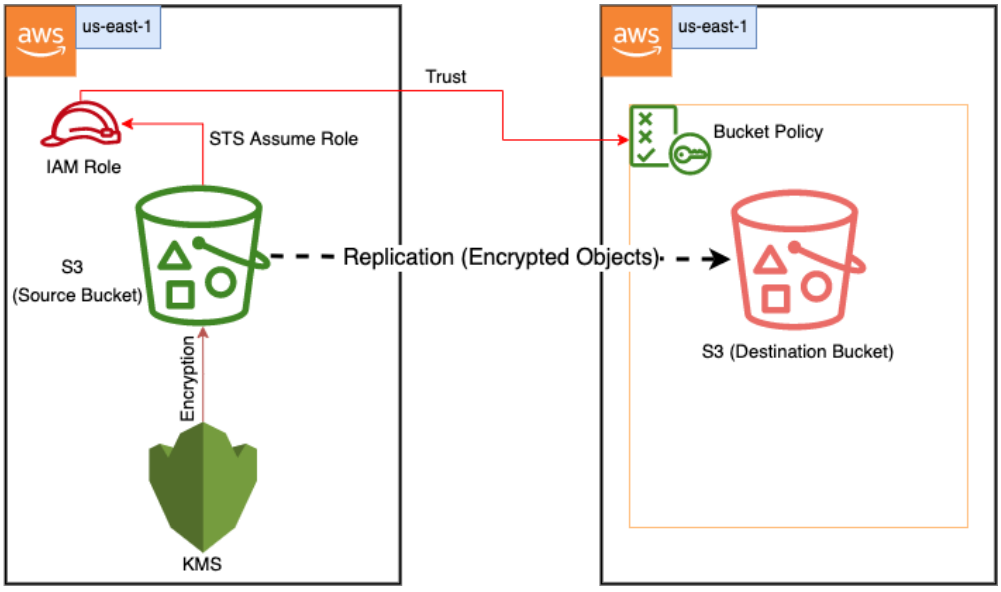
- Regularly monitor your system's performance and traffic patterns. Adjust your auto scaling and load balancing configurations as needed to optimize resource utilization and customer experience.
- By following these steps, you can ensure that your online store is highly available and responsive while benefiting from the cost advantages of spot instances.

2. Static website hosting in S3 bucket

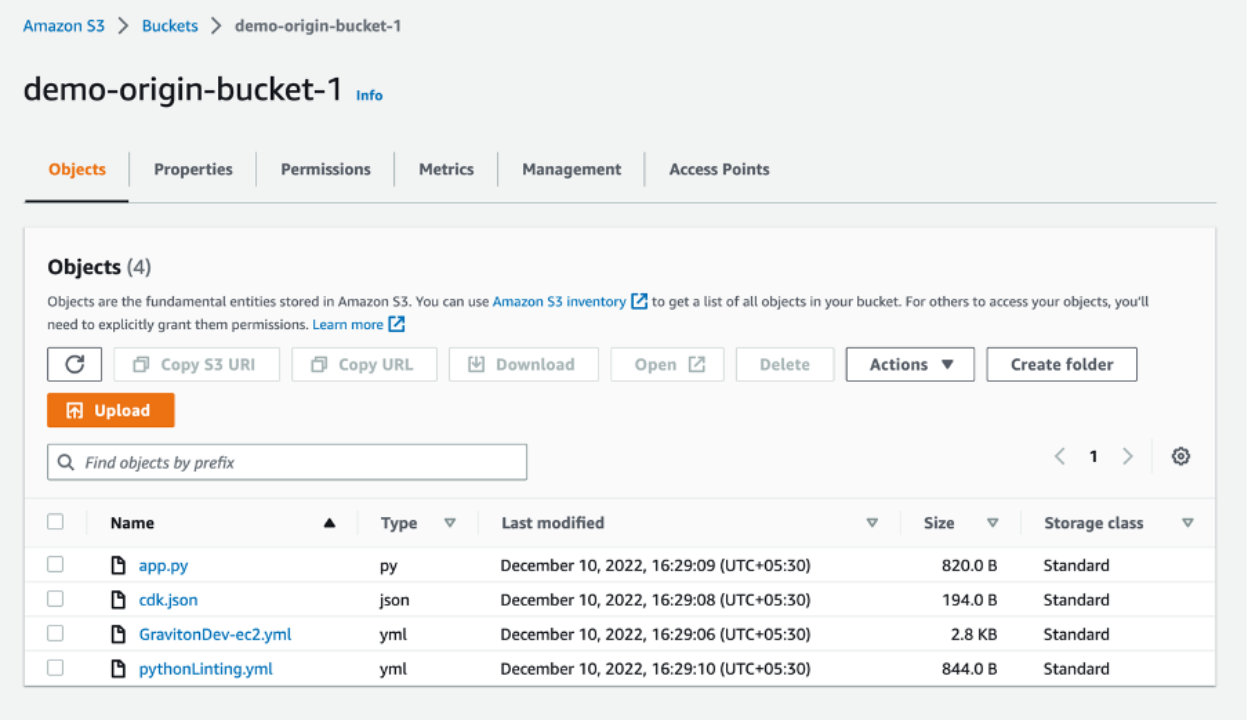
Agenda:

- S3
- Objects
- Static Website Hosting
- ACL(Access Control Lists)

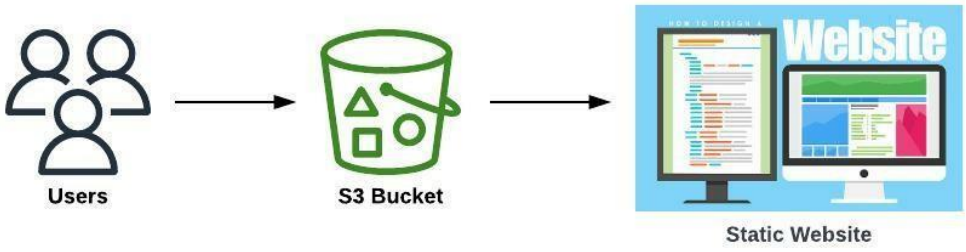
- S3: Amazon S3 (Simple Storage Service) is a scalable object storage service provided by Amazon Web Services (AWS). An S3 bucket is a container for storing objects which can consist of data files, images, videos, and any other type of binary or text data. S3 buckets are fundamental building blocks of Amazon S3, and they play a crucial role in cloud storage and web hosting on AWS.The data could be sent securely from one bucket in a region to another bucket in another region using users authentication and encryption.



- Objects: are the fundamental entity that you store in a bucket. An object consists of data, a key (unique within a bucket), and metadata. When you upload a file to an S3 bucket, you are creating an object. Objects can be anything from a simple text file to complex data, images, videos, or other binary data.

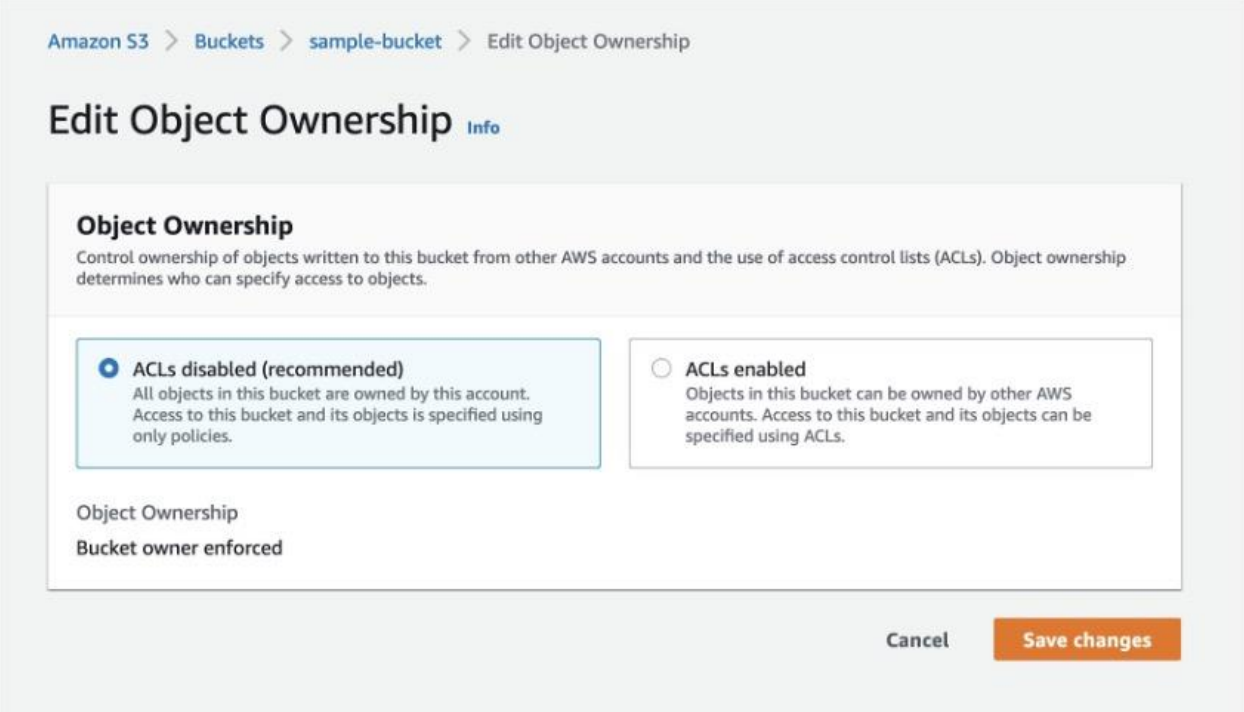


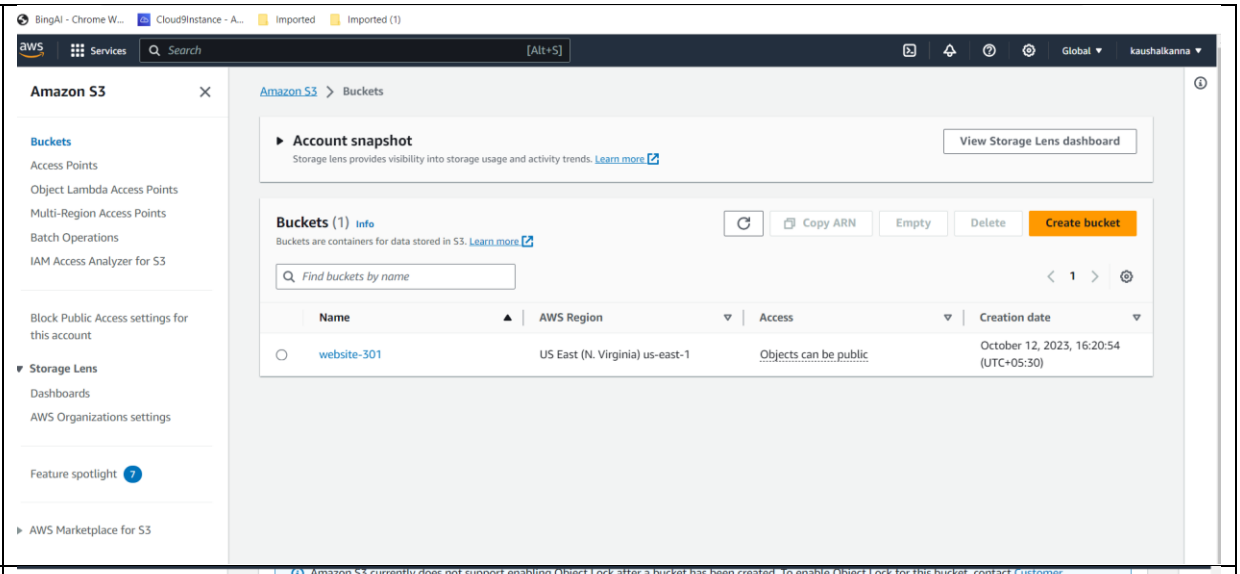
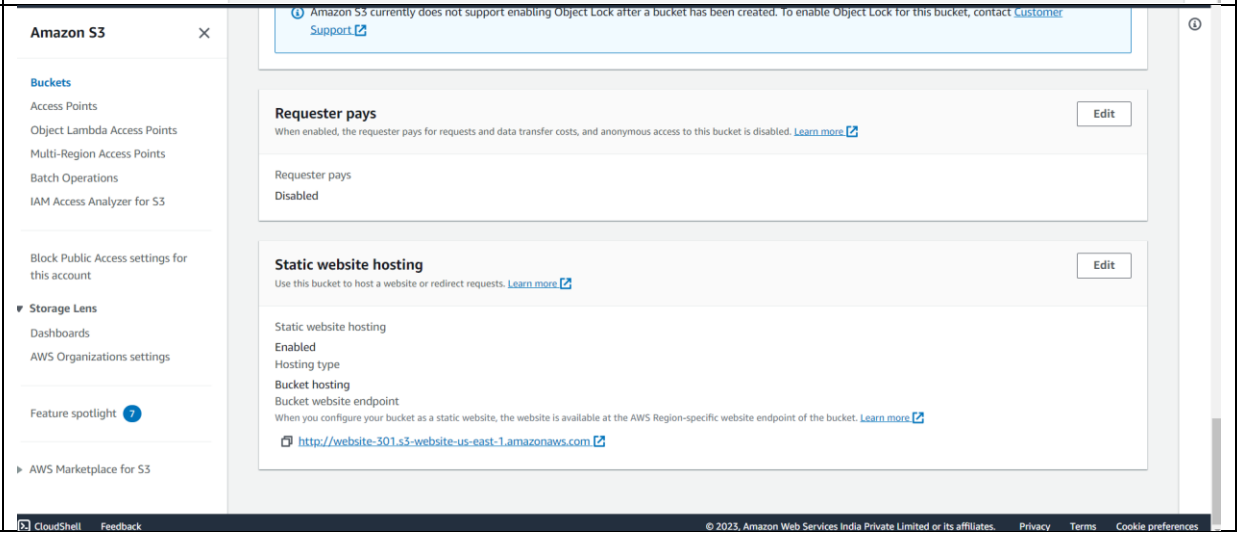
- Static Website Hosting: refers to the ability to host and serve static web content directly from Amazon S3, a scalable and durable object storage service. A static website consists of HTML, CSS, JavaScript, images, and other static assets that do not require server-side processing or a database. Hosting a static website in S3 is a cost-effective and straightforward way to make web content available to users.

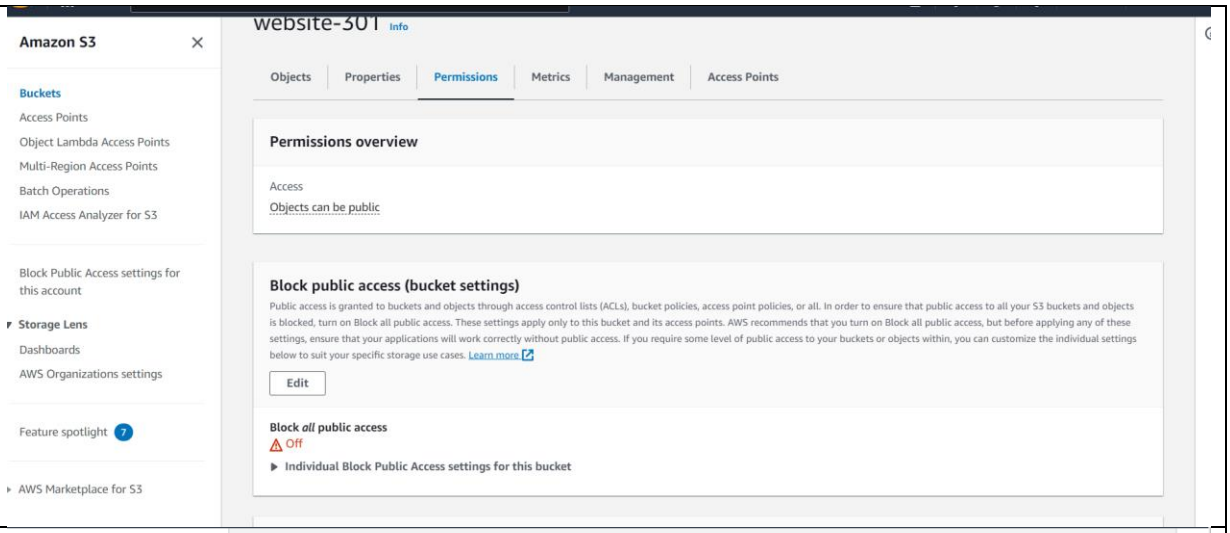
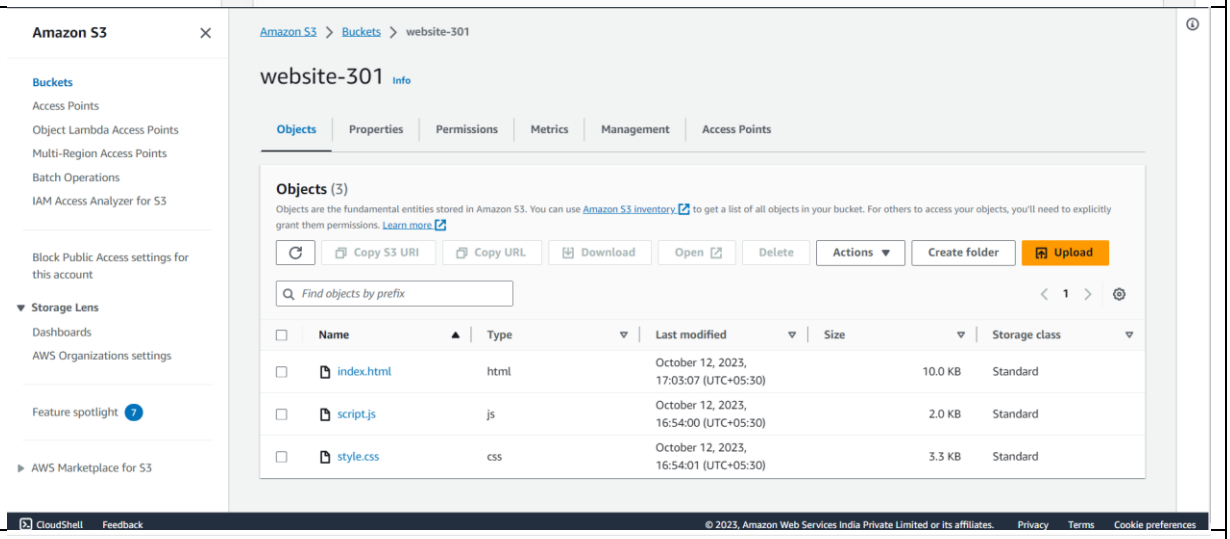


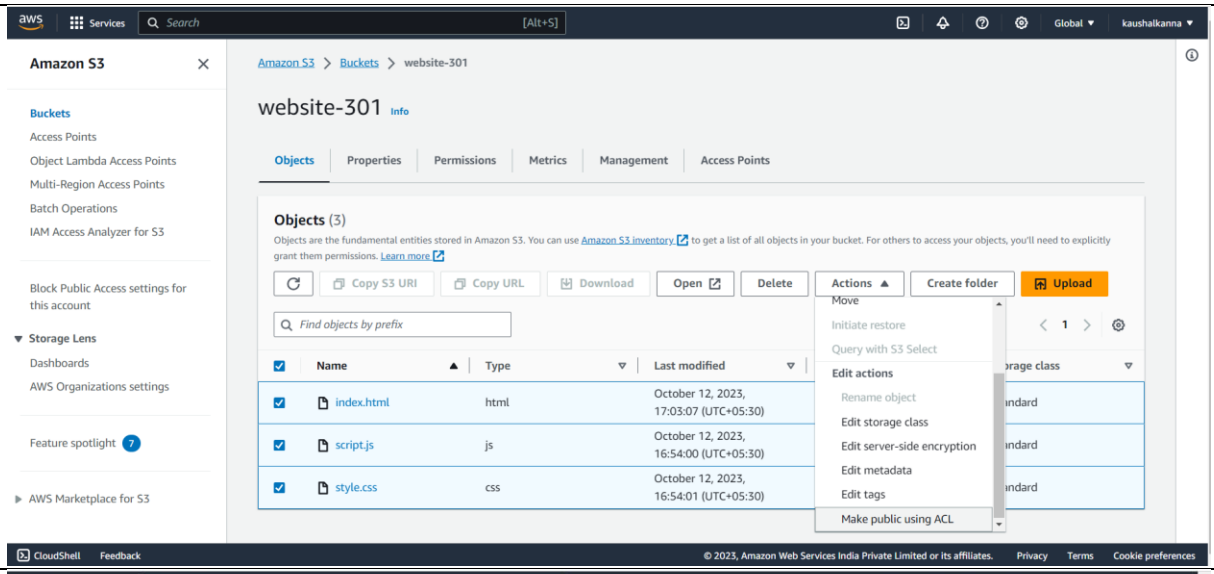
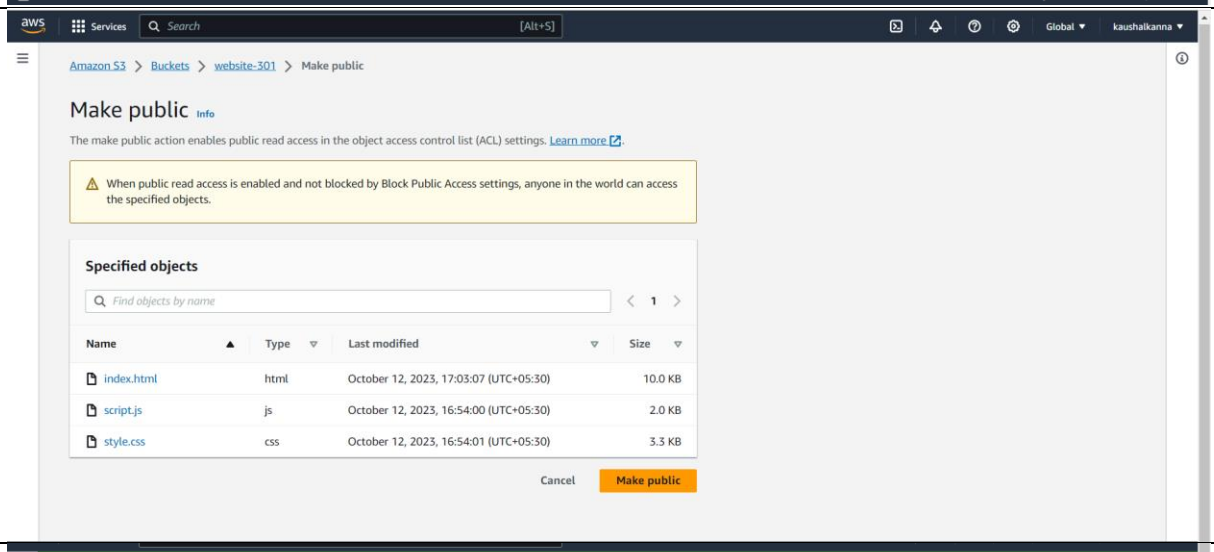
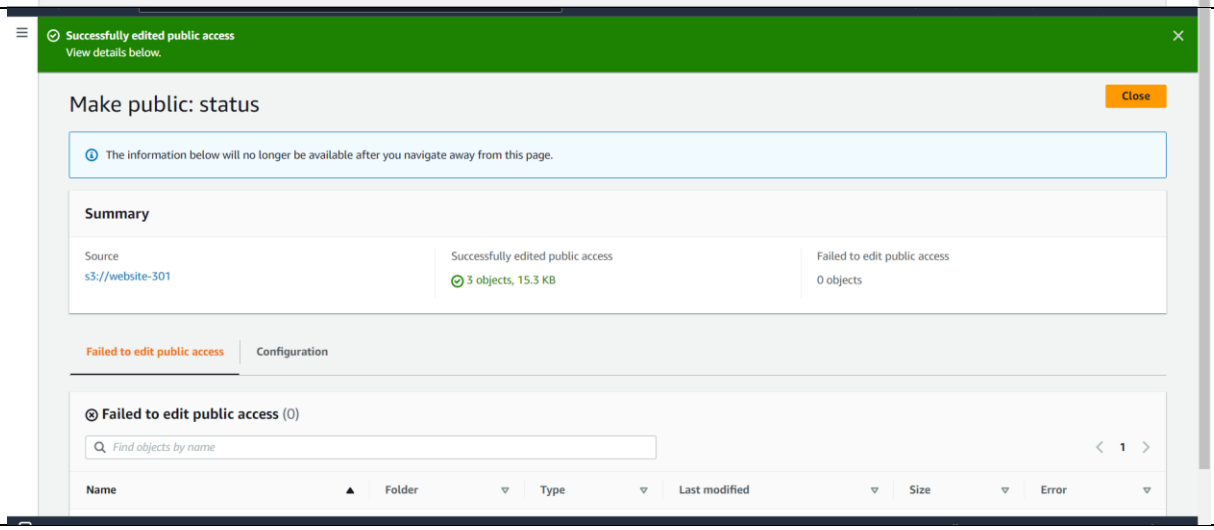
Static Website Hosting with S3

