Cloud Security & Management

Title:

Implementing various security tools on E-commerce Website Deployed on AWS.

Presented by:

Ayush Juyal(R214220316)
Rishabh Anand(R2142201862)

Content

20. Implementing CloudFront
21. Implement KMS
22. Creating RDS and implementing security Features
23. SOT Analysis
24. Pert Chart
25. Objectives covered

1. Introduction

- Now-a-days Online shopping offers unparalleled convenience to customers as they can shop from the comfort of their homes or offices. This has made online shopping an attractive option for people who are short on time or have busy schedules.
- Ecommerce websites allow businesses to reach a global audience without having to set up physical stores in different locations. This makes it easier for businesses to expand their reach and find new customers.
- Even Setting up an ecommerce website is often more cost-effective than setting up a physical store. This is because there are no rent or utility expenses to worry about, and businesses can also save money on staffing costs.

Contd.

- Ecommerce sites are susceptible to cyber-attacks and data breaches. A data breach can lead to huge financial losses for the ecommerce company and harm their reputation.
- A website breach also lead to downtime, meaning that your website is inaccessible to visitors. Securing your website helps prevent downtime, ensuring that your website is always accessible.
- Unsecured websites are vulnerable to hacking attempts, which can lead to a loss of data, downtime, and potential financial losses.
- Many countries have laws and regulations that require ecommerce sites to take security measures to protect customer data. Failure to comply with these regulations can result in legal action and penalties.
- A secure website creates a positive user experience by ensuring that visitors can access your site without fear of malware or other security risks.
- Google and other search engines prioritize secure websites, meaning that securing your site can improve your search engine ranking and drive more traffic to your site.

2. Problem Statement

The increasing popularity of e-commerce websites has also increased the risk of cyber threats and attacks, leading to compromised customer data and financial losses. The objective of this project is to identify and address the security vulnerabilities in an e-commerce website to ensure that customer data and transactions are safe and secure.

3. Motivation

The project aims to have the following features:-

- Using security tools to protect sensitive user data from unauthorized access or theft..
- Improve the overall user experience by reducing the risk of security incidents and providing peace of mind to users.
- Identify potential vulnerabilities in a website and provide as soon as possible .
- Maximum utilization of same resources to get better performance.

4. Objectives

- To Develop the sample Ecommerce website.
- To Deploy website on the AWS ec2 instance.
- To secure the deployed website.
- To manage and maintain the performance of website

5. Technology Stack

Development

- AWS
- VS code
- GitHub
- HTML

E commerce Website

- Home
- About
- Products
- Shop
- Blog
- Contact

- Search
- Login / Register
- Wishlist 5
- Basket: rs318.00 4

New Summer Shoes Collection

Competently expedite alternative benefits whereas leading-edge catalysts for change. Globally leverage existing an expanded array of leadership.

Shop Now →

- **Men Collections**

  Explore All →

- **Women Collections**

  Explore All →

- **Sports Collections**

  Explore All →

Bestsellers Products

- All
- Nike
- Adidas
- Puma
- Bata
- Apex

The website shows the product for the sale and it also show the discount in the item .

The website show the shpping feature and the secure payment feature it also show the return policies .