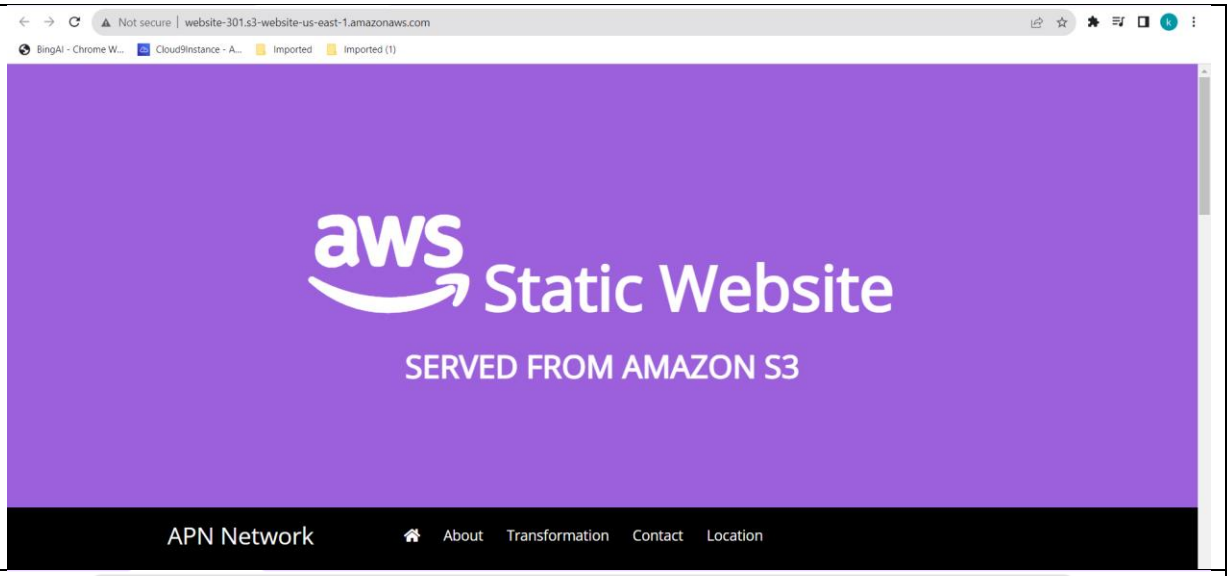
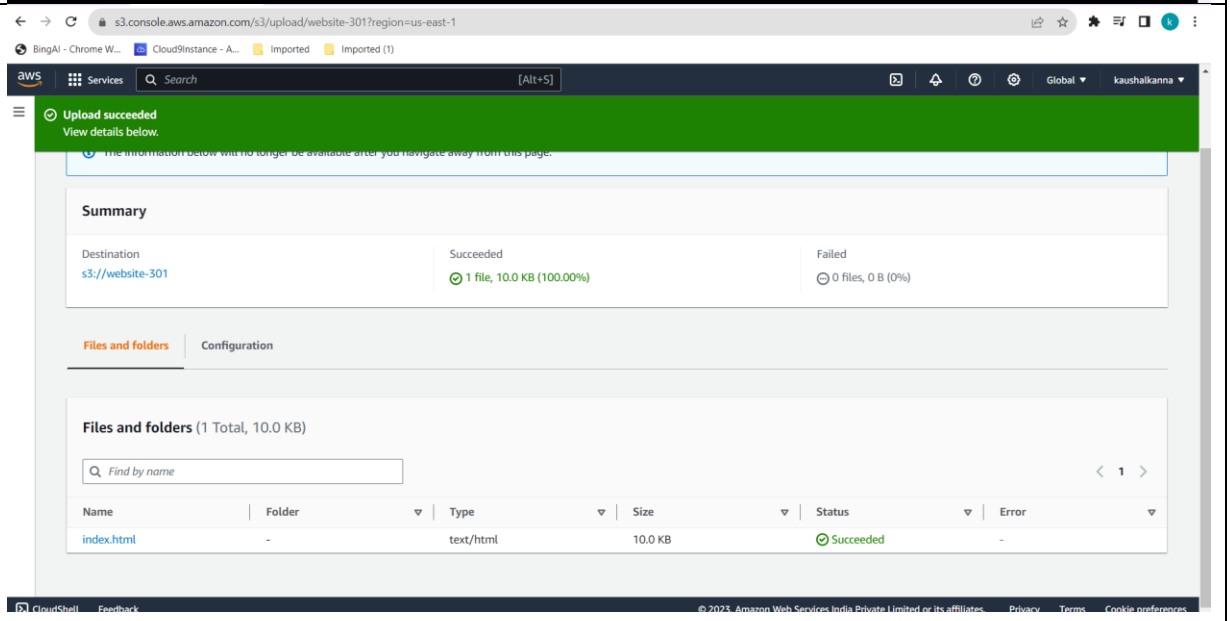
- ACL: ACL stands for Access Control List. An S3 ACL is a set of permissions that define who can access objects and buckets in S3 and what actions they can perform. ACLs are used to manage access to S3 resources at a finer granularity than what is provided by bucket policies alone.

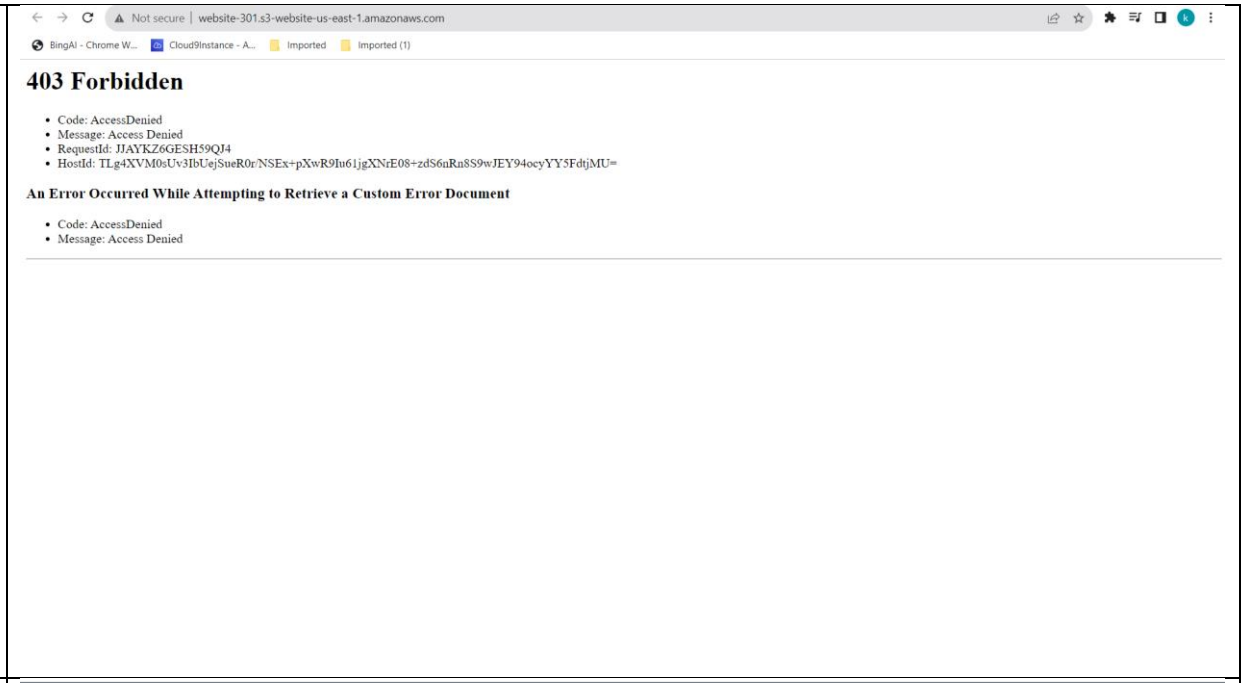
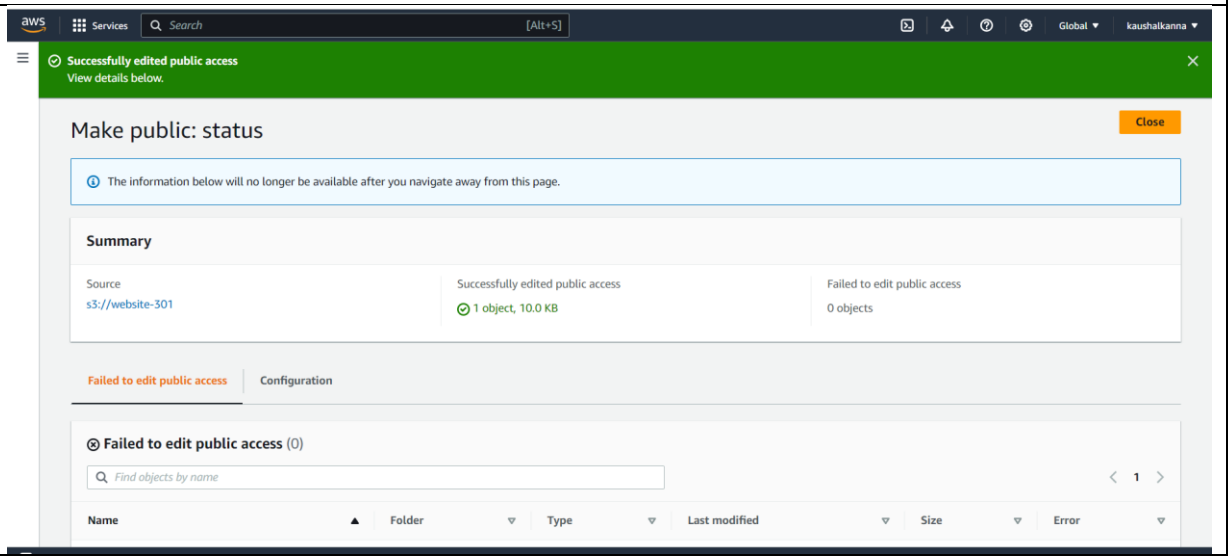


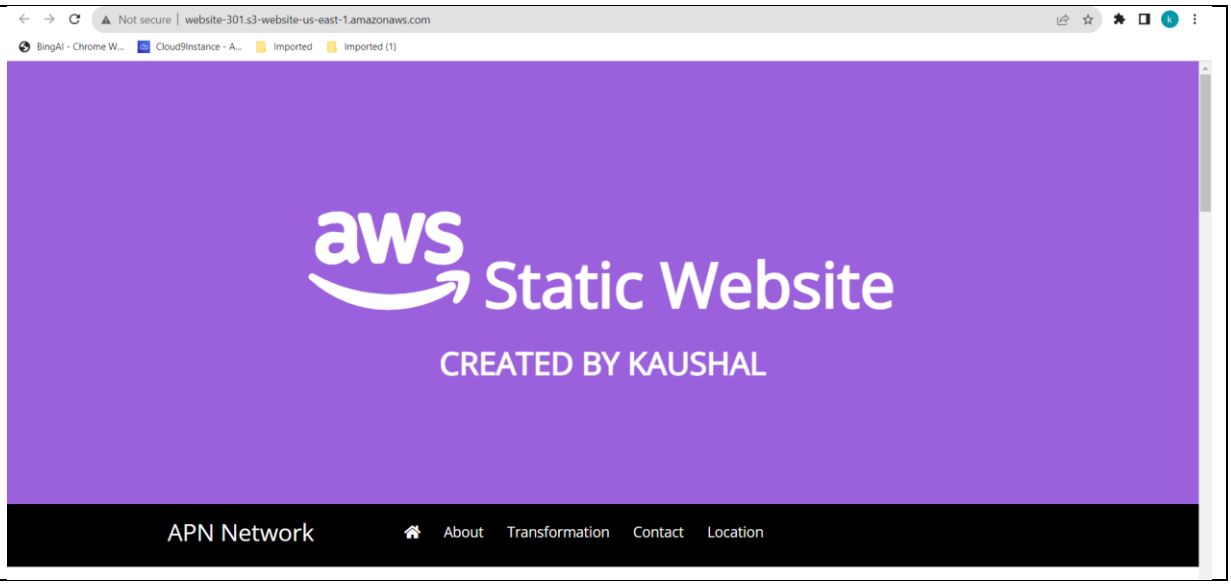
*Manually Create a S3 bucket in aws cloud	
*Enable static website hosting in S3 bucket which would assign url for website	

*Block public access should be disabled so that all users can access the website publicly	
*The html,javascript and css files are uploaded to S3 bucket as objects for hosting a website in public	
*The website shows an error because I didn't give the objects in public using ACL.	**403 Forbidden** • Code: AccessDenied • Message: Access Denied • RequestId: GP78FAVPHHNM0S99 • HostId: uwLoCR0RiydraviaTxfgJz1Imoo3V6m7JNS3D03tvUv4p8o5KuoBCpWYZP2txvAKBKN7TFqlqUM= **An Error Occurred While Attempting to Retrieve a Custom Error Document** • Code: AccessDenied • Message: Access Denied

*Through actions I have to give all objects in public using ACL	
*All the objects have been selected to make it public	
*Successfully edited the public access for users to access the website publicly	

*The static website created from S3 bucket objects in aws cloud	
*Now I am changing website content by modifying the code in index.html file. *The content served from amazon s3 will be replaced by created by kaushal. *After modifying code in index.html then I am replacing with old index.html and I am uploading file to s3 bucket and again we have to make public using acl to index.html object to access the static website.	

*The website shows error because the object in S3 bucket is not given public using ACL after replacing the file	
*The object has been successfully given public access using ACL	

*The static website content has been changed	

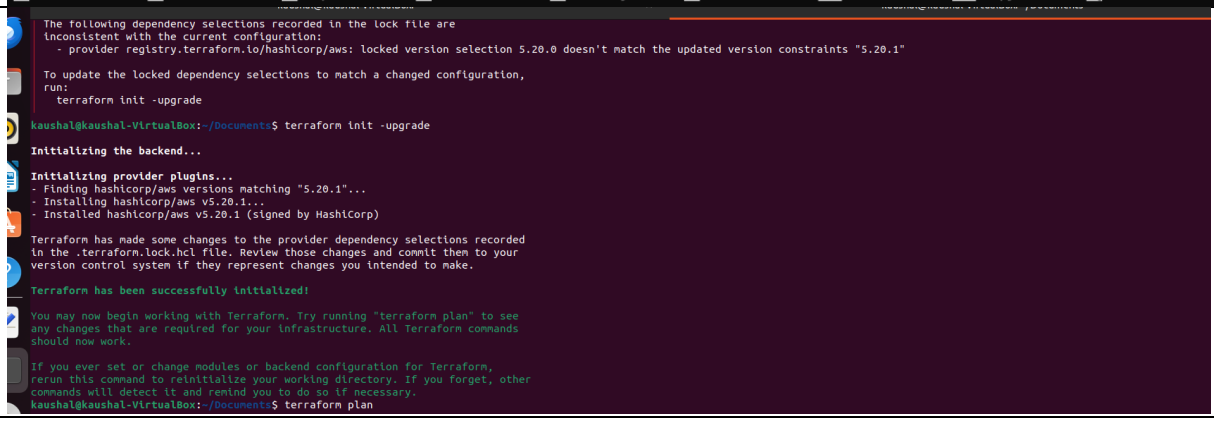
3.Terraform Scripting:

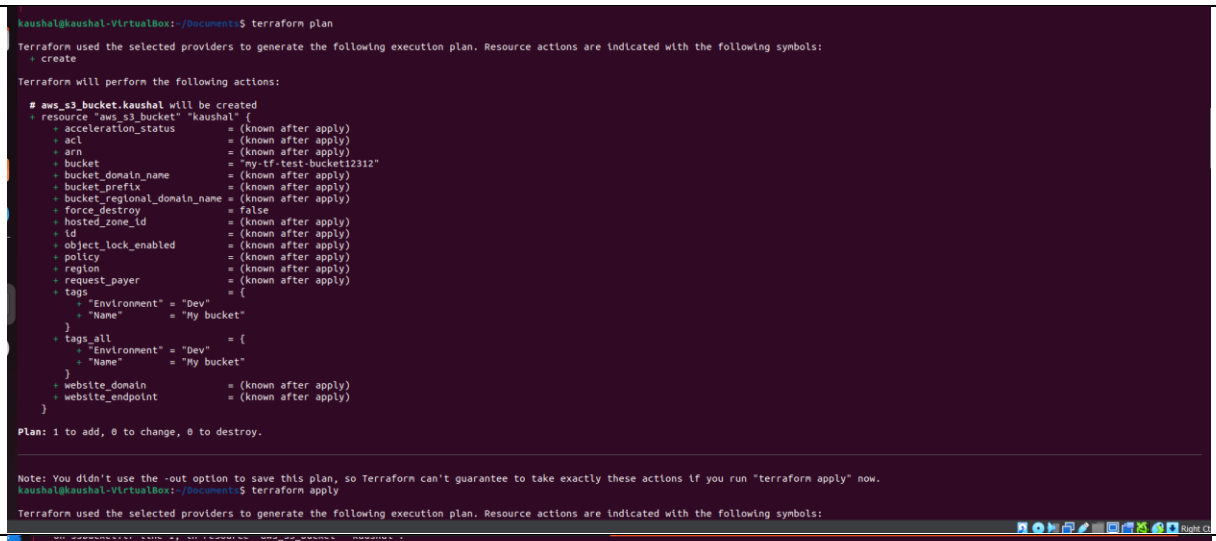
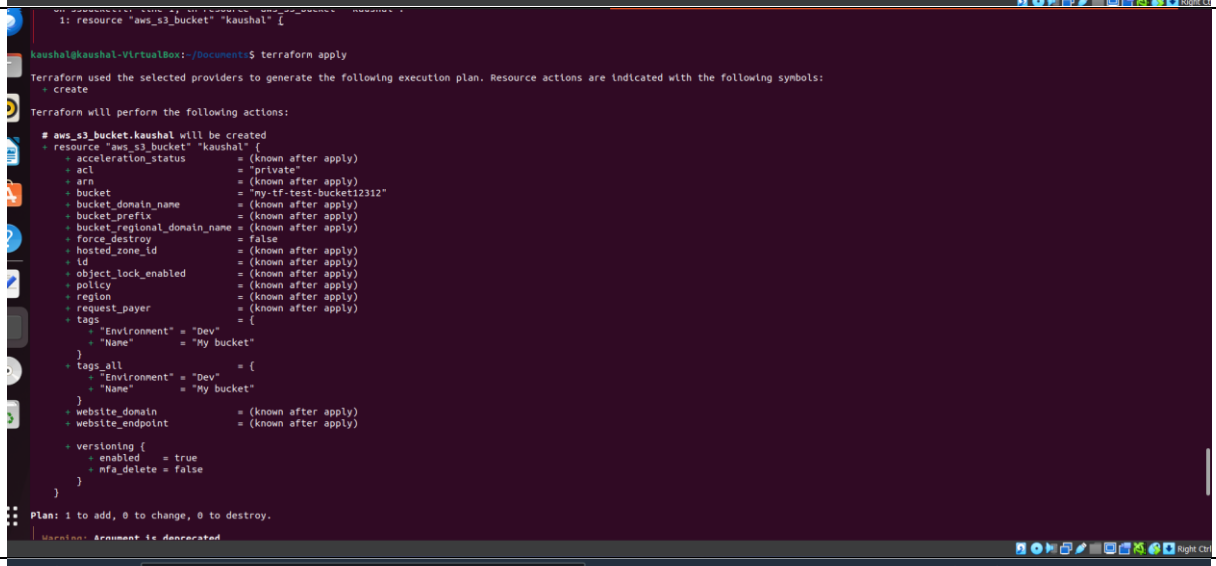
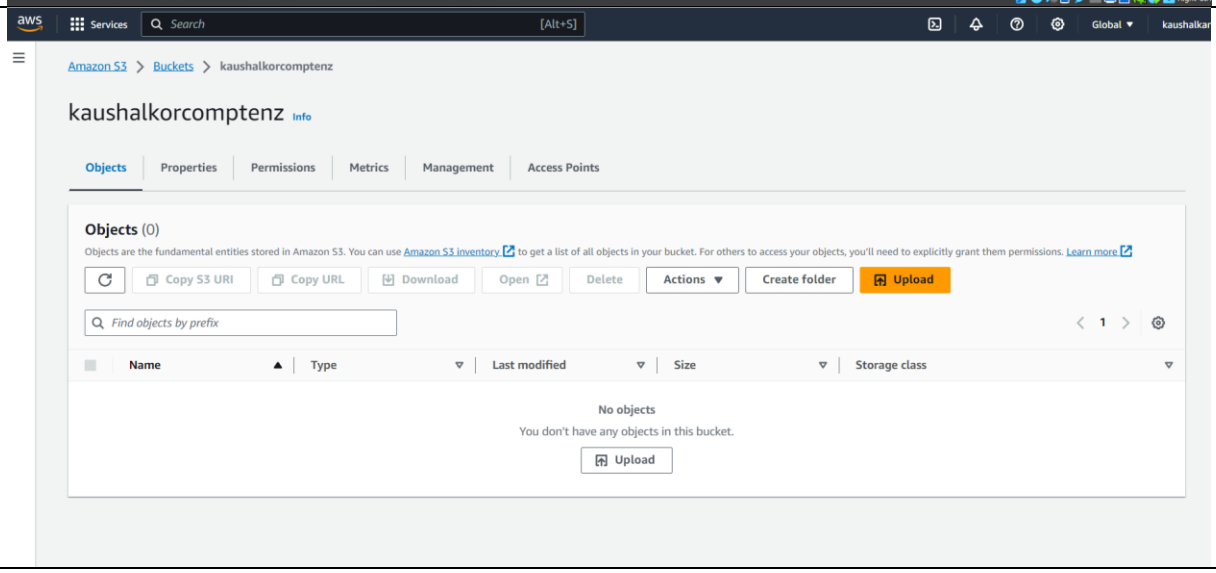
Agenda:

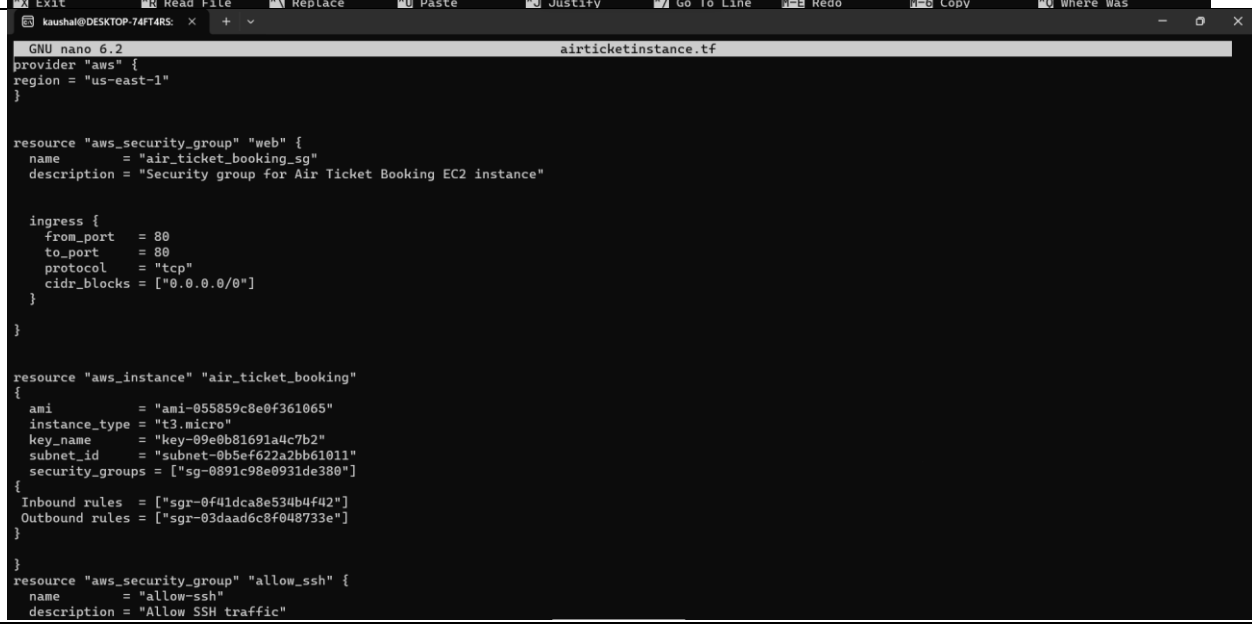
- Terraform
- Aws cloud resources

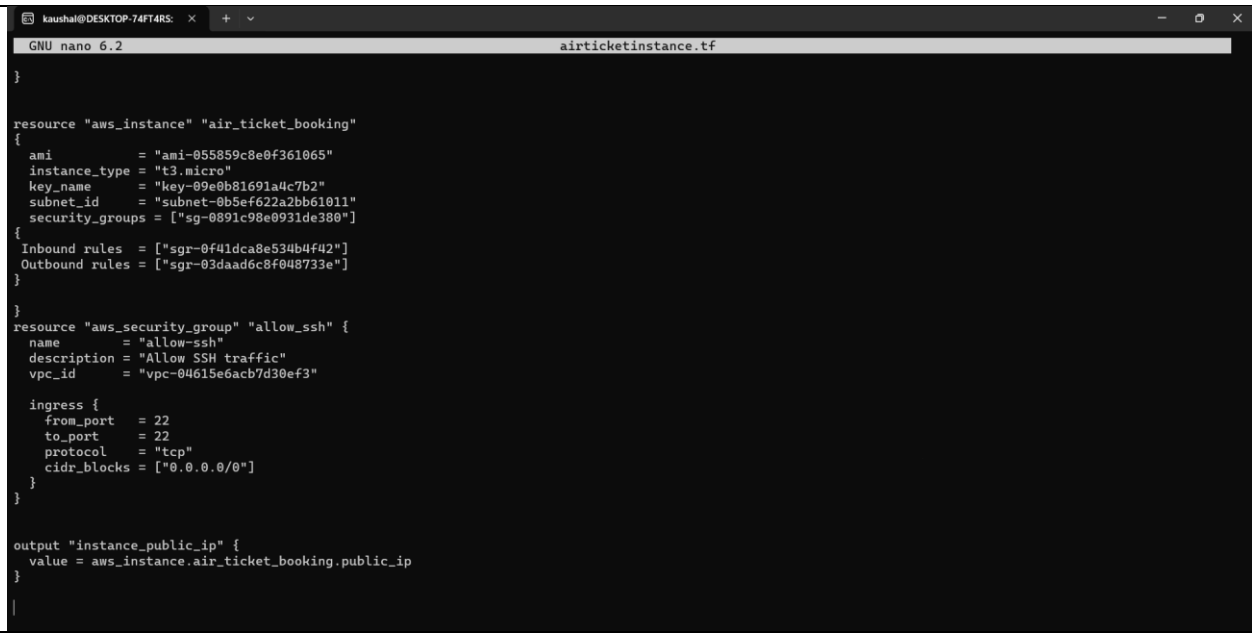
- Terraform: is an open-source Infrastructure as Code (IaC) tool developed by HashiCorp. It enables users to define and provision infrastructure resources in a declarative configuration language.With Terraform we can describe the entire infrastructure stack—including compute instances, storage, networking, and then use that code to create and manage the infrastructure on various cloud providers or on-premises environments.

- Cloud resources: refer to the computing components and services provided by cloud computing platforms.These resources are made available over the internet and can be accessed and managed remotely. Cloud providers offer a variety of resources that organizations and individuals can leverage to build, deploy, and manage applications and infrastructure

*Terraform script to create S3 bucket in cloud	
*First the Terraform has to be initialized	

*We have to to give plan to read the configuraton and the state of resources that we are going to create in cloud	
*Finally we have to give apply to create resources in cloud	
*The following S3 bucket would be created in cloud	

*Terraform script to create ec2 instance	
*Terraform script to create instance in airticketbooking	

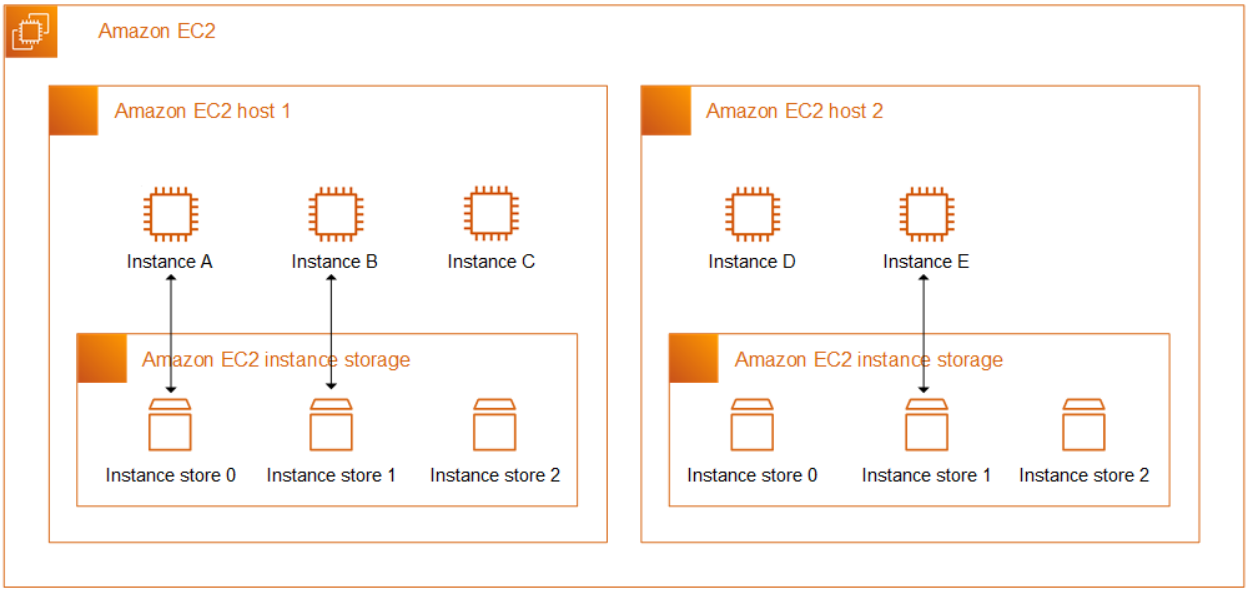
* Terraform script to create instance in airticketbooking	

4.Launch templates on ec2 instance with role associated

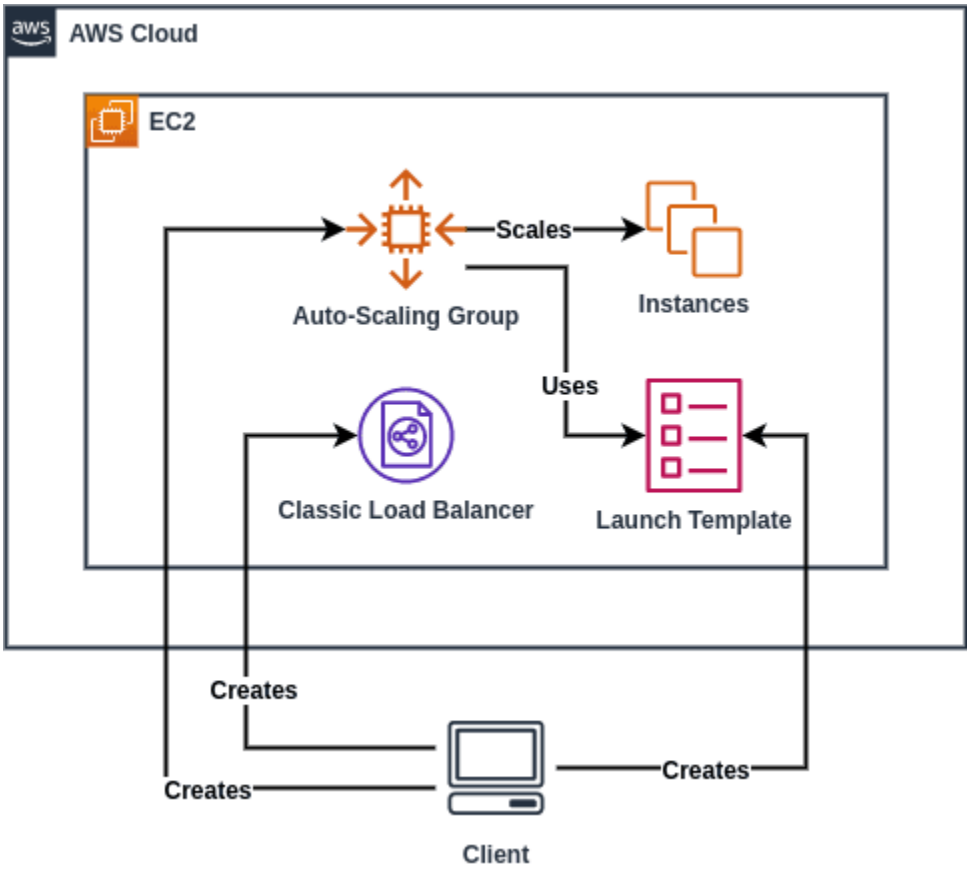
Agenda:

- EC2 Instance
- Launch Templates
- IAM Role

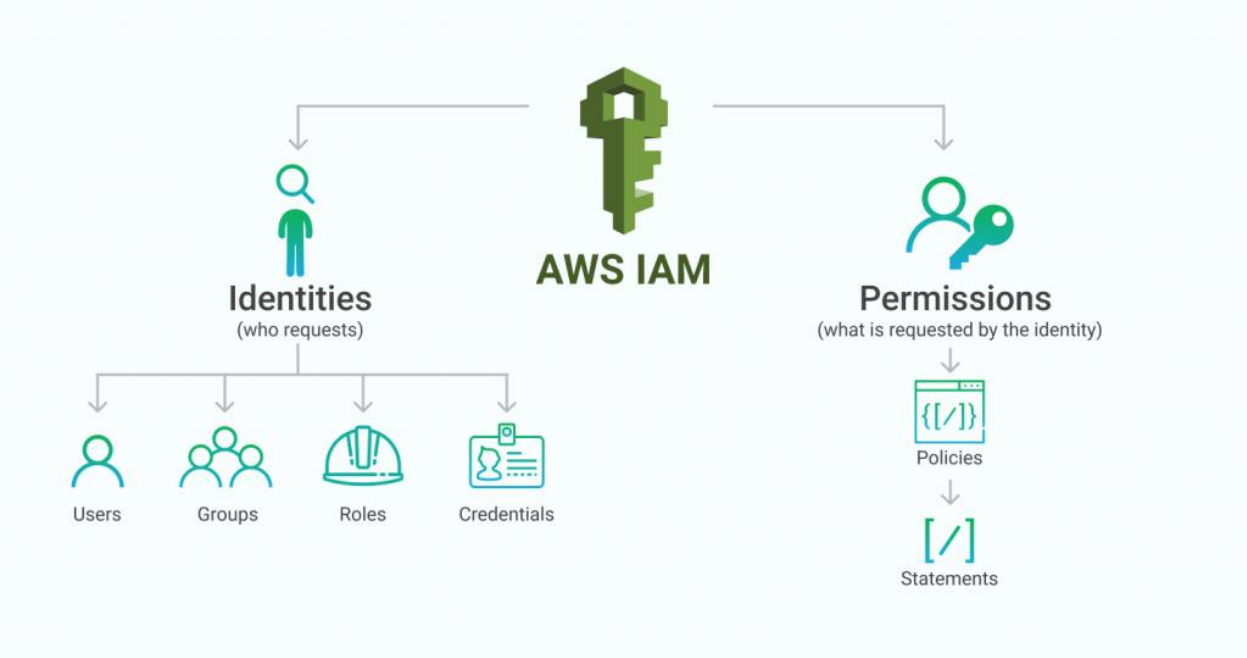
- EC2 Instance: a web service provided by Amazon Web Services (AWS) that allows users to rent virtual servers known as instances in the cloud. EC2 instances provide scalable computing capacity and are designed to run various types of applications from simple to complex workloads. Users can choose from a variety of instance types where each instance type optimized for different use cases.

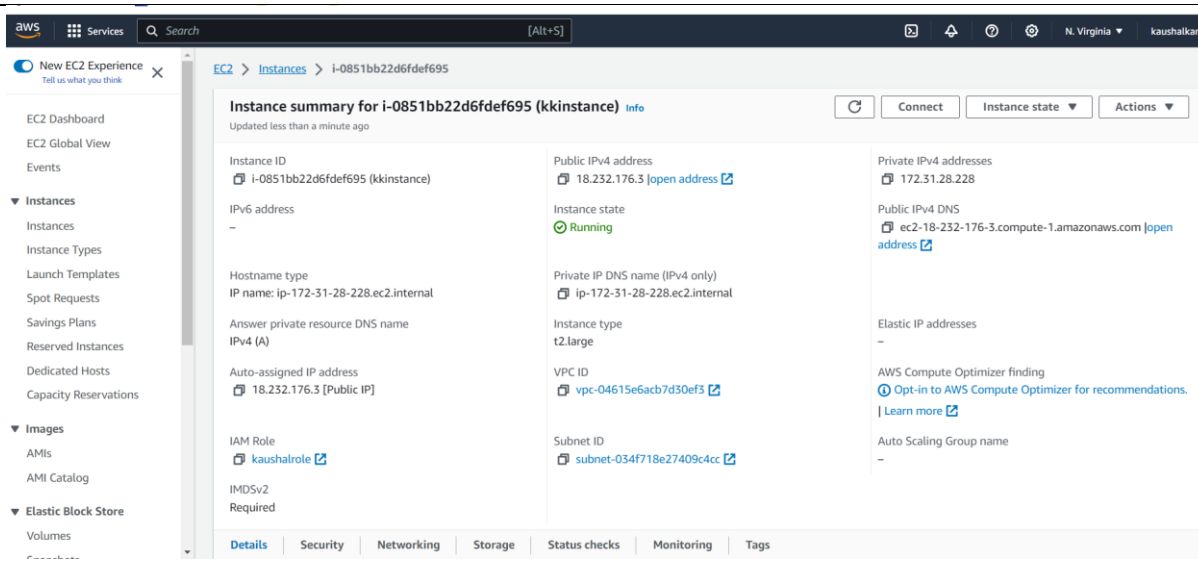
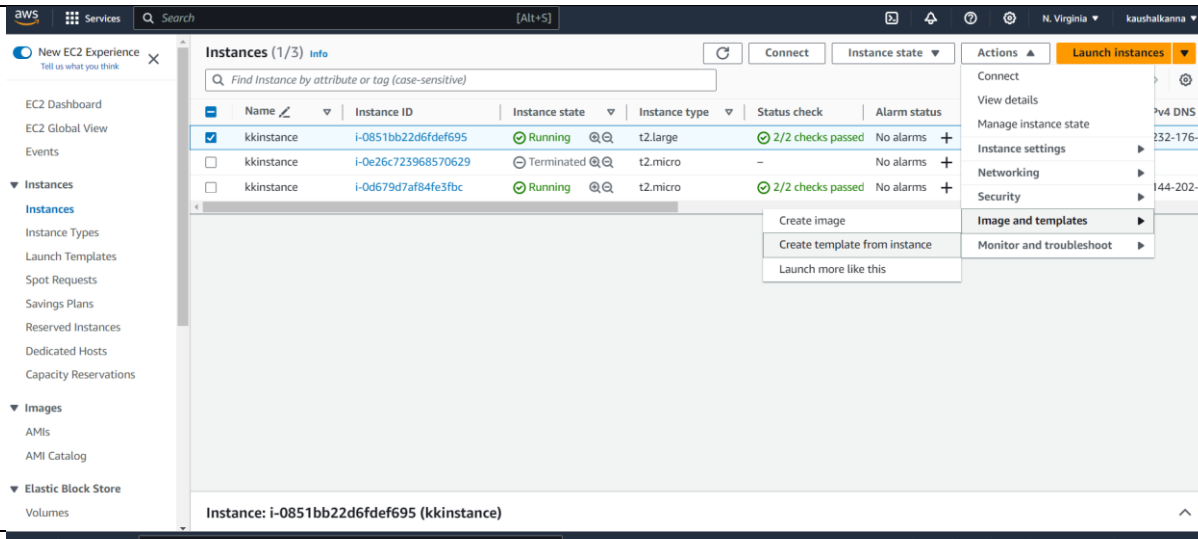
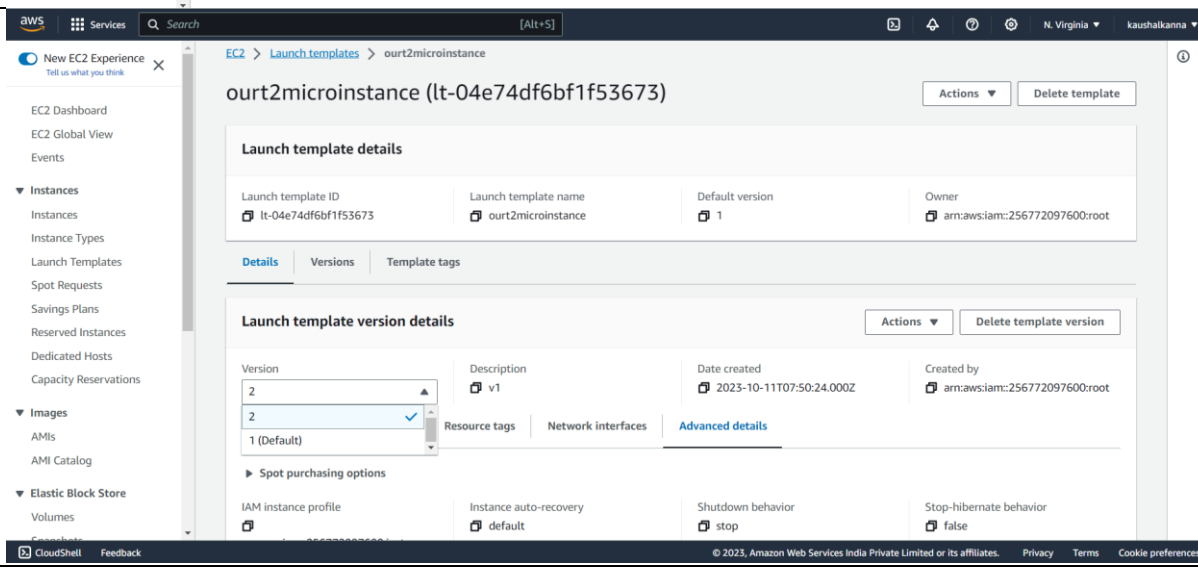