Compare

Men / Sports

Air Jordan 7 Retro

rs170.85 ~~rs200.21~~

- **Free Shiping**

  All orders over rs150

- **Quick Payment**

  100% secure payment

- **Free Returns**

  Money back in 30 days

- **24/7 Support**

  Get Quick Support

Nike Special

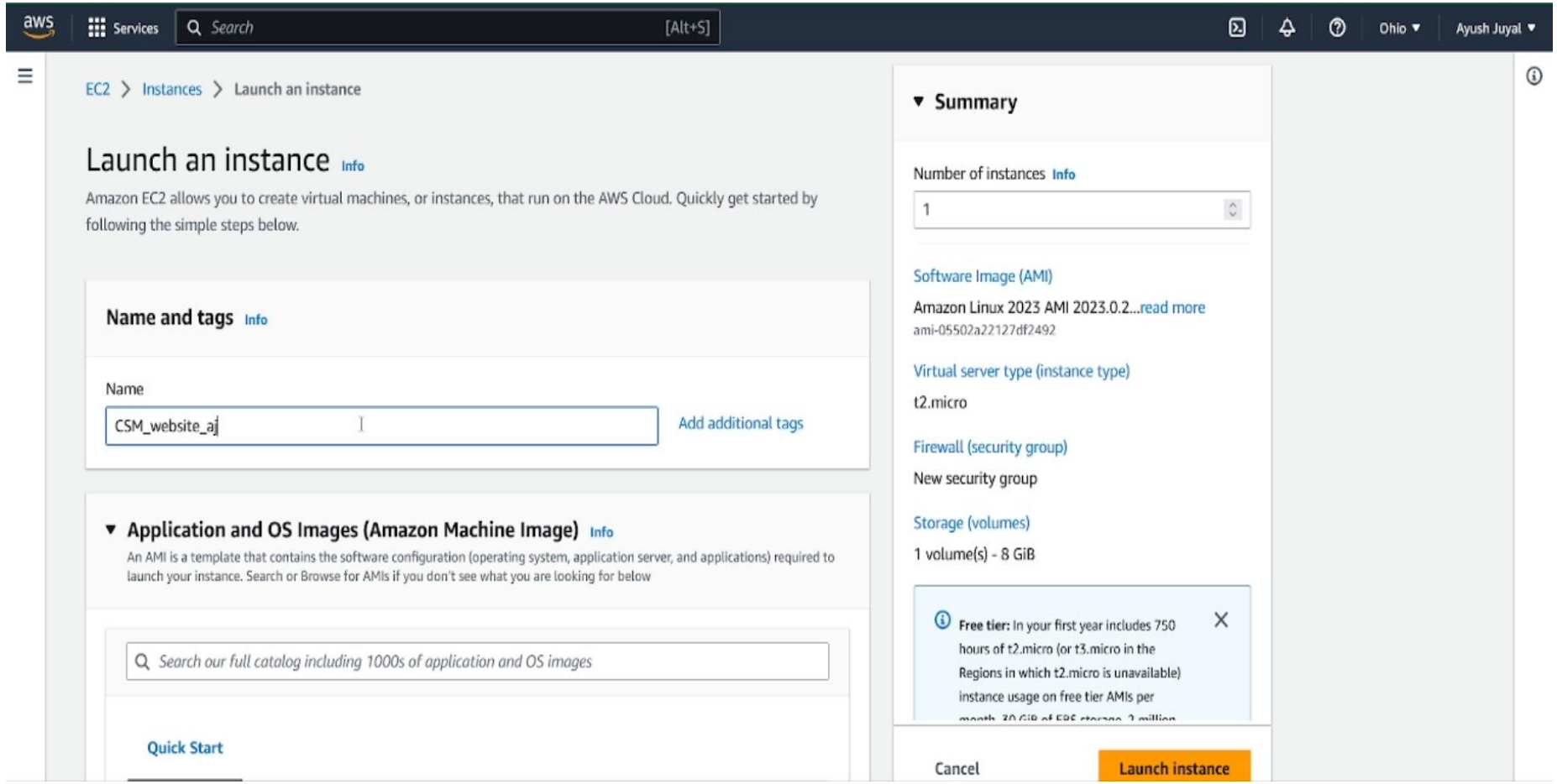
- 

  New

  - Add to Cart
  - Add to Whishlist
  - Quick View
  - Compare

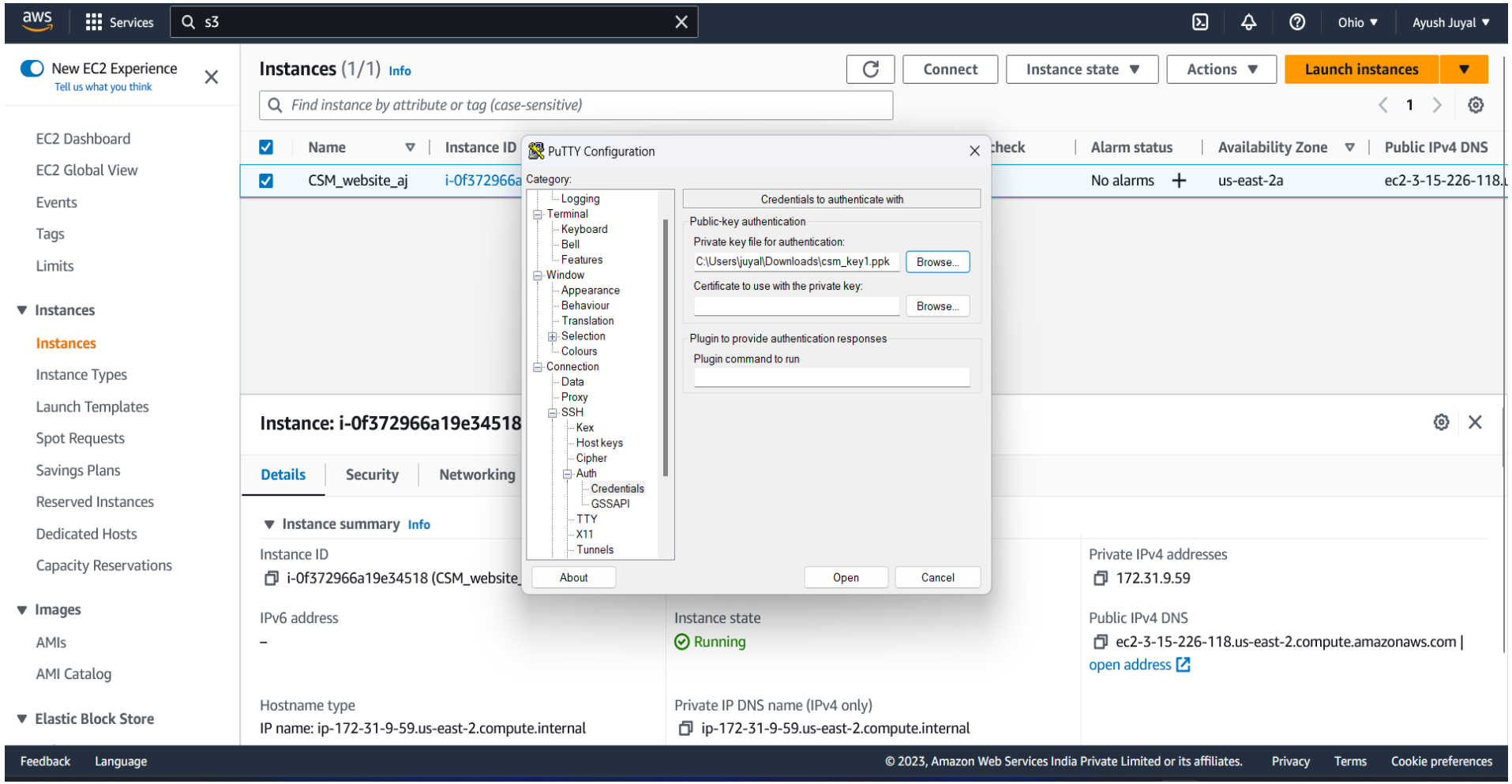