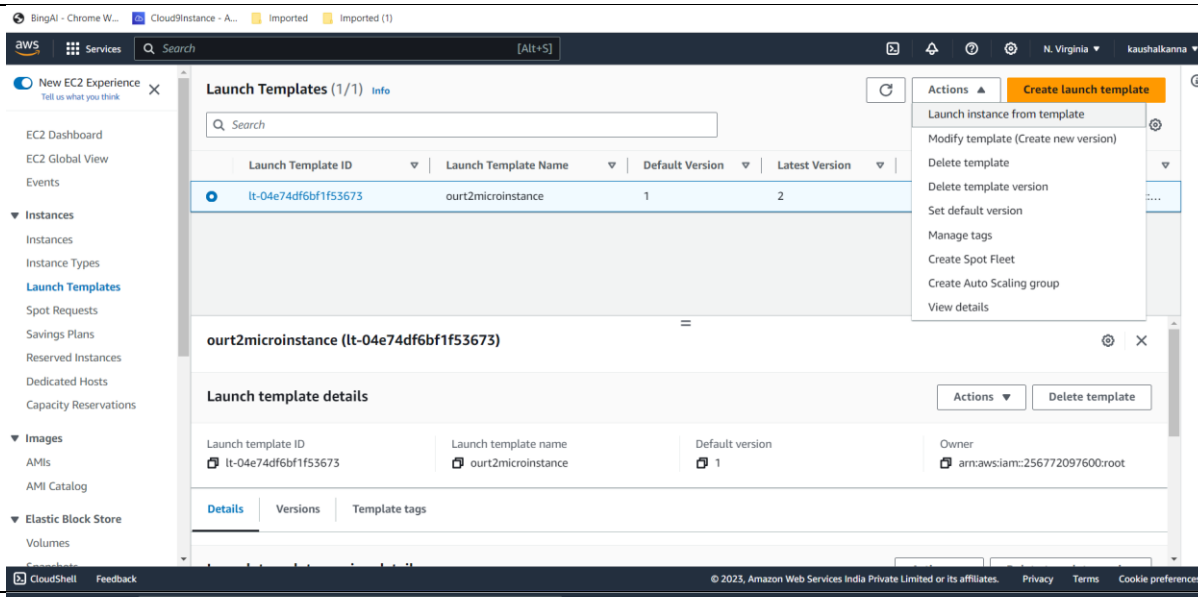
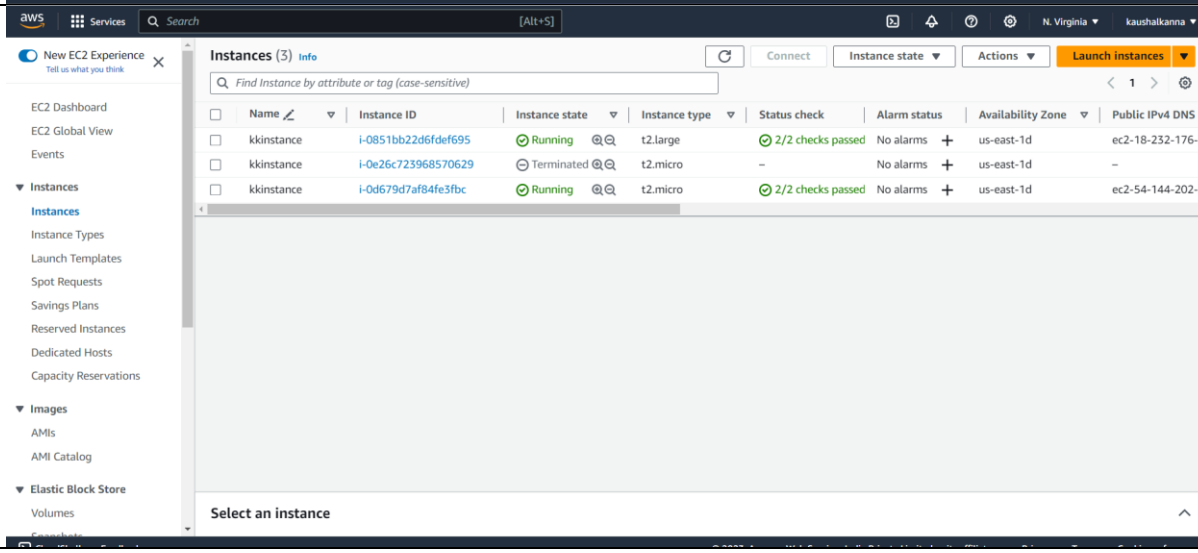
- Launch Templates: are a feature that allows users to define a set of parameters and configurations for launching EC2 instances. Launch Templates streamline the process of launching instances by providing a template that includes various specifications, such as the Amazon Machine Image (AMI), instance type, key pair, security groups, block device mappings, and other settings.



- IAM Role: are a fundamental component of the AWS security model. IAM roles define a set of permissions that determine what actions users, applications, or AWS services can perform within your AWS environment. Unlike IAM users or groups, IAM roles are not associated with a specific identity. Instead roles can be assumed by IAM users, AWS services, or federated identities.



*Create a instance with instance type,vpc,subnets and iam role to be associated with it	
*Once the instance have been created then create template from the instance	
*Specify the version of template that we have to launch from the instance	

*The template has been launched with its latest version. *We can also launch instance from template	
*These are the instances launched from the template	

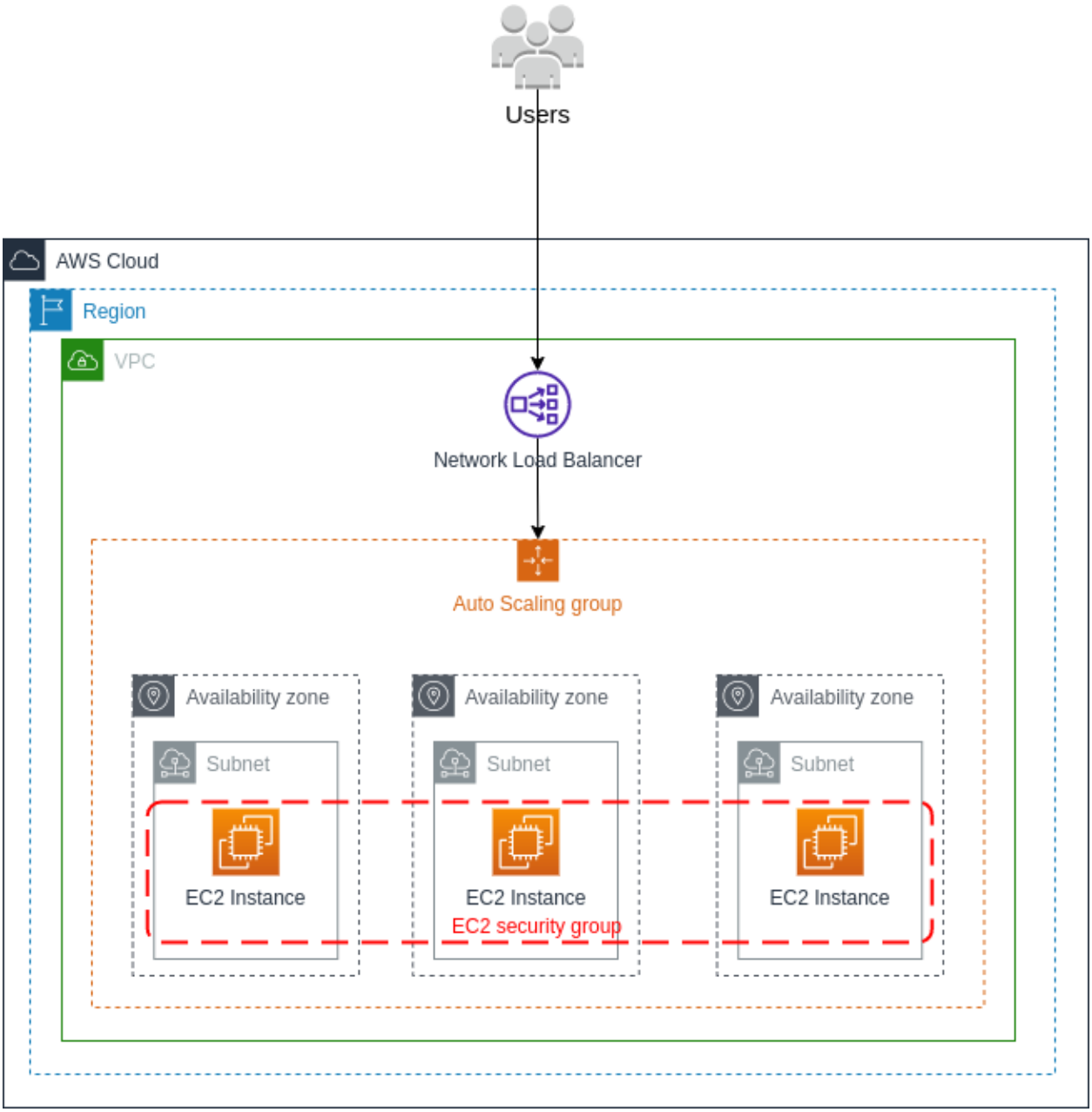
5.Air ticket booking system project in aws cloud

When building an air ticket booking system in the AWS Cloud.We would typically use a combination of AWS services rather than traditional "instances." Here are key components and AWS services to consider for such a project:

Agenda:

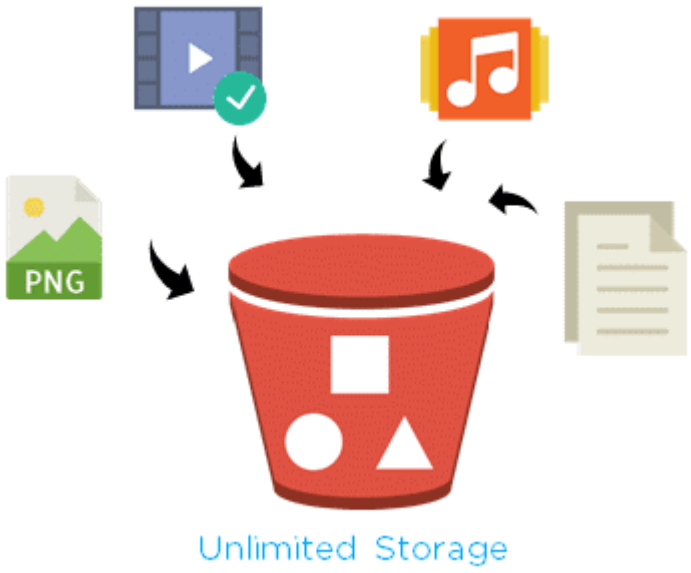
- EC2 Instances
- Relational Database Service(RDS)
- S3
- Route 53
- Elastic Beanstalk
- CloudFront
- Lambda
- Cognito
- Simple Email Service(SES)
- Step Functions
- CloudWatch
- Identity and Access Management (IAM)
- API Gateway
- DynamoDB
- Simple Notification Service(SNS)
- Simple Queue Service(SQS)
- Redshift
- Polly

- EC2 Instances: can play a crucial role in providing the necessary computing resources to support the various components of the system.It can host web servers to serve the booking system's front-end application for users to get access to it.

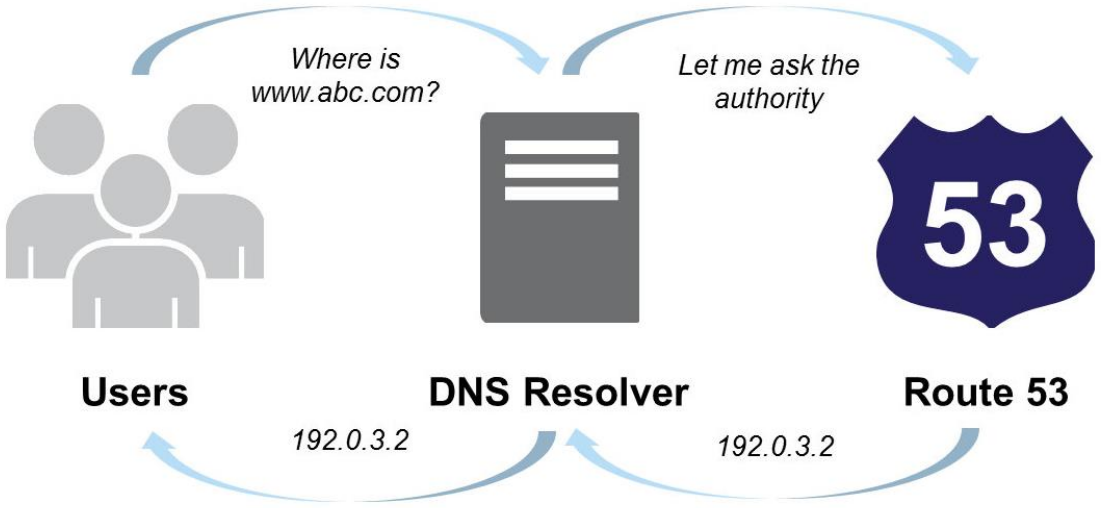


- Relational Database Service(RDS): is a managed database service provided by Amazon Web Services (AWS). It simplifies the process of setting up, operating, and scaling a relational database in the cloud. RDS supports several popular database engines, making it easier for users to deploy and manage databases without the need to handle the administrative tasks associated with database management.

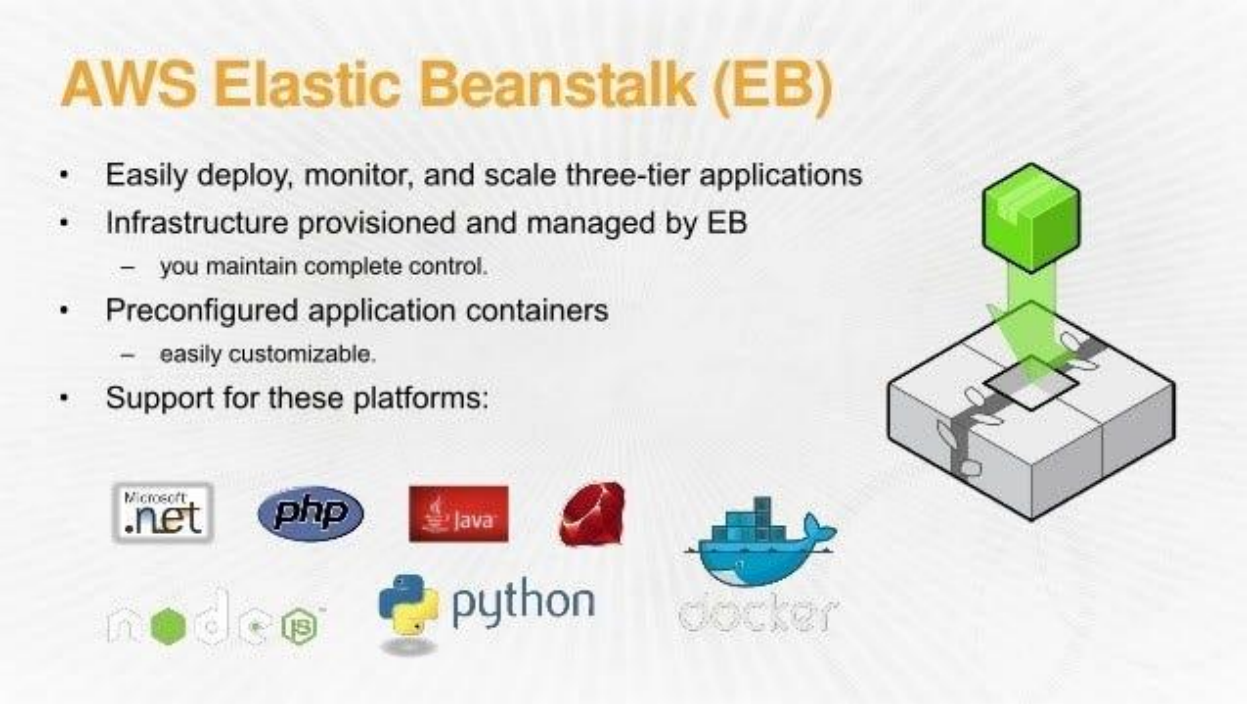
- S3: is designed to store and retrieve any amount of data from anywhere on the web and is commonly used for a variety of purposes including data storage, backup and recovery, data archiving, content distribution and as a foundation for various AWS services.



- Route 53: is a scalable and highly available Domain Name System (DNS) web service provided by Amazon Web Services (AWS). DNS is a fundamental component of the internet that translates user-friendly domain names (like www.example.com) into IP addresses that computers use to identify each other on the network. Route 53 not only offers domain registration services but also provides DNS and health checking services.

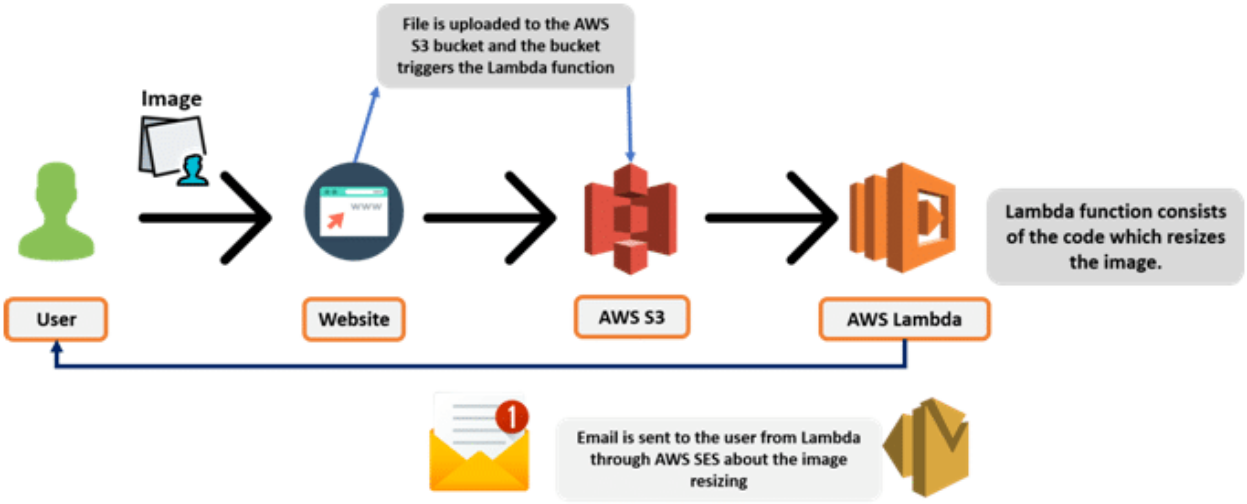


- Elastic Beanstalk: is a fully managed service provided by Amazon Web Services (AWS) that makes it easy to deploy and run applications in multiple programming languages, including Java, .NET, Python, Node.js, Ruby, Go, and more. Elastic Beanstalk abstracts the underlying infrastructure and automates the deployment, scaling, and management of applications, allowing developers to focus on writing code rather than managing the underlying infrastructure.

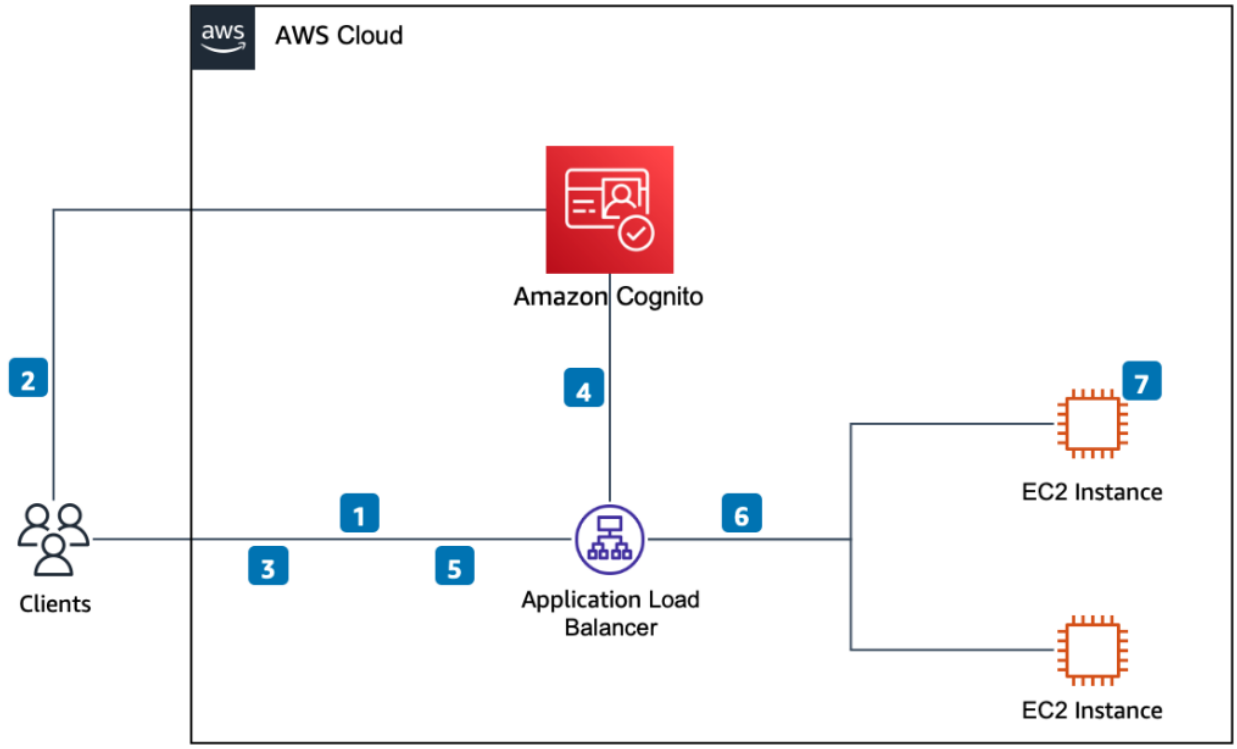


- Cloudfront: is a content delivery network (CDN) service provided by Amazon Web Services (AWS). CloudFront helps deliver static and dynamic web content, including images, videos, scripts and other files to users with low latency and high transfer speeds. It is designed to enhance the performance, reliability and scalability of web applications and websites by caching content at edge locations distributed around the world.

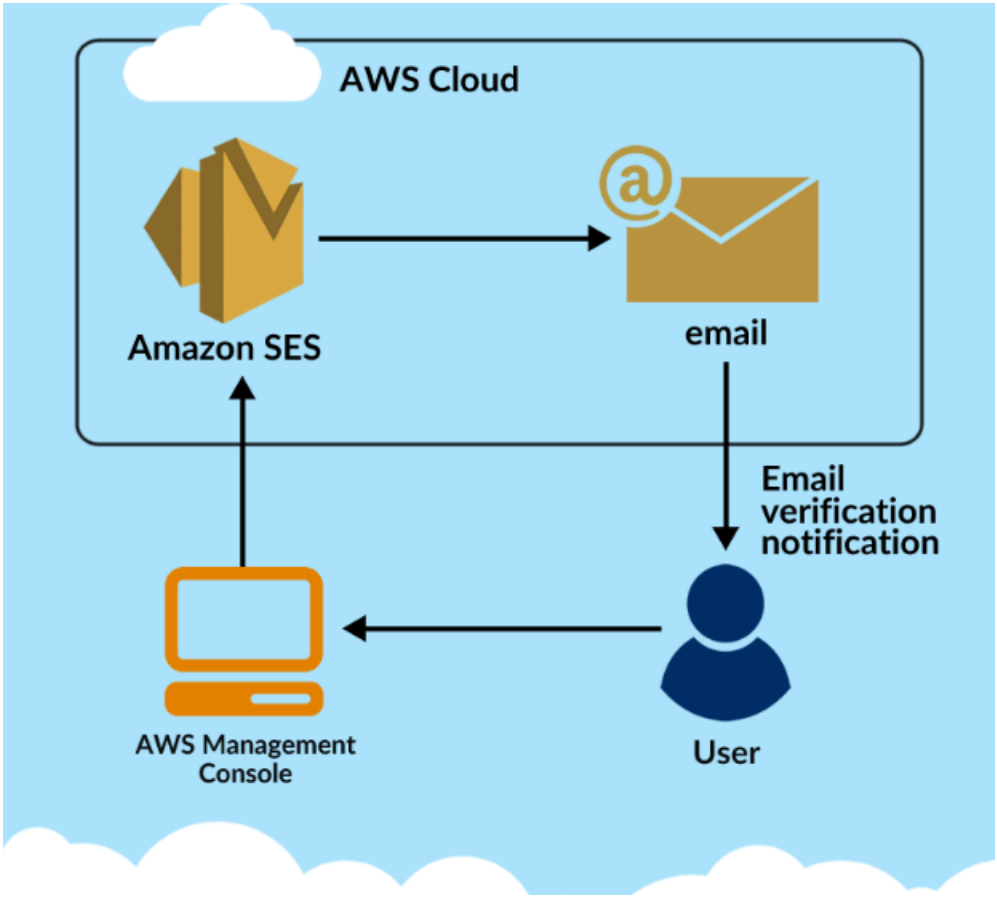
- Lambda:is a serverless compute service provided by Amazon Web Services (AWS). Lambda allows you to run code without provisioning or managing servers. With Lambda, you can execute code in response to events and automatically manage the compute resources needed to run that code. This serverless model eliminates the need to worry about infrastructure, allowing developers to focus solely on their application code.



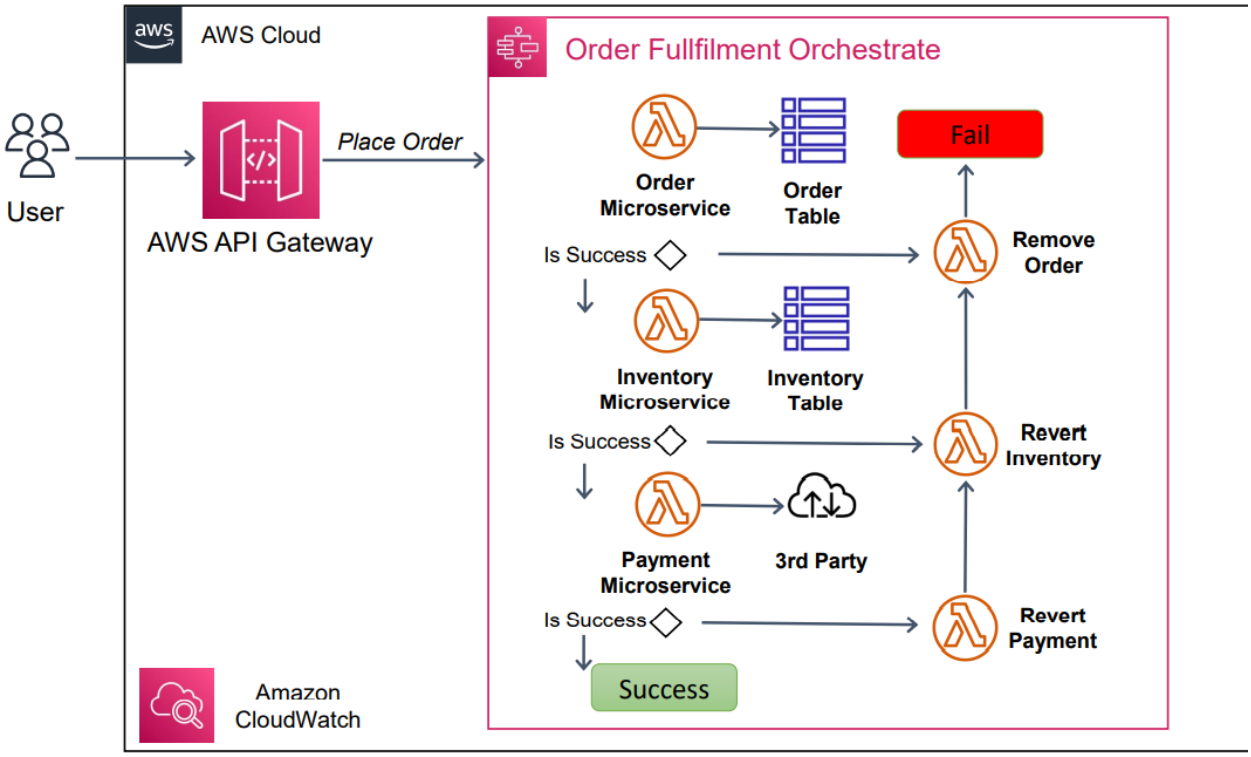
- Cognito: is a fully managed identity service provided by Amazon Web Services (AWS) that makes it easy to add authentication, authorization and user management to your applications. It enables you to securely authenticate users and manage their identities for both web and mobile applications.



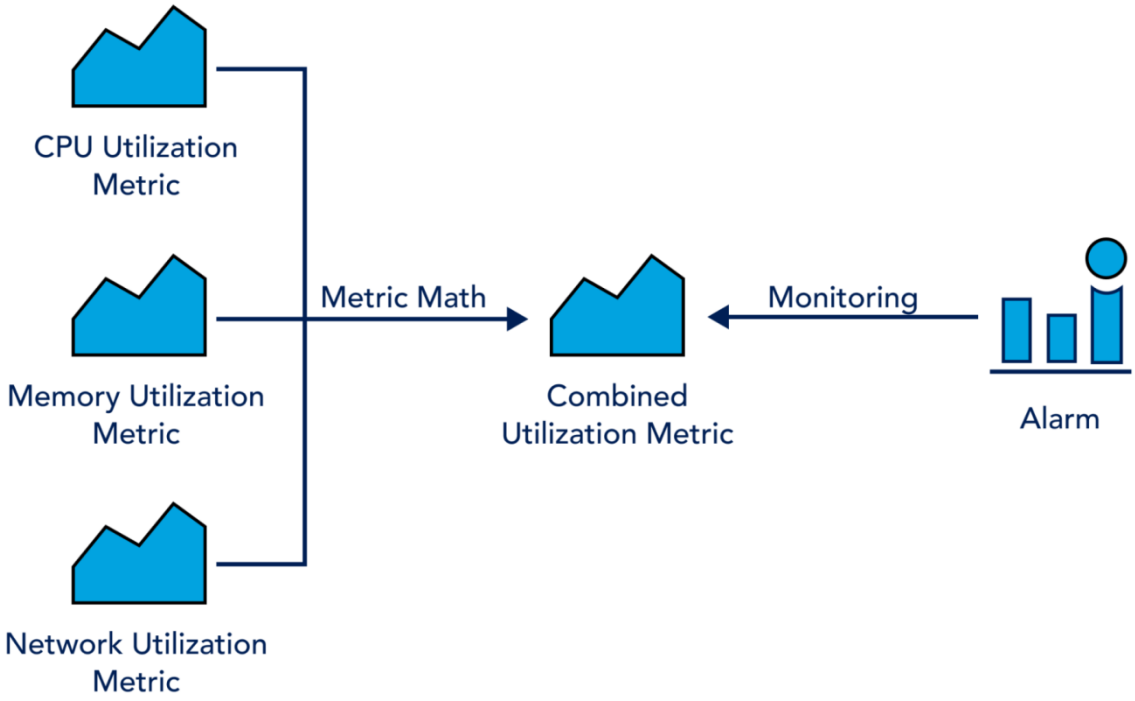
- Simple Email Service(SES): is a fully managed email sending service provided by Amazon Web Services (AWS). SES is designed to help businesses and developers send emails securely, reliably, and at scale. It allows you to send marketing emails, transactional messages, and other types of communications to your customers.



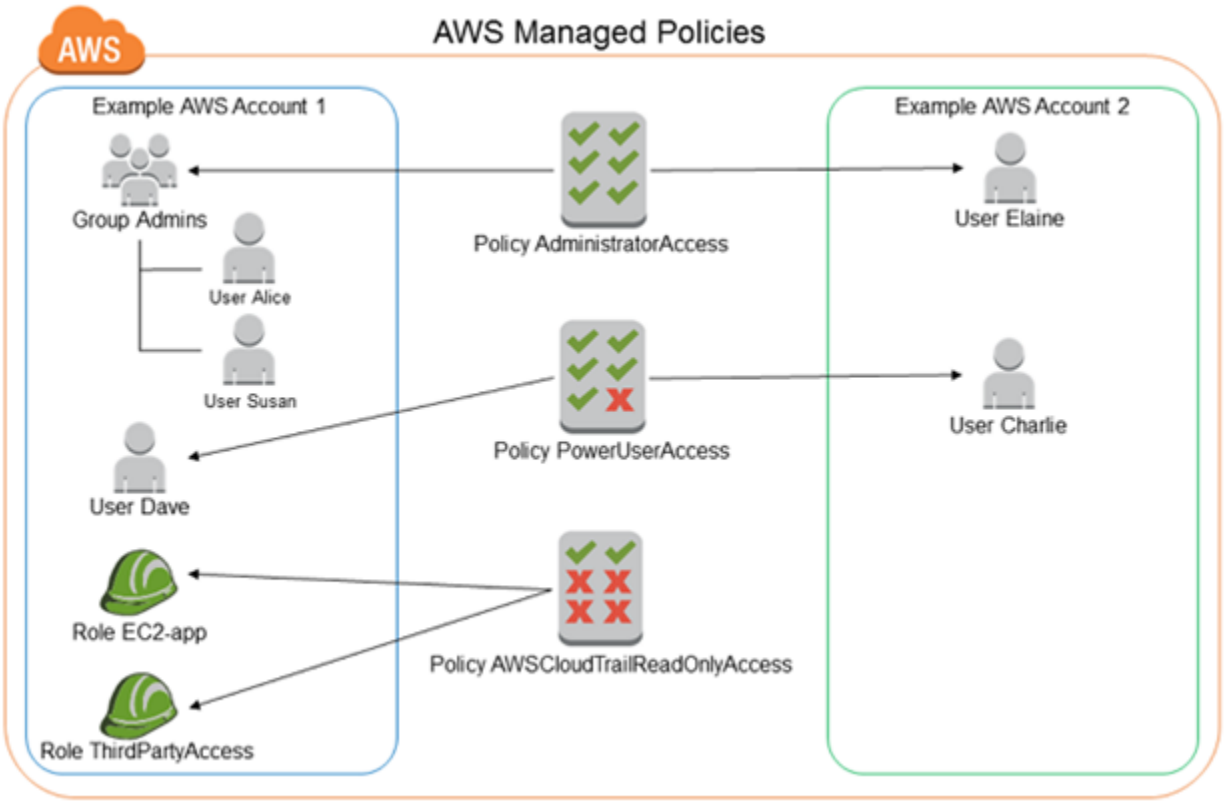
- Step Functions: is a fully managed AWS service that makes it easy to coordinate the components of distributed applications and microservices using visual workflows. Step Functions allows you to design and run workflows that integrate services such as AWS Lambda, AWS Fargate, and more.



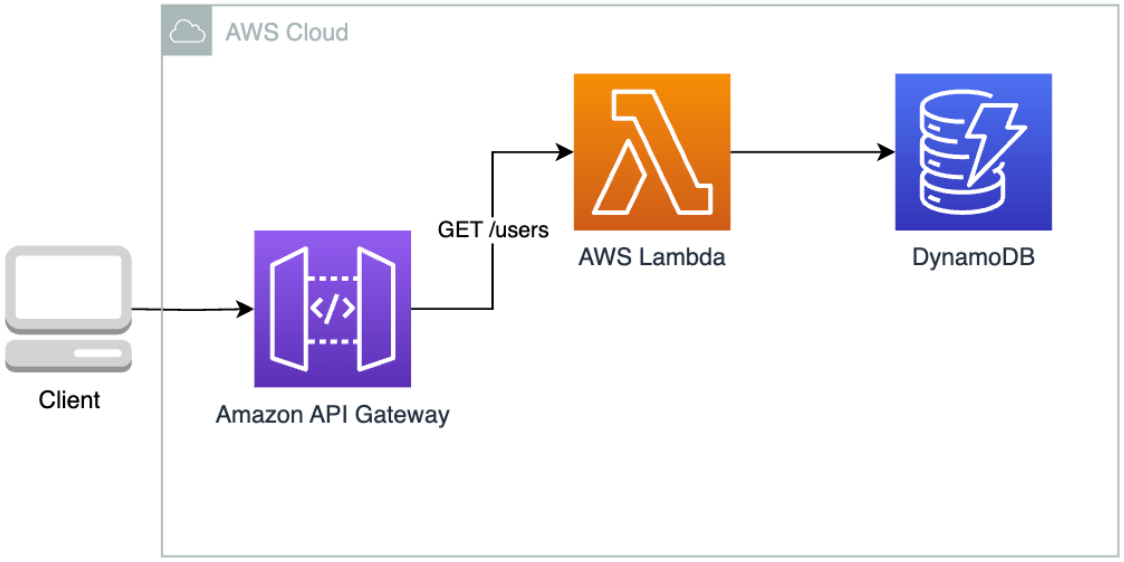
- CloudWatch: is a monitoring and observability service provided by Amazon Web Services (AWS). It is designed to collect and track various metrics, collect and monitor log files and set alarms. CloudWatch allows you to gain insights into your applications,systems and AWS resources helping you ensure the performance, availability and reliability of your applications.



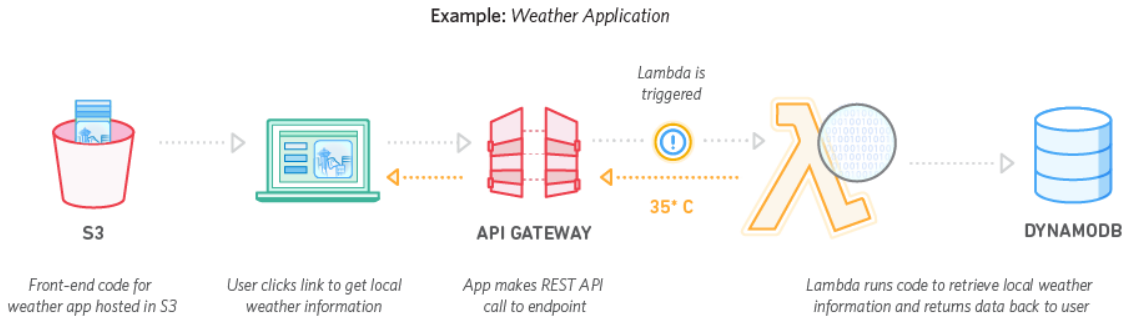
- Identity and Access Management(IAM) role: in AWS (Amazon Web Services) are a fundamental part of AWS security. IAM roles define a set of permissions for making AWS service requests and are not associated with a specific user or group. Instead IAM roles are meant to be assumed by trusted entities such as AWS services, applications or users.



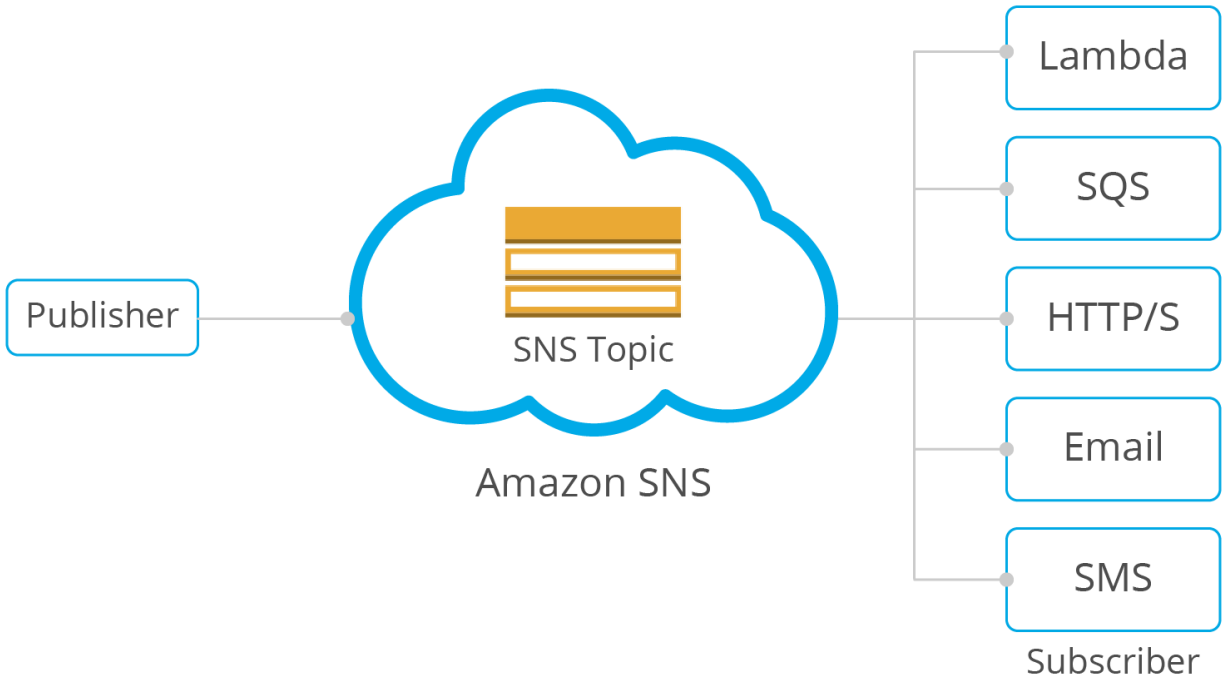
- API Gateway: is a fully managed service in AWS that makes it easy for developers to create, publish, maintain, monitor and secure APIs (Application Programming Interfaces) at any scale. It acts as a gateway for applications to connect with backend services and allows you to create, deploy, and manage APIs quickly and efficiently.



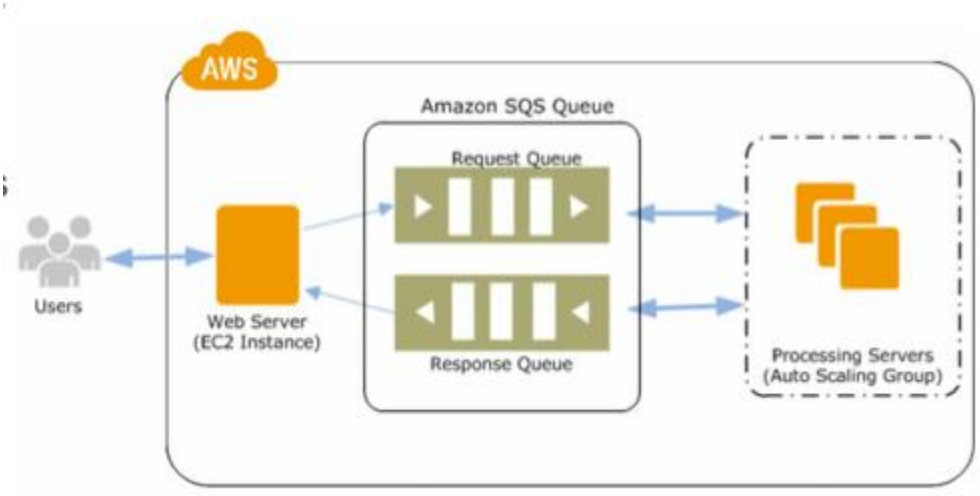
- DynamoDB: is a fully managed NoSQL database service provided by Amazon Web Services (AWS). It is designed to provide fast and predictable performance with seamless scalability. DynamoDB is a key-value and document database that can be used for a wide range of applications, from simple web and mobile applications to complex enterprise-level solutions.



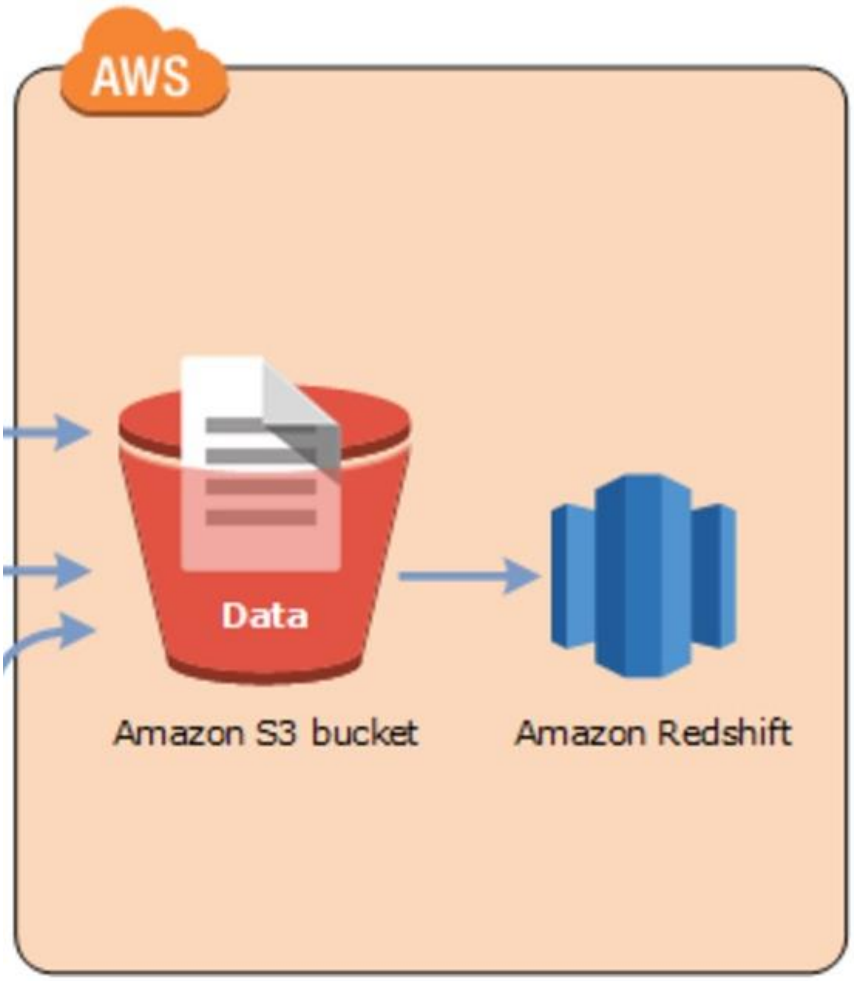
- Simple Notification Service(SNS): is a fully managed messaging service provided by Amazon Web Services (AWS). It enables you to send messages or notifications to a distributed set of recipients via various communication protocols, such as SMS, email, Amazon Simple Queue Service (SQS), AWS Lambda, and more. SNS simplifies the process of sending messages to a large number of subscribers or endpoints.