  Men / Women

Creating an instance



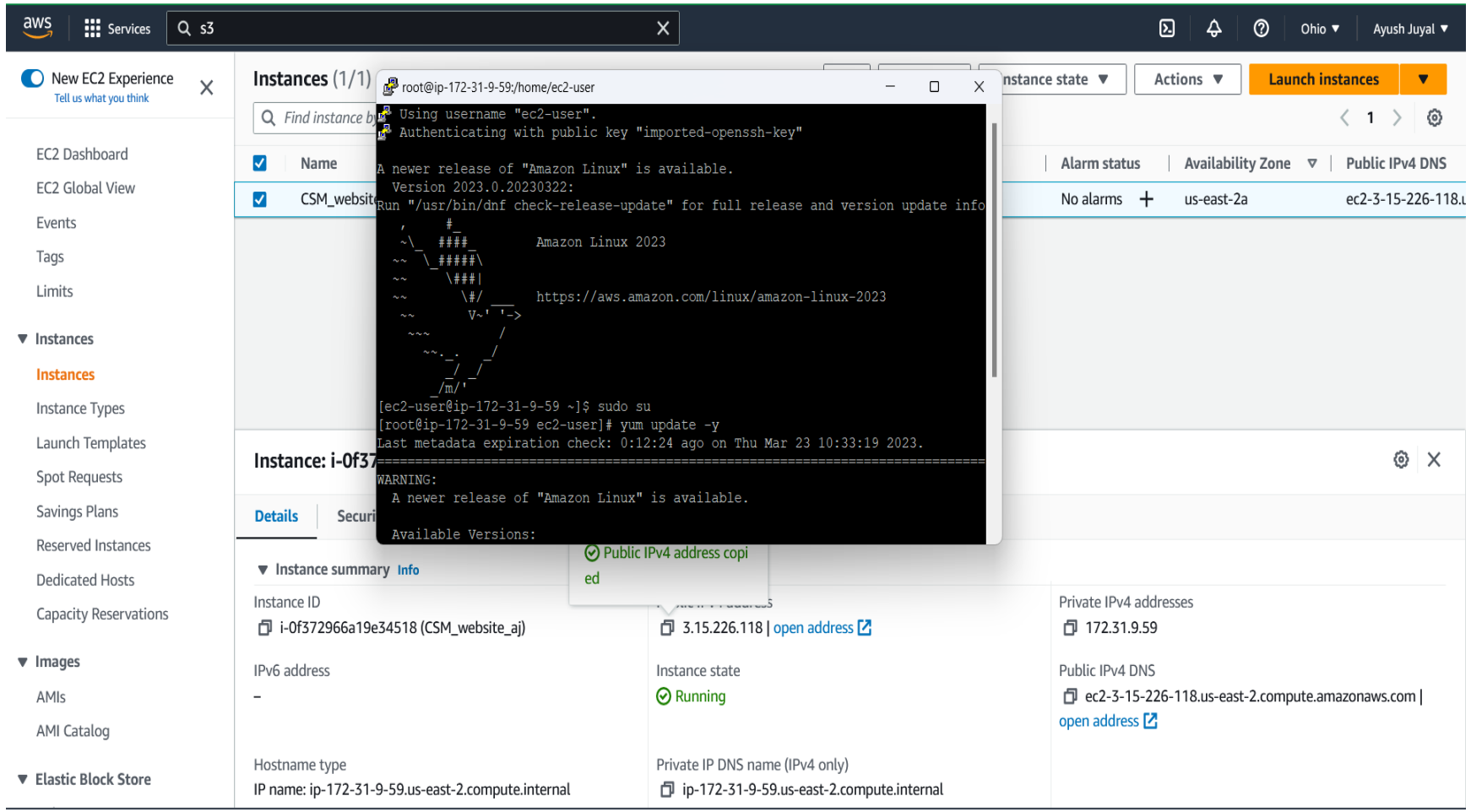
- Amazon Linux 2023 AMI
- t2.micro instance type
- 8Gb General purpose SSD
- Downloaded the keypair