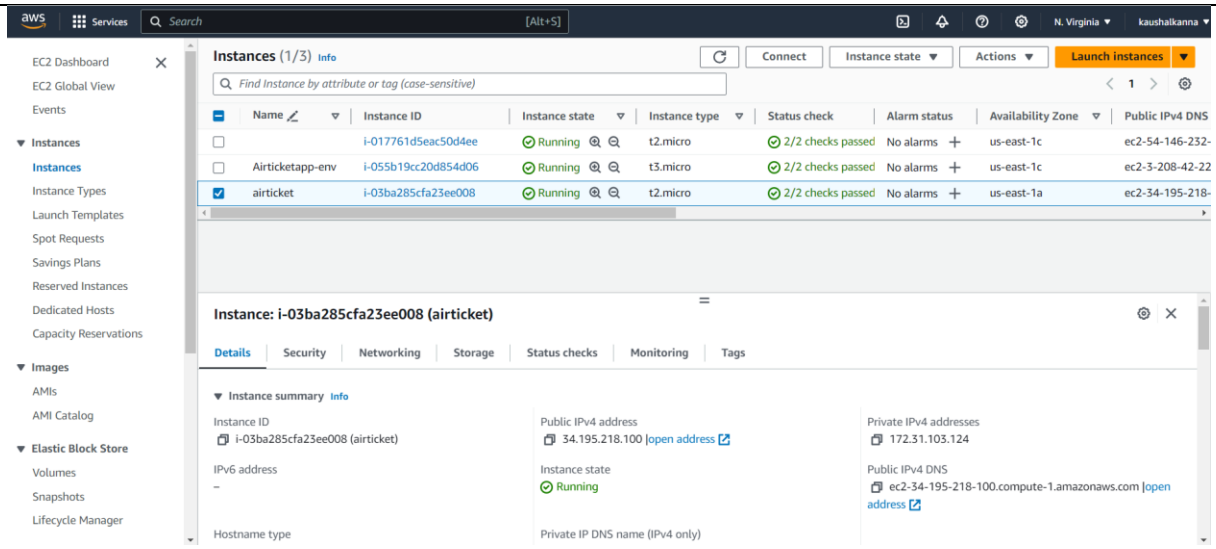
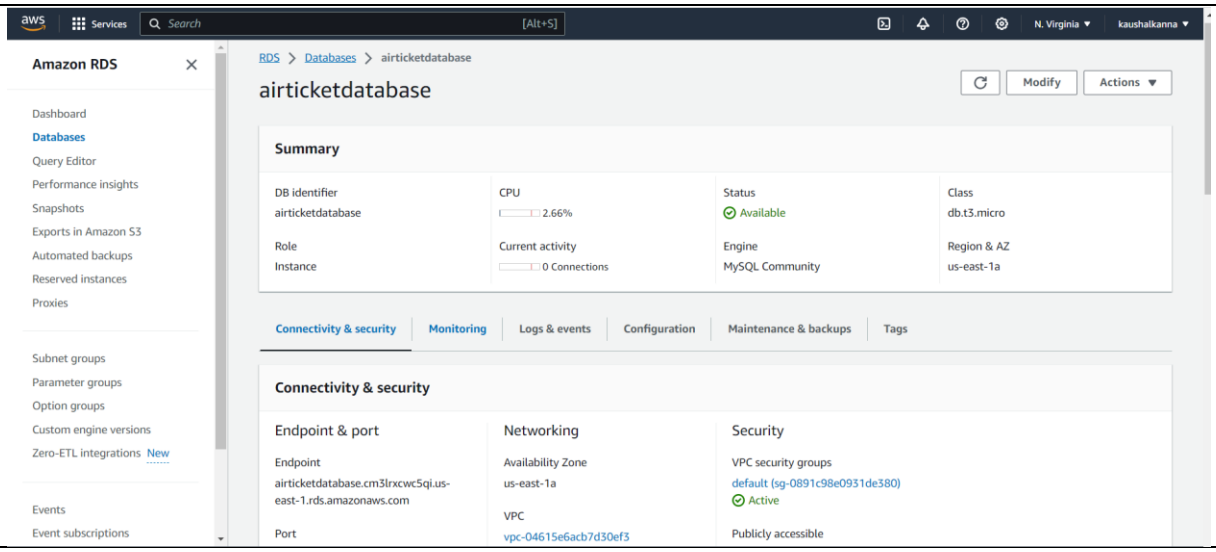
- Simple Queue Service (SQS): is a fully managed message queuing service provided by Amazon Web Services (AWS).SQS enables decoupling of the components of a cloud application allowing them to operate independently and asynchronously.It provides a reliable and scalable way to transmit any volume of data between distributed systems, applications and microservices.

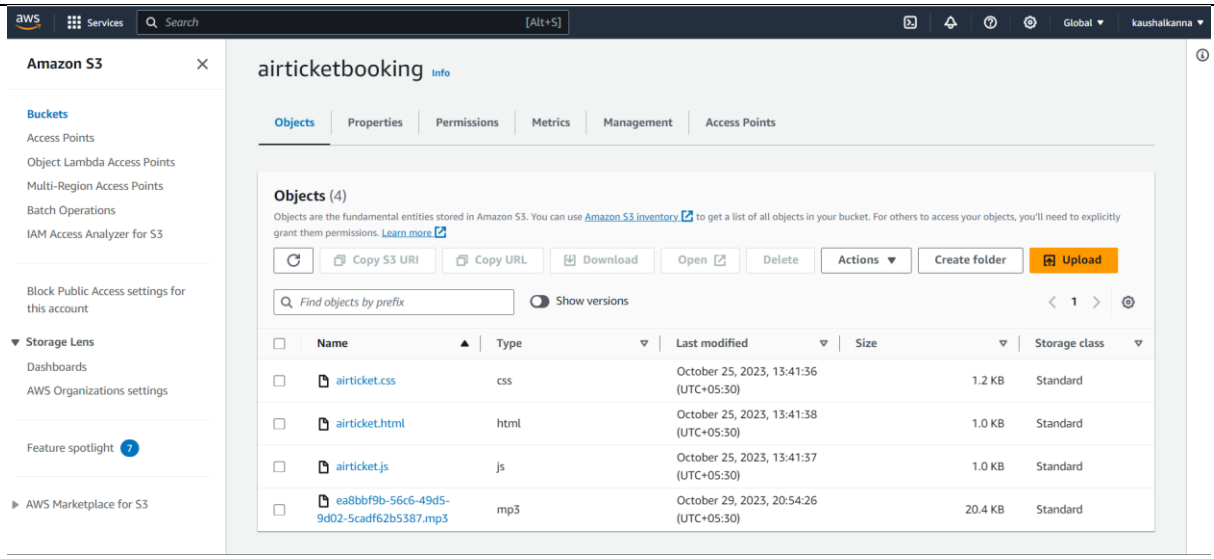
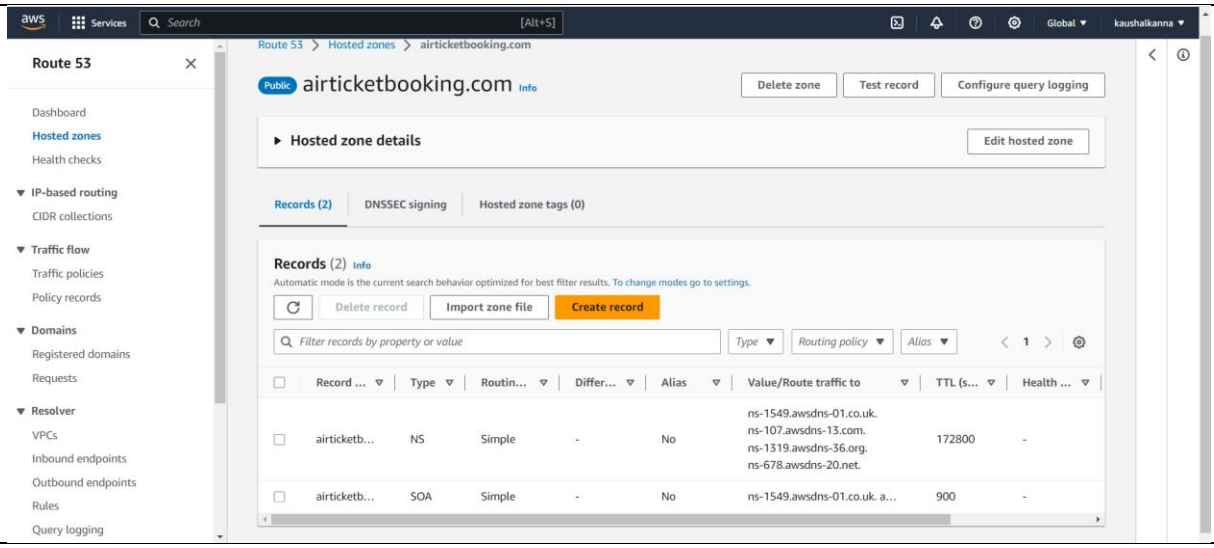


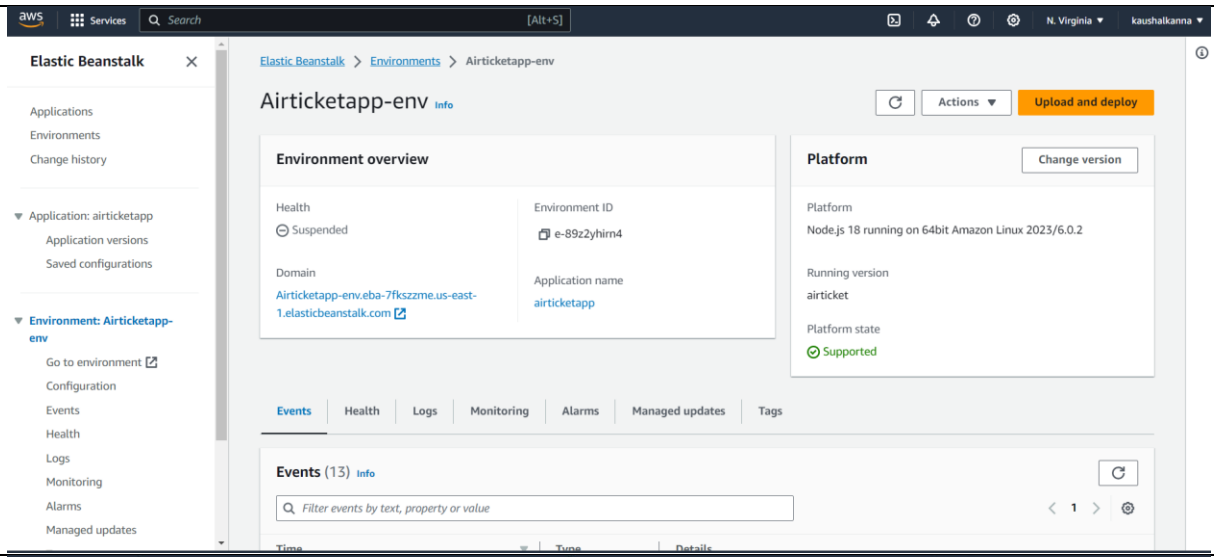
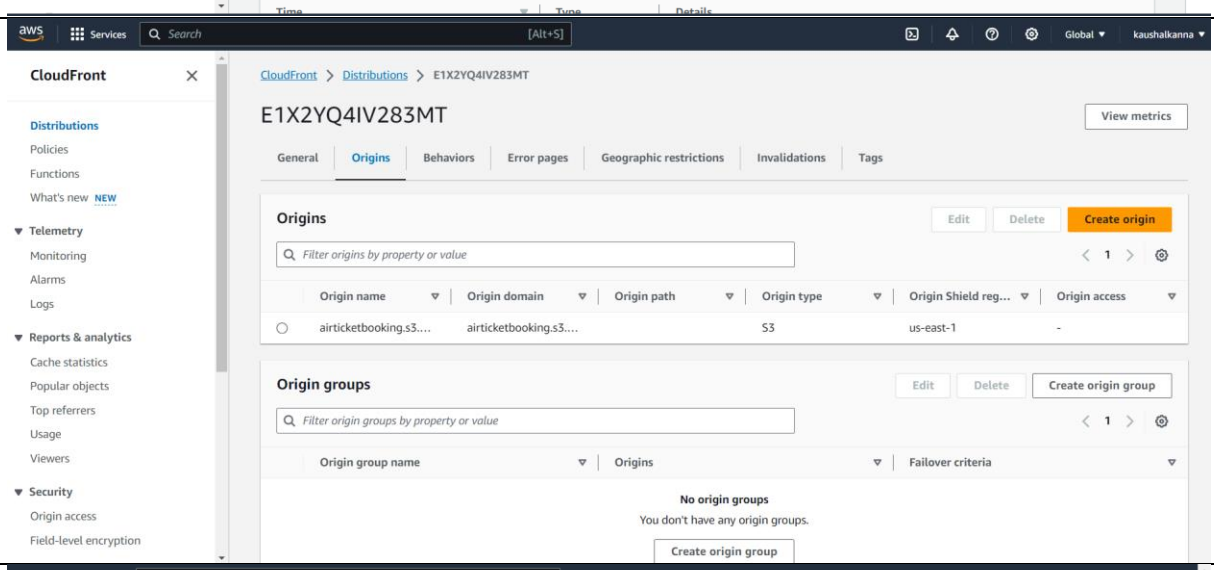
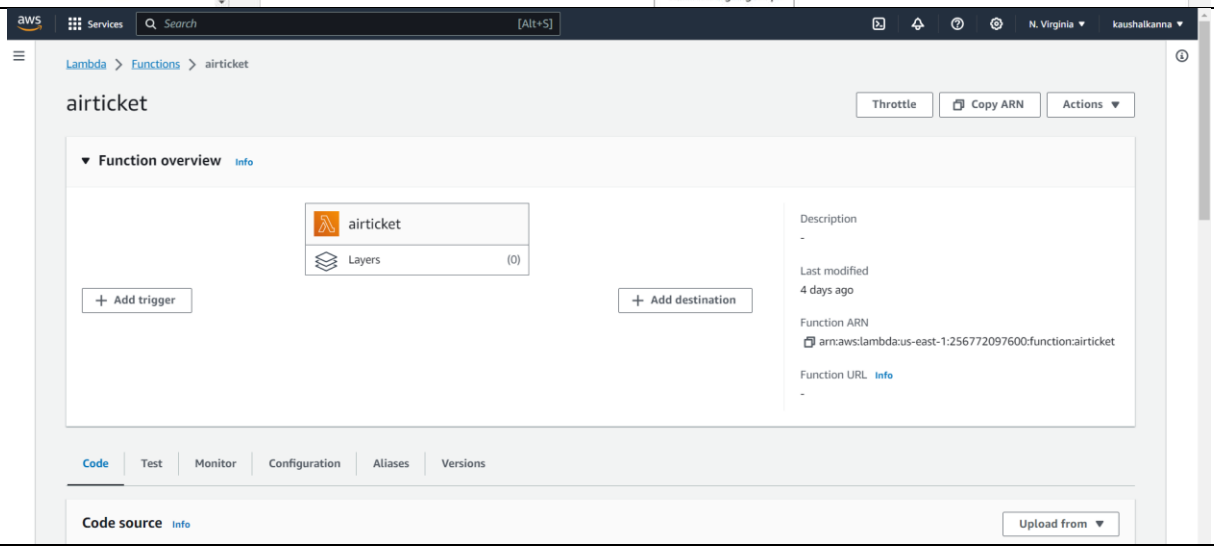
- Redshift:is a fully managed petabyte-scale data warehouse service in the cloud provided by Amazon Web Services (AWS). It is designed to handle large datasets and perform complex queries for analytics and business intelligence purposes. Redshift is based on a scalable,columnar storage architecture that allows for efficient querying and analysis of structured data.

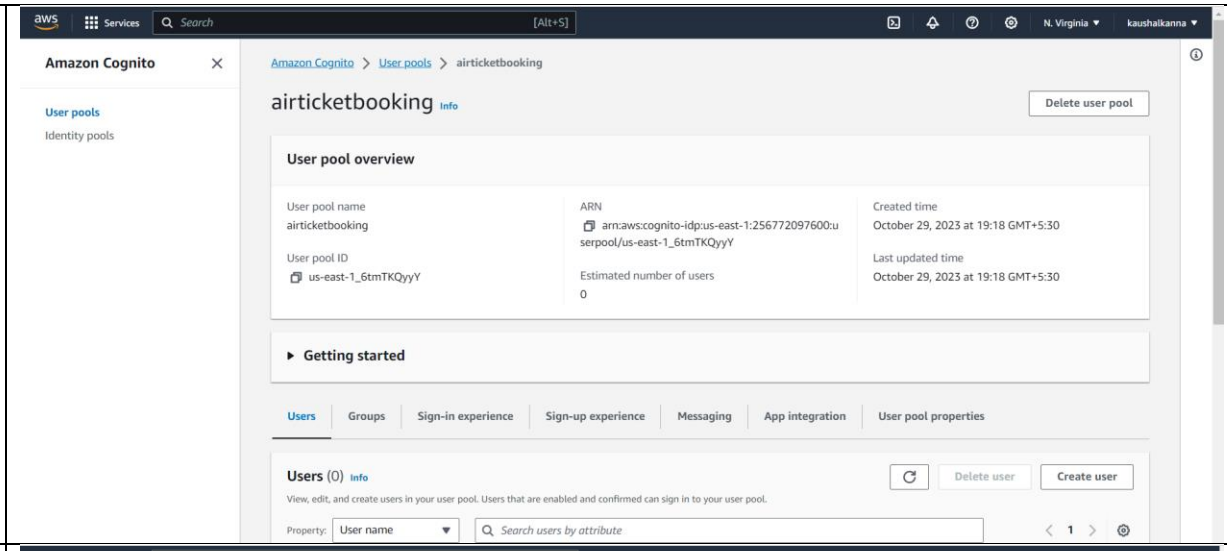
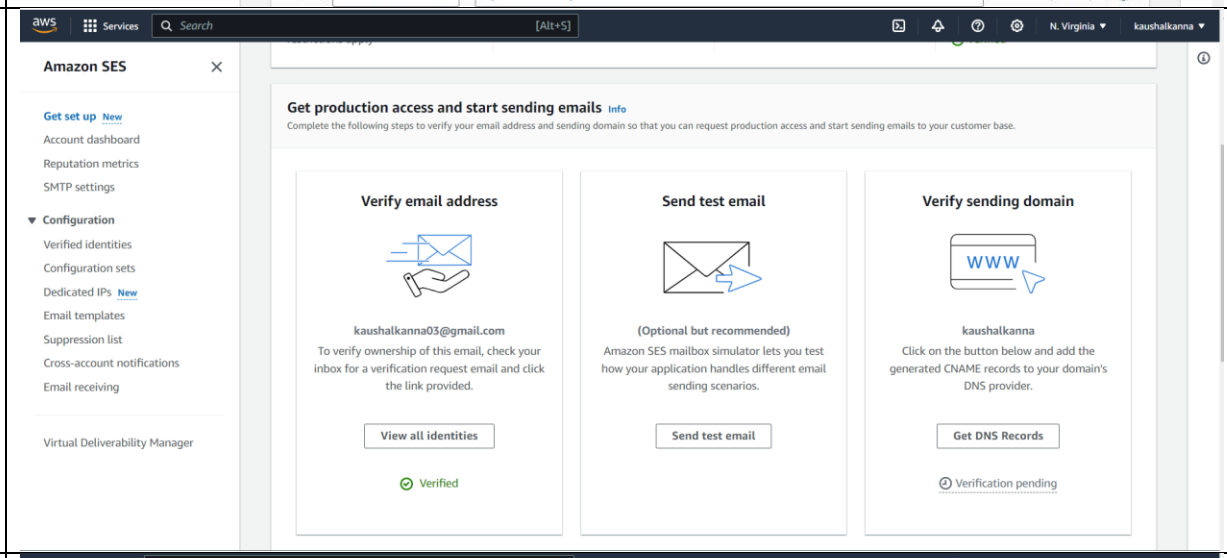
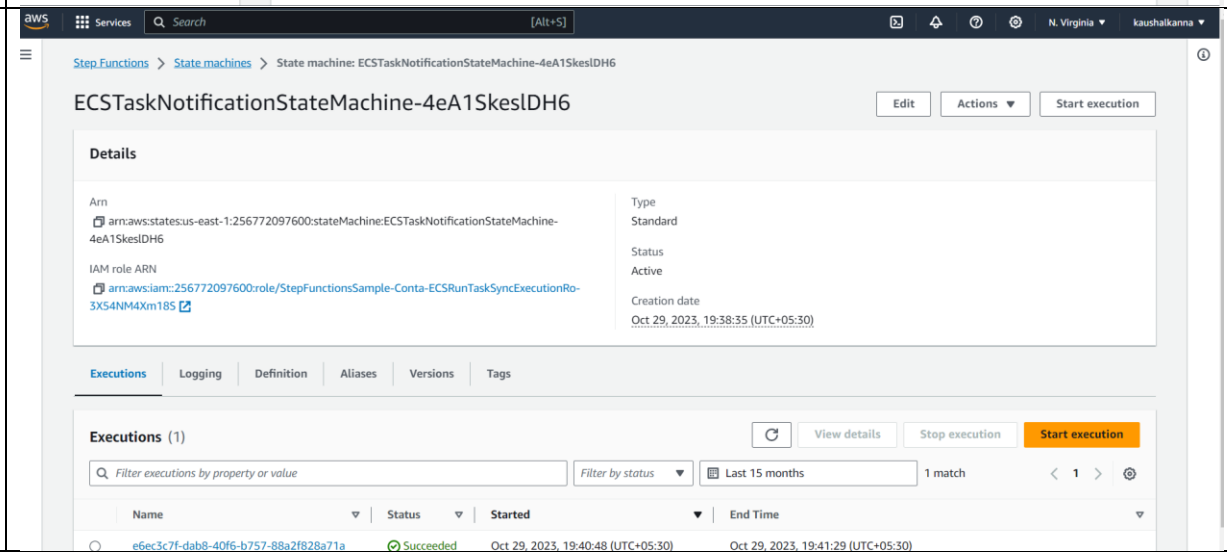


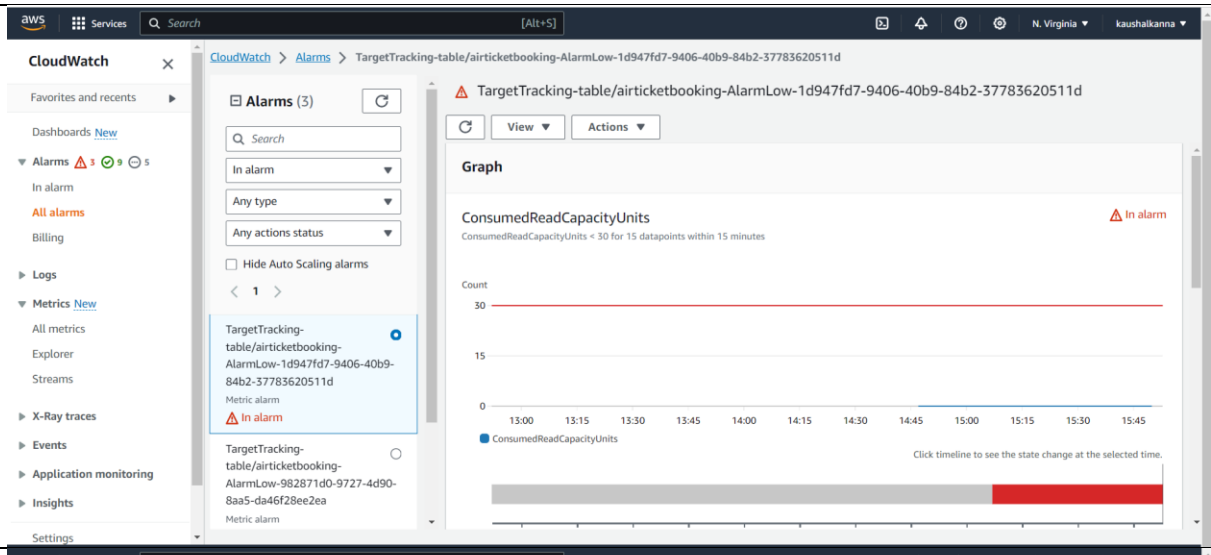
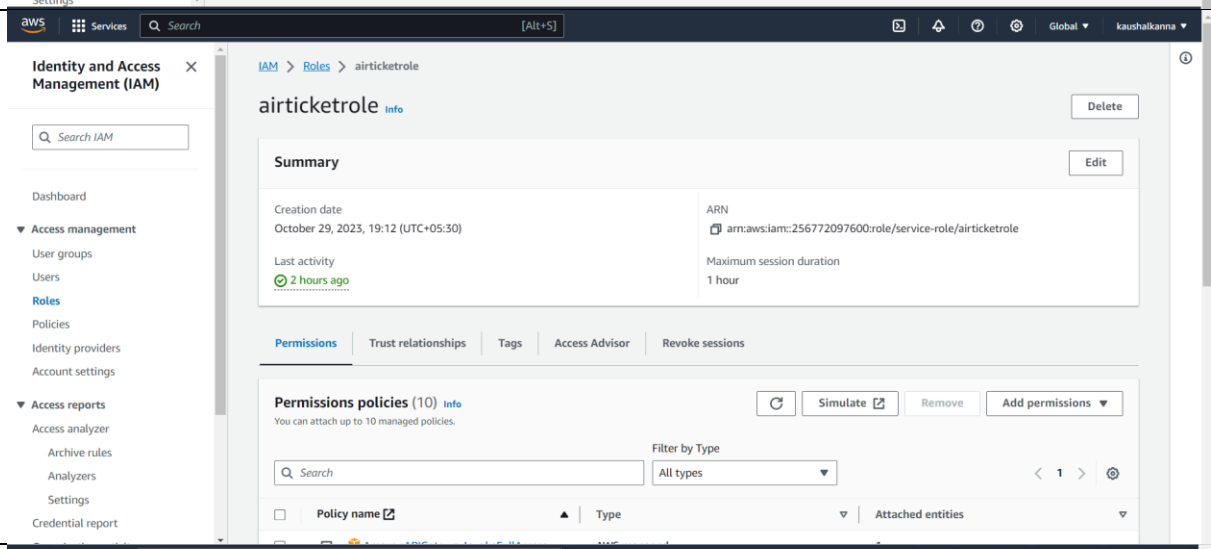
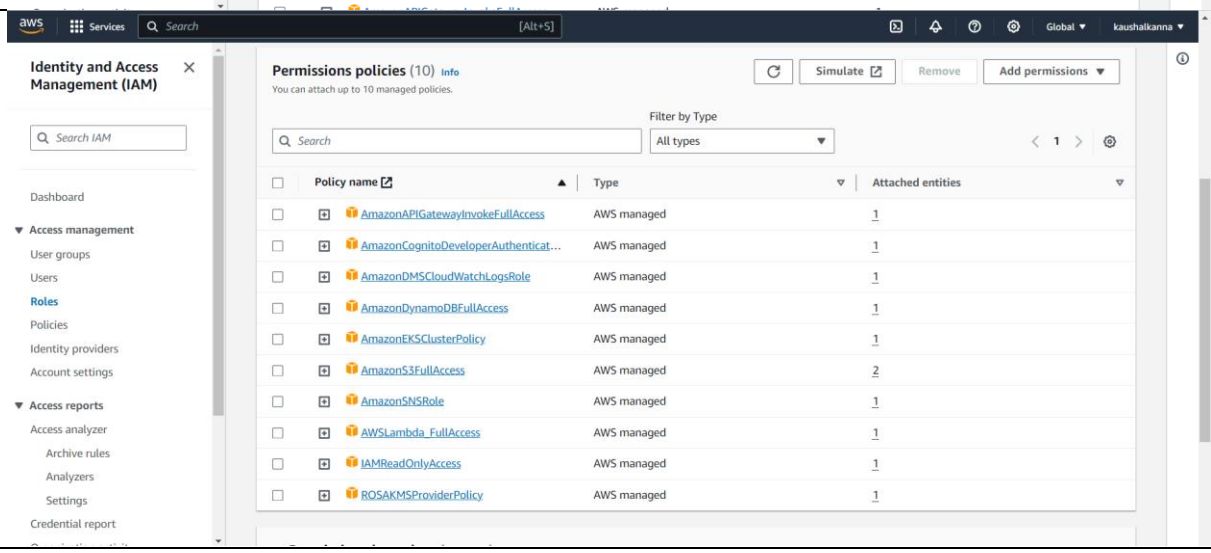
- Polly:is a service provided by Amazon Web Services (AWS) that turns text into lifelike speech. It is a text-to-speech (TTS) service that uses advanced deep learning technologies to convert written text into spoken words. Polly supports a variety of languages and voices allowing developers to create applications with natural-sounding and expressive speech.

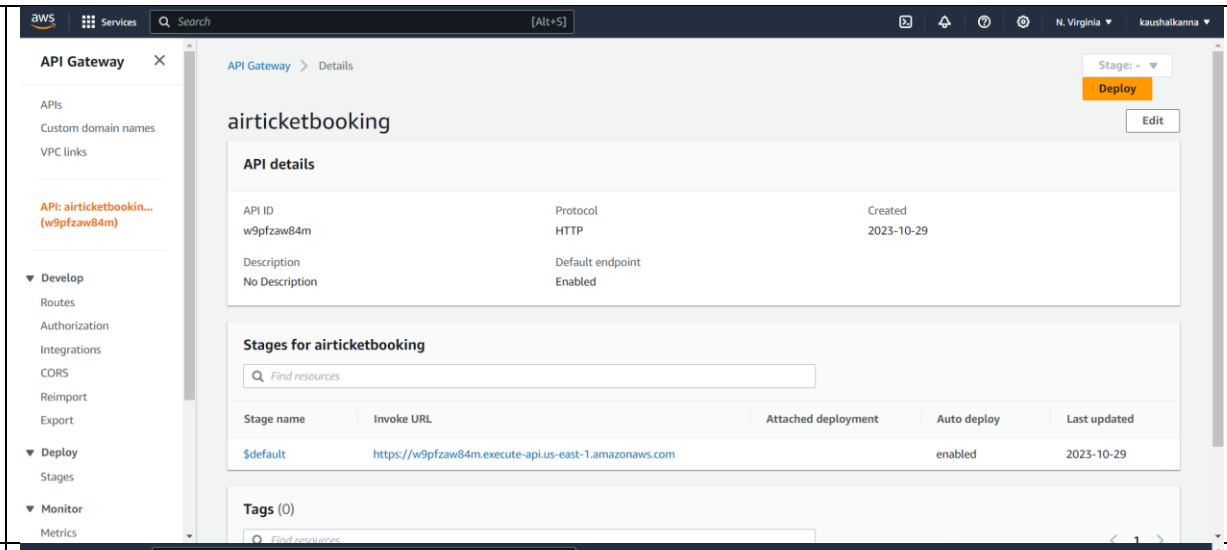
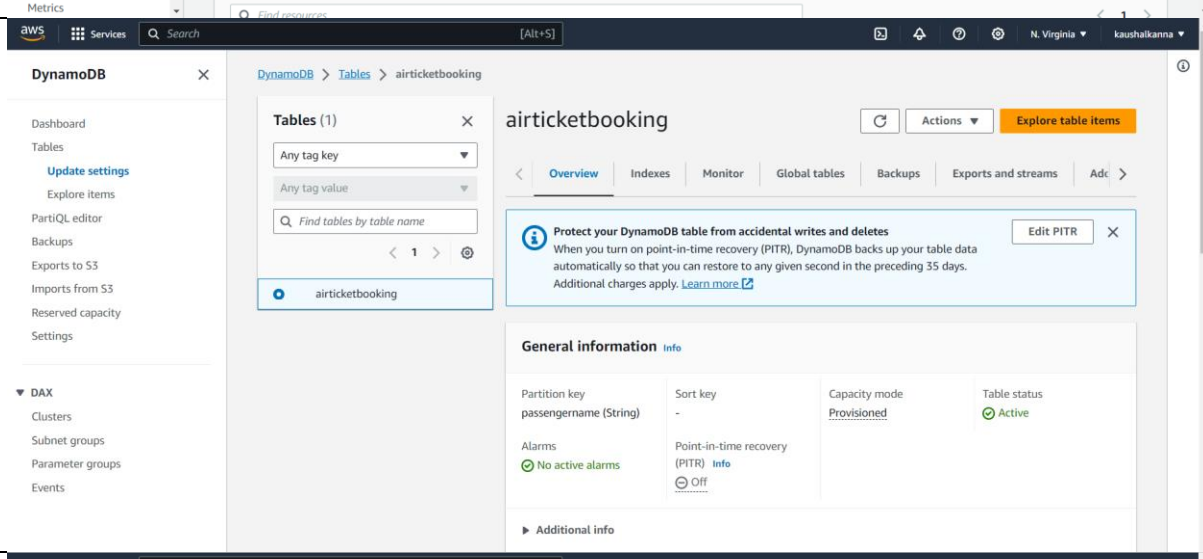
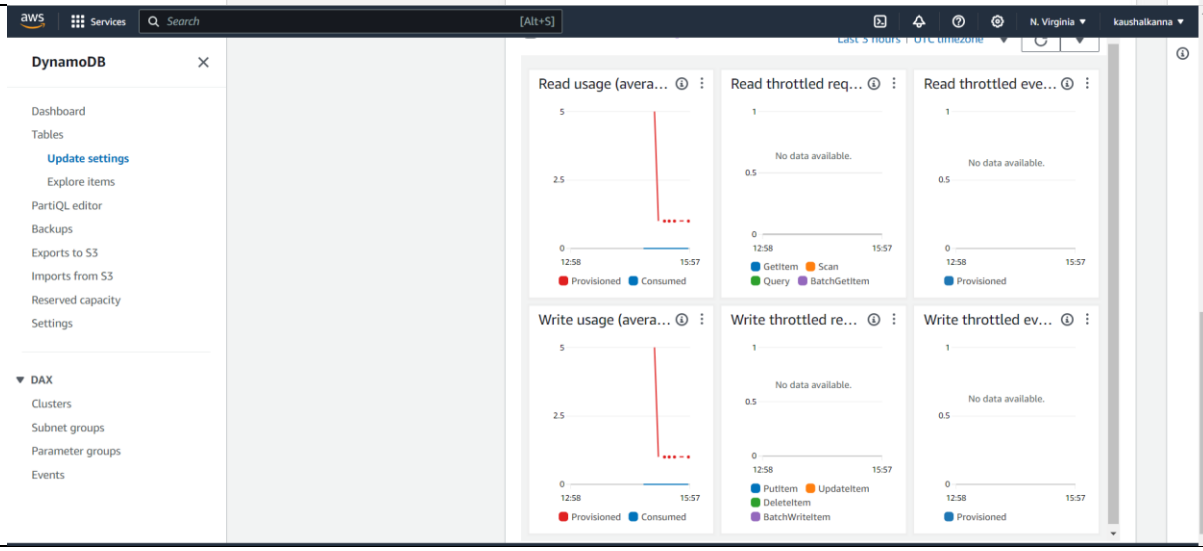
*You can use EC2 instances for running the backend application and handling business logic.	
*Store data such as flight schedules, customers information and bookings in RDS (Relational Database Service).	

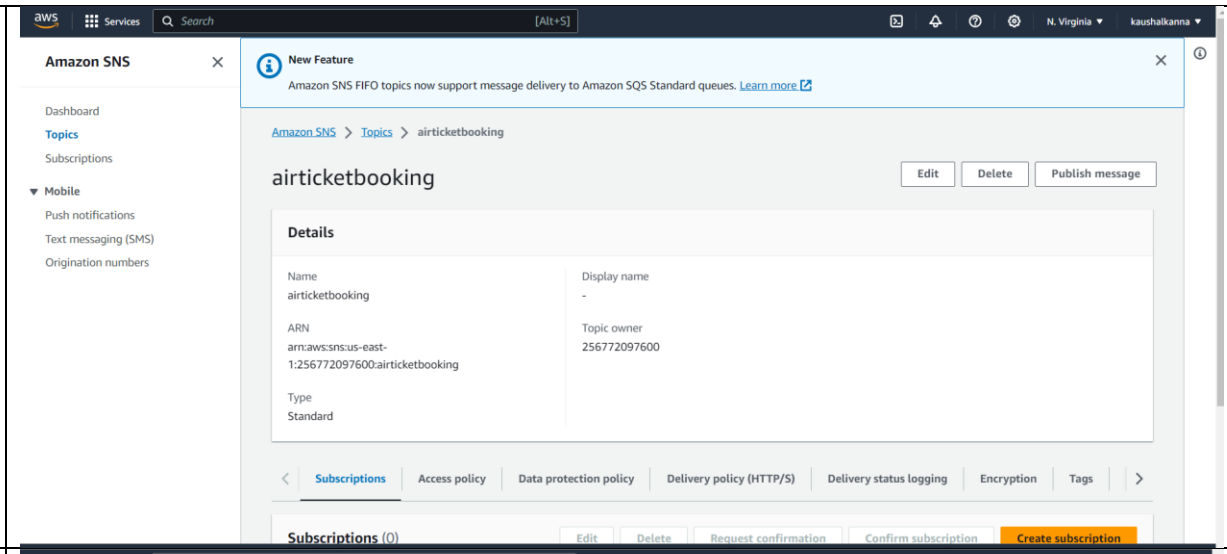
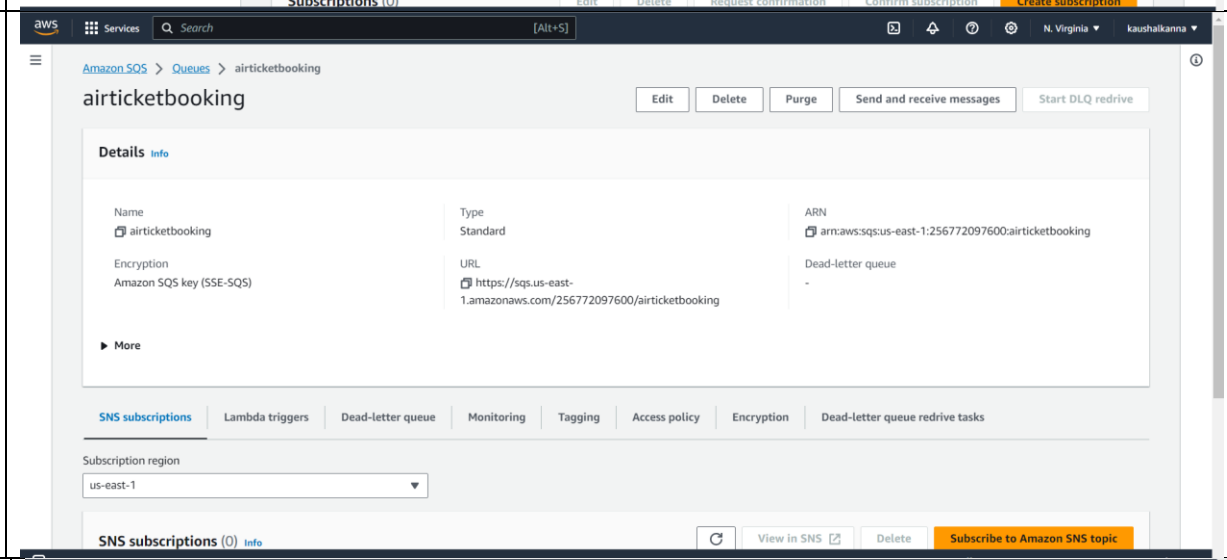
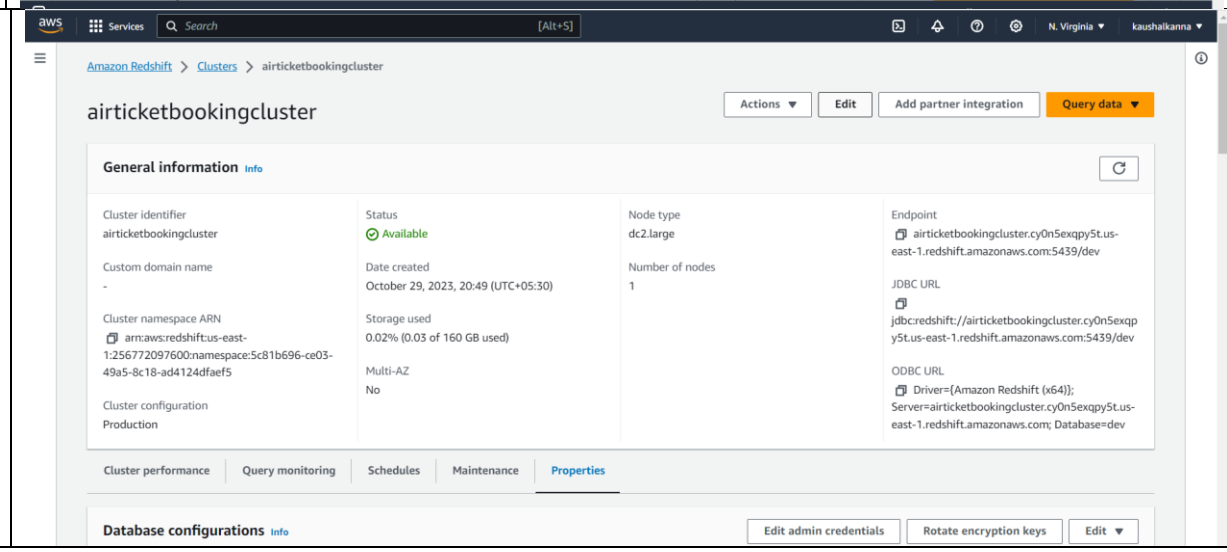
*Store static assets like images and styles in S3 for efficient content delivery.	
*Use Route 53 for domain registration and DNS routing to your application	

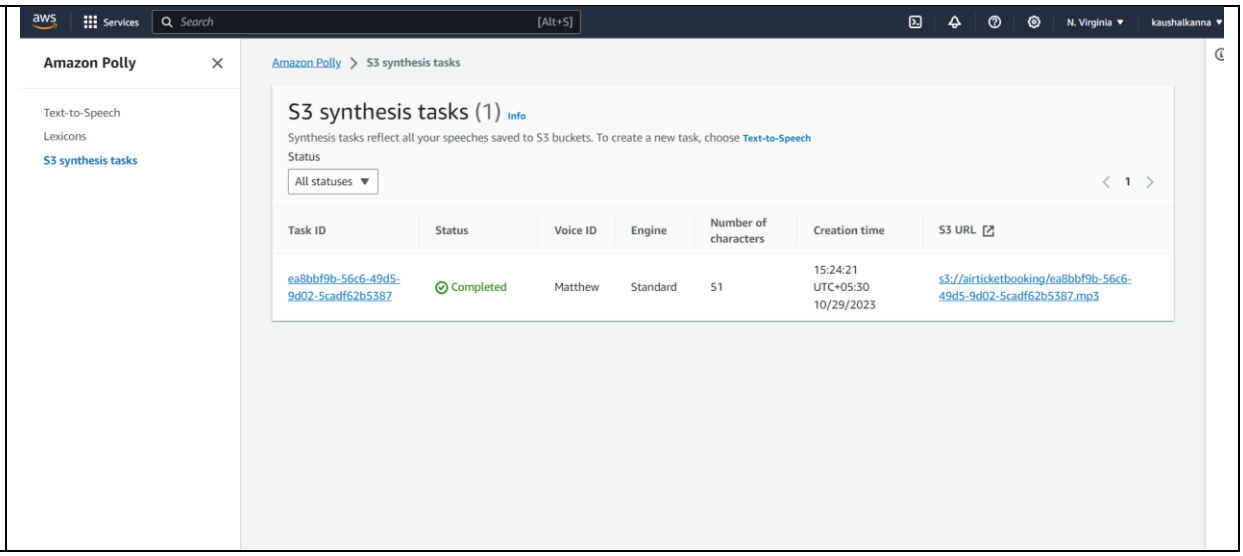
*Deploy the application using Elastic Beanstalk for easy management of the application stack.	
*Set up CloudFront for a Content Delivery Network to improve content delivery and reduce latency	
*Use Lambda for serverless functions that can be triggered for specific tasks such as sending email confirmations	

*Implement Cognito for user authentication and authorization	
*Use SES(Simple Email Service) For sending transactional and promotional emails.	
*Use Step Functions to create workflows for tasks like booking confirmation and payment processing.	

*Set up CloudWatch for monitoring and logging.	
*Define IAM roles and permissions for various components	
*The following iam permission policies to access these services	

*Use API Gateway to create restful APIs for flight search and booking	
*Use DynamoDB If you require a NoSQL database for specific use cases	
*These are the following database usuage by users	

*Use SNS(Simple Notification Service)to push notifications for users.	
*Use SQS(Simple Query Service)If you need a message queuing service for asynchronous processing	
*Use Redshift for data analytics and reporting.	

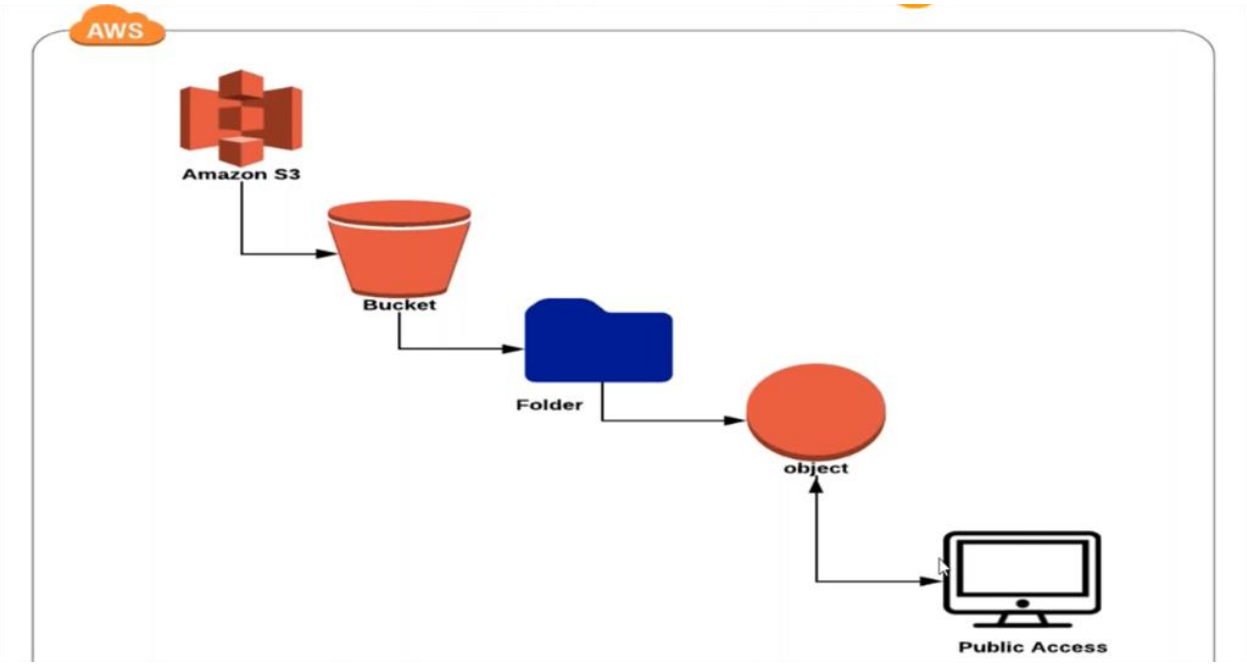
*Use Polly for text-to-speech functionality in your application.	