Connecting to the instance



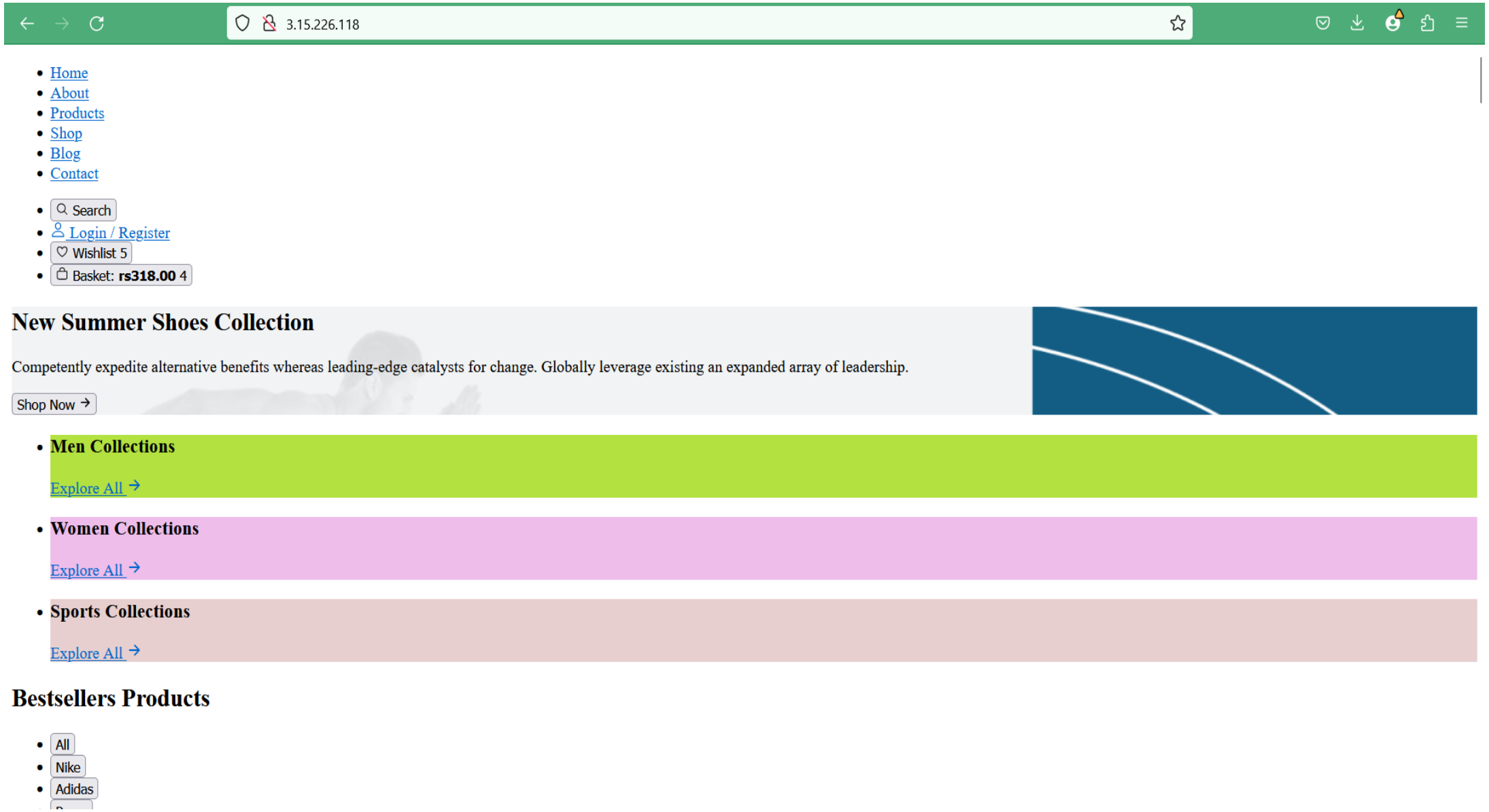
- Generated private key from puttygen
- Using that private key to connected to instance using putty

Deployment of Website on instance



- Set as root user
- Updating the package
- Installing Apache
- Setting path to locate object.
- Download files from S3
- Unzipping
- Moving contents to current directory
- Starting Apache.

- Website deployed successfully. Copying public ipv4 address of instance and searching it on browser.