6.Object file code that are uploaded in S3 bucket

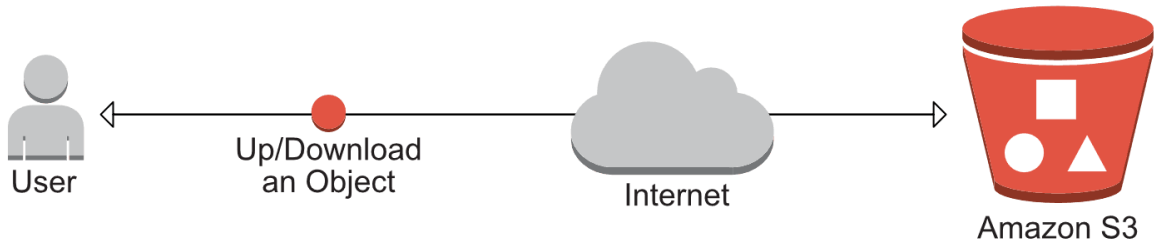
Agenda:

- S3
- Objects
- HTML scripts
- Style scripts

- S3: Simple Storage Service is a scalable and highly durable object storage service provided by Amazon Web Services (AWS). It is designed to store and retrieve any amount of data from anywhere on the web and is widely used for various purposes including data storage, backup, content distribution and hosting static websites.



- Objects: are the fundamental storage entity in S3.Objects are essentially files that can be stored in S3.



- HTML Scripts(Hypertext markup language): typically refers to the code written in the HTML language to define the structure and content of a web page. HTML is a markup language used to create the structure of a webpage by using tags to define elements such as headings, paragraphs, images, links, and more.HTML scripts consist of a series of HTML tags that are interpreted by web browsers to render the content on a webpage.

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
  <title>Code Example</title>
</head>
<body>
  <!-- Load external script with integrity check and crossorigin attribute -->
  <script src="https://HappyCoding.com/scripts/example.js"
    integrity="sha123-BEF6380502"
    crossorigin="anonymous"></script>

  <!-- Rest of HTML content goes here -->
  <h1>Welcome HAPPY CODING!</h1>
  <p>Check out this example showing how to use the script tag with the integrity attribute</p>
</body>
</html>
```

- Style scripts: typically refer to the code written in CSS (Cascading Style Sheets) to define the visual presentation and layout of HTML elements on a webpage. CSS is a stylesheet language used to describe the look and formatting of a document written in HTML or XML.CSS allows developers to control aspects such as colors, fonts, spacing, positioning, and responsiveness of HTML elements. Style scripts written in CSS are used to apply styles to specific HTML elements or groups of elements.

- Html code which has been used to create static website in S3 bucket

```
<!DOCTYPE html>
<!--[if lt IE 7 ]><html class="ie ie6" lang="en"> <![endif]-->
<!--[if IE 7 ]><html class="ie ie7" lang="en"> <![endif]-->
<!--[if IE 8 ]><html class="ie ie8" lang="en"> <![endif]-->
<!--[if (gte IE 9)|!(IE)]><!-->
<html lang="en">
<!--<![endif]-->

<head>
   <meta charset="utf-8">
   <meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1">
   <meta name="description" content="">
   <meta name="author" content="">
   <!--[if IE]>
      <meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1">
      <![endif]-->
```

```
    <title>My Static Website</title>
    <!--REQUIRED STYLE SHEETS-->
    <!-- BOOTSTRAP CORE STYLE CSS -->
    <link href="https://stackpath.bootstrapcdn.com/bootstrap/3.1.1/css/bootstrap.min.css"
rel="stylesheet" />
    <!-- FONTAWESOME STYLE CSS -->
    <link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.5.0/css/all.css"
crossorigin="anonymous">
    <!-- CUSTOM STYLE CSS -->
    <link href="style.css" rel="stylesheet" />
    <!-- GOOGLE FONT -->
    <link href="https://fonts.googleapis.com/css?family=Open+Sans" rel='stylesheet'
type='text/css'>
    <!-- HTML5 shim and Respond.js IE8 support of HTML5 elements and media queries -->
    <!--[if lt IE 9]>
      <script src="https://oss.maxcdn.com/libs/html5shiv/3.7.0/html5shiv.js"></script>
      <script src="https://oss.maxcdn.com/libs/respond.js/1.3.0/respond.min.js"></script>
    <![endif]-->
</head>

<body>
    <!--Header section  -->
    <div class="container" id="home">
        <div class="row text-center">
            <div class="col-md-12">
                <h1 class="head-main"> <i class="fab fa-aws fa-2x"></i> Static Website</h1>
                <h2 class="head-sub-main"> Created by Kaushal </h2>
            </div>
        </div>
    </div>
    <!--End Header section  -->

    <!-- Navigation -->
    <nav class="navbar-inverse" role="navigation">
        <div class="container">
            <div class="navbar-header">
                <button type="button" class="navbar-toggle" data-toggle="collapse" data-
target=".navbar-ex1-collapse">
                    <span class="sr-only">Toggle navigation</span>
                    <span class="icon-bar"></span>
                    <span class="icon-bar"></span>
                    <span class="icon-bar"></span>
                </button>
                <a class="navbar-brand" href="http://aws.amazon.com/partners/">APN Network </a>
            </div>
            <!-- Collect the nav links for toggling -->
            <div class="collapse navbar-collapse navbar-ex1-collapse">
                <ul class="nav navbar-nav">
                    <li><a href="#home"><i class="fas fa-home"></i></a>
                    </li>
```

```
            <li><a href="#about">About</a>
            </li>
            <li><a href="#work-sec">Transformation</a>
            </li>
            <li><a href="#contact-sec">Contact</a>
            </li>
            <li><a href="#map-sec">Location</a>
            </li>
          </ul>
        </div>
        <!-- /.navbar-collapse -->
      </div>
      <!-- /.container -->
    </nav>
    <!--End Navigation -->

    <!--About Section-->
    <section class="color-white " id="about">
      <div class="container">
        <div class="row text-center">
          <div class="col-md-8 col-md-offset-2 ">
            <h2 style="padding-top:50px;">About Us</h2>
            <h4>
              <strong>
                Lorem ipsum dolor sit amet, consectetur adipiscing elit.
                Curabitur nec nisl odio. Mauris vehicula at nunc id posuere.
              </strong>
            </h4>
          </div>
        </div>
      </div>
    </section>

    <section class=" color-light">
      <div class="container">
        <div class="row text-center">
          <div class="col-md-12">
            <h2> Our exclusive services</h2>
            <p>
              Lorem ipsum dolor sit amet, consectetur adipiscing elit.
              Curabitur nec nisl odio. Mauris vehicula at nunc id posuere.
              Lorem ipsum dolor sit amet, consectetur adipiscing elit.
              Curabitur nec nisl odio. Mauris vehicula at nunc id posuere.
              Lorem ipsum dolor sit amet, consectetur adipiscing elit.
              Curabitur nec nisl odio. Mauris vehicula at nunc id posuere.
              Lorem ipsum dolor sit amet, consectetur adipiscing elit.
              Curabitur nec nisl odio. Mauris vehicula at nunc id posuere.
              Lorem ipsum dolor sit amet, consectetur adipiscing elit.
              Curabitur nec nisl odio. Mauris vehicula at nunc id posuere.
              Lorem ipsum dolor sit amet, consectetur adipiscing elit.
```

```
                Curabitur nec nisl odio. Mauris vehicula at nunc id posuere.
                Lorem ipsum dolor sit amet, consectetur adipiscing elit.
                Curabitur nec nisl odio. Mauris vehicula at nunc id posuere.
                Lorem ipsum dolor sit amet, consectetur adipiscing elit.
                Curabitur nec nisl odio. Mauris vehicula at nunc id posuere.
            </p>
        </div>
    </div>
</section>
<!--End About Section-->

<!-- Transformation Section -->
<section class="color-white " id="work-sec">
    <div class="container">
        <div class="row text-center">
            <div class="col-md-8 col-md-offset-2 ">
                <h2>Transformation</h2>
                <h4>
                    <strong>
                        Lorem ipsum dolor sit amet, consectetur adipiscing elit.
                        Curabitur nec nisl odio. Mauris vehicula at nunc id posuere.
                    </strong>
                </h4>
            </div>
        </div>

        <div class="row text-center g-pad-bottom">
            <div class="col-md-4">
                <div class="work-div">
                    <i class="fas fa-box-open fa-5x"></i>
                    <h3>Package </h3>
                </div>
            </div>

            <div class="col-md-4 ">
                <div class="work-div">
                    <i class="fas fa-arrows-alt-h fa-5x"></i>
                </div>
            </div>

            <div class="col-md-4 ">
                <div class="work-div">
                    <i class="fas fa-cloud fa-5x"></i>
                    <h3>The Cloud</h3>
                </div>
            </div>

        </div>
    </div>
```

```
    </section>
    <!--End Transformation Section -->
    <!-- Contact Section -->
    <section class="color-light " id="contact-sec">
        <div class="container">
            <div class="row text-center">
                <div class="col-md-8 col-md-offset-2 ">
                    <h2>Locate Us here</h2>
                    <h4>
                        <strong>
                            Lorem ipsum dolor sit amet, consectetur adipiscing elit.
                            Curabitur nec nisl odio. Mauris vehicula at nunc id posuere.
                        </strong>
                    </h4>
                </div>
            </div>

            <div class="row">
                <div class="col-md-5 contact-cls">
                    <h3>Our Address</h3>
                    <div>
                        <span><i class="fa fa-home"> </i> Address: 123/56, Your City, USA </span>
                        <br />
                        <span><i class="fa fa-phone"> </i> Phone: 82-230-555-8899</span>
                    </div>

                </div>
                <div class="col-md-7">
                    <br />
                    <div id="social-icon">
                        <a href="#"><i class="fab fa-facebook fa-2x"></i></a>
                        <a href="#"><i class="fab fa-twitter fa-2x"></i></a>
                        <a href="#"><i class="fab fa-linkedin fa-2x"></i></a>
                        <a href="#"><i class="fab fa-google-plus fa-2x"></i></a>
                        <a href="#"><i class="fab fa-pinterest fa-2x"></i></a>
                    </div>
                </div>
            </div>
        </div>
    </section>
    <!--End Contact Section -->

    <!--Map Section -->
    <section class="color-light " id="map-sec">
        <iframe class="cnt"
src="https://www.google.com/maps/embed?pb=!1m18!1m12!1m3!1d2999841.293321206!2d-75.80920404999999!3d42.75594204999997!2m3!1f0!2f0!3f0!3m2!1i1024!2i768!4f13.1!3m3!1m2!1s0x4ccc4bf0f123a5a9%3A0xddcfc6c1de189567!2sNew+York!5e0!3m2!1sen!2s!4v1395313088825"></iframe>
    </section>
```

```
    <!--End Map Section -->
    <!--footer Section -->
    <div class="for-full-back " id="footer">
       CSS Template from <a href="https://www.free-css.com/free-css-templates/page203/image-less">Free CSS</a>
    </div>
    <!--End footer Section -->
    <!-- JAVASCRIPT FILES PLACED AT THE BOTTOM TO REDUCE THE LOADING TIME  -->
    <!-- CORE JQUERY  -->
    <script src="https://code.jquery.com/jquery-1.10.2.min.js"></script>
    <!-- BOOTSTRAP CORE SCRIPT   -->
    <script src="https://stackpath.bootstrapcdn.com/bootstrap/3.1.1/js/bootstrap.min.js"></script>
    <!-- CUSTOM SCRIPTS -->
    <script src="script.js"></script>

<script defer
src="https://static.cloudflareinsights.com/beacon.min.js/v8b253dfea2ab4077af8c6f58422dfbfd1689876627854" integrity="sha512-
bjgnUKX4azu3dLTVtie9u6TKqgx29RBwfj3QXYt5EKfWM/9hPSAI/4qcV5NACjwAo8UtTeWefx6Zq5PHcMm7Tg==" data-cf-
beacon='{"rayId":"814ef2a9294440bc","token":"a73834a4a1444e9ab89e8da06da41720","version":"2023.8.0","si":100}' crossorigin="anonymous"></script>
</body>

</html>
```

- This is the style script which has been used to create static website

```
/*=======================================

    GENERAL STYLES

=================================================*/

body {

        font-family:'Open Sans', sans-serif;

    font-size:14px;

  background-color:#9B60DB;

}


.nav a {
```

```
    color:#ffffff !important;
}
.navbar-header a {
    color:#ffffff !important;
    padding-right:100px;
}

.text-center {
    text-align:center;
}

h1, h2, h3, h4, h5, h6 {
    font-family:'Open Sans', sans-serif;
}

h1 {
    line-height:80px;
    font-weight:900;
    font-size:60px;
    padding:30px 20px 10px 20px;
}
h2 {
    line-height:40px;
    font-weight:900;
    font-size:30px;
     padding:20px 20px 10px 20px;
```

```
     text-transform:uppercase;
}
h3 {
    line-height:30px;
    padding-bottom:20px;
}
h4 {
    line-height:40px;
    padding-bottom:15px;
}

p {
    font-weight:300;
    line-height:30px;
    padding-bottom:20px;
}

.space-free {
    height:100px;
}


section {
    padding-top:50px;
    padding-bottom:30px;
}
```

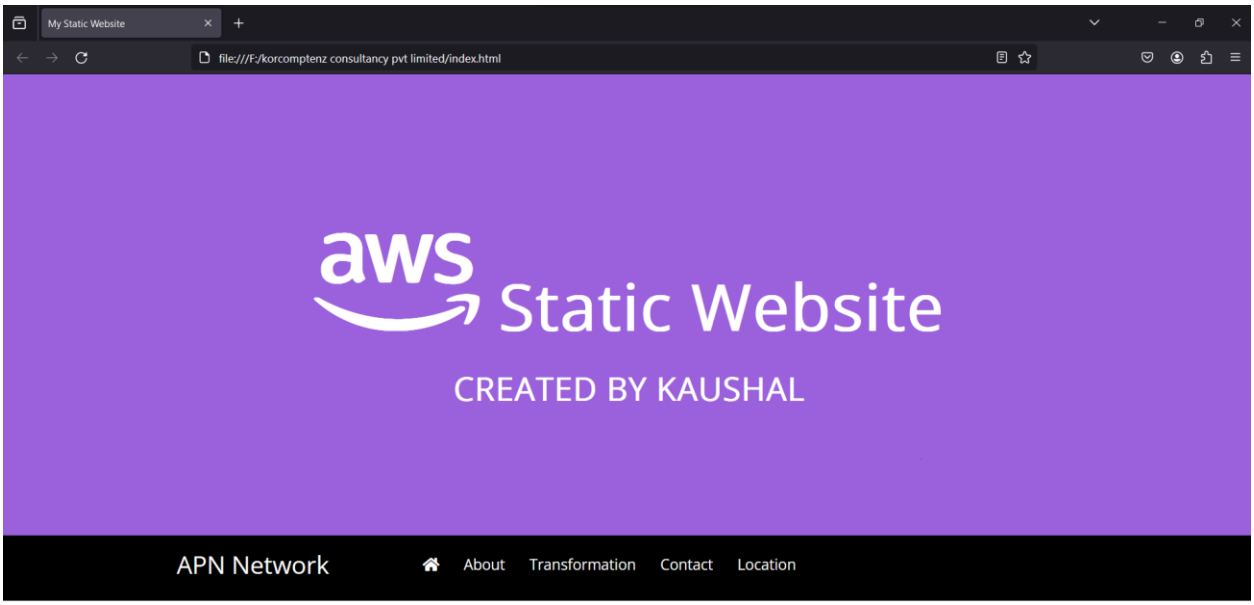
```
/*=======================================
    COLOR CODES
=================================================*/
.color-white {
   background-color:#fff!important;
}
.color-dark {
   background-color:#696969!important;
}

.color-light {
   background-color:#E9E9E9!important;
}

/*=======================================
    HEADER STYLES
=================================================*/
.head-main {
    font-size:80px !important;
    font-weight:900!important;
    color:#fff!important;
    padding:150px 20px 10px 20px;
}
.head-sub-main {
    font-size:40px !important;
```

```
    font-weight:600!important;
    color:#fff!important;
    padding:5px 20px 150px 20px;
}


/*=======================================
    NAVIGATION STYLES
=================================================*/


nav {
        position: absolute;
        width: 100%;
    background: #fff;
    z-index:99;
}
.fixed {
        position: fixed;
        top: 0;
        min-height: 50px;
        z-index: 99;
}

.navbar-inverse {
background-color: #000000;
```

```
border-color: #000000;
min-height: 80px;
padding-top: 10px;
font-size: 18px;
}
.navbar-brand {

font-size: 30px;
}

/*=======================================
   WORK FLOW STYLES
=================================================*/



.work-div {
        width: 100%;
        border: 0;
   padding:20px;
   margin-bottom:30px;
}
/*=======================================
   CONTACT STYLES
=================================================*/

#social-icon a {
```

```
    color:#616161!important;
    margin-right:10px;
}
.contact-cls {
    font-size:20px;
    line-height:40px;
}
.cnt {
width: 100%;
min-height: 350px;
border: 1px solid #e1e1e1;
}
/*=======================================
    FOOTER STYLES
=================================================*/


#footer {
    background-color:#000;
    color:#fff;
    padding:20px 50px 20px 50px;
    text-align:right;
}
```

- This is the following static website created from above html and style scripts in S3 bucket for static website hosting



- Html code to create airticketbooking website in S3 bucket

```
<!DOCTYPE html>
<html>
<head>
   <title>Flight Booking Form</title>
</head>
<body>
   <h1>Flight Booking</h1>
   <form action="book_flight.php" method="post">
      <label for="from">From:</label>
      <input type="text" id="from" name="from" required><br><br>

      <label for="to">To:</label>
      <input type="text" id="to" name="to" required><br><br>
```

```
        <label for="departure">Departure Date:</label>
        <input type="date" id="departure" name="departure" required><br><br>

        <label for="return">Return Date:</label>
        <input type="date" id="return" name="return"><br><br>

        <label for="passengers">Passengers:</label>
        <select id="passengers" name="passengers">
            <option value="1">1</option>
            <option value="2">2</option>
            <option value="3">3</option>
            <option value="4">4</option>
            <option value="5">5</option>
        </select><br><br>

        <input type="submit" value="Book Now">
    </form>
</body>
</html>
```

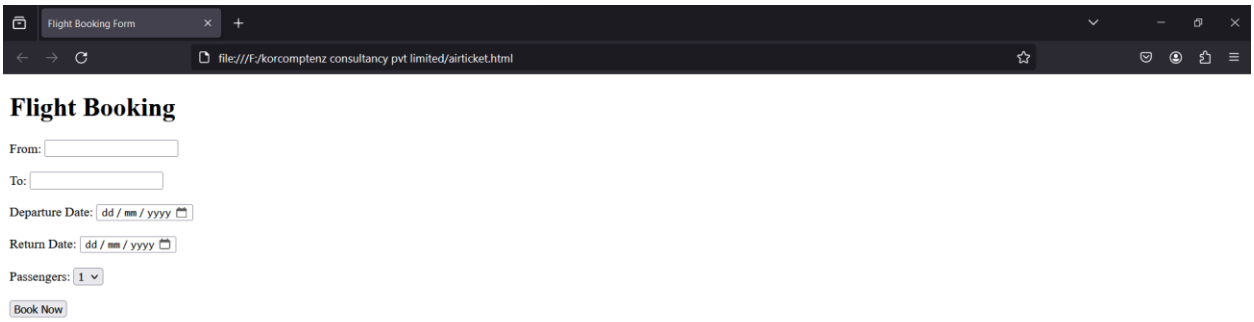
- This is the style script used for airticketbooking website in S3 bucket

```
/* Reset some default styles for consistency */

body, html {

  margin: 0;

  padding: 0;

  font-family: Arial, sans-serif;

}


/* Style the container for the booking system */

.booking-container {

  width: 80%;

  margin: 0 auto;

  padding: 20px;

  background-color: #f0f0f0;

  border: 1px solid #ccc;

  border-radius: 5px;
```

```
  box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
}

/* Style the header */
.header {
  text-align: center;
  font-size: 24px;
  margin-bottom: 20px;
}

/* Style flight listings */
.flight-list {
  list-style: none;
  padding: 0;
}

.flight-item {
  border: 1px solid #ddd;
  border-radius: 5px;
  padding: 10px;
  margin: 10px 0;
  background-color: #fff;
}

/* Style buttons */
.button {
  background-color: #007BFF;
  color: #fff;
```

```
  border: none;
  padding: 10px 20px;
  border-radius: 5px;
  cursor: pointer;
}

.button:hover {
  background-color: #0056b3;
}

/* Style the footer */
.footer {
  text-align: center;
  margin-top: 20px;
  color: #777;
}

/* Media query for responsiveness (adjust breakpoints as needed) */
@media (max-width: 768px) {
  .booking-container {
    width: 100%;
  }
}
```

- This is the airticket booking website that been created from html and style scripts in S3 bucket.



- This is the Hello World linux user datascript to display hello world website

```
#install httpd(Linux 2 Version)

yum update -y

yum install -y httpd

systemctl start httpd

systemctl enable httpd

echo "<h1> Hello World from $(hostname -f)</h1>">/var/www/html/index.html
```

Great Place To Work®
Certified
APR 2023 - APR 2024
INDIA

Korcomptenz

Korcomptenz is a technology transformation provider that partners with clients to improve their digital experience and insights, We unlock the power of technology in the areas of ERP,CRM, infrastructure management, and cloud to empower our clients with intelligent and experiential solutions. We #FocusOnYou

Discover how we #FocusOnYou at www.korcomptenz.com

Get in touch with us:

+97142285285 | sales@korcomptenz.com