Implementing MFA

- MFA stands for Multi-Factor Authentication, which is a security mechanism that requires users to provide two or more forms of authentication to access a system or application.
- In the context of AWS (Amazon Web Services), MFA refers to a feature that adds an extra layer of security to user accounts, making it harder for unauthorized users to access sensitive resources.
- In AWS, MFA requires users to provide two forms of authentication: something they know (like a password or PIN) and something they have (like a physical security token or a mobile device).
- Once MFA is enabled, users are required to provide their regular password and a unique, time-based, six-digit code generated by their MFA device, such as a smartphone or hardware token.
- MFA can be enabled for AWS accounts, IAM (Identity and Access Management) users, and for specific API actions. By adding MFA to an AWS account, users can help prevent unauthorized access to their AWS resources, reduce the risk of compromised credentials, and comply with security requirements.

aws

Multi-factor authentication

Your account is secured using multi-factor authentication (MFA). To finish signing in, turn on or view your MFA device and type the authentication code below.

Email address: juyal.aj2003@gmail.com

MFA code

099483

Submit

Troubleshoot MFA

Cancel

English

Link: https://drive.google.com/file/d/1fVNy43W0VLM4pr0yaSBnmKPQVKwH65f0/view?usp=share_link

Granting access to only one S3 Bucket

- Amazon S3 (Simple Storage Service) is a cloud-based storage service provided by Amazon Web Services (AWS). An S3 bucket is a storage container within S3 that can store any type of file, including documents, videos, images, and software code.

- S3 buckets are highly scalable and reliable, allowing users to store and retrieve large amounts of data from anywhere in the world.

- S3 buckets can be accessed using AWS Management Console, AWS CLI (Command Line Interface), or APIs (Application Programming Interfaces). They can also be configured for versioning, access control, and encryption to ensure data security and compliance with regulatory requirements.

- S3 also provides a range of storage classes with different performance and cost characteristics, allowing users to choose the right option based on their data access patterns and budget.

Creating the custom policy to restrict user to access other S3 bucket.

aws | Services | s3 | Global ▾ | Ayush Juyal ▾

Create policy

1 2 3

A policy defines the AWS permissions that you can assign to a user, group, or role. You can create and edit a policy in the visual editor and using JSON. Learn more

Visual editor | JSON | Import managed policy

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                        "s3:GetBucketLocation",
                        "s3:ListAllMyBuckets"
                    ],
            "Resource": "arn:aws:s3:::*"
        },
        {
            "Effect": "Allow",
            "Action": "s3:*",
            "Resource": [
                "arn:aws:s3:::csm-sample-bucket",
                "arn:aws:s3:::csm-sample-bucket/*"
            ]
```

Security: 0 | Errors: 0 | Warnings: 0 | Suggestions: 0

Try to access Bucket other than authorized .

aws | Services | Search [Alt+S] | Global | s3-user @ 1315-2149-5488

Amazon S3

- Buckets
- Access Points
- Object Lambda Access Points
- Multi-Region Access Points
- Batch Operations
- IAM Access Analyzer for S3

- Block Public Access settings for this account

- Storage Lens
  - Dashboards
  - AWS Organizations settings

- Feature spotlight 3

- AWS Marketplace for S3

Amazon S3 > Buckets > csm001-project-bucket

csm001-project-bucket Info

Objects | Properties | Permissions | Metrics | Management | Access Points

Objects

Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

Copy S3 URI | Copy URL | Download | Open | Delete | Actions | Create folder | Upload

Find objects by prefix | Show versions | 1

Name	Type	Last modified	Size	Storage class

Insufficient permissions to list objects

After you or your AWS administrator have updated your permissions to allow the s3:ListBucket action, refresh the page. Learn more about Identity and access management in Amazon S3

Feedback | Language | © 2023, Amazon Web Services India Private Limited or its affiliates. | Privacy | Terms | Cookie preferences

https://drive.google.com/file/d/1Jug3wqGcYUQGfHLjRTeFqMEMaYwKqKQB/view?usp=share_link

Advantages:

1. Improved Security: By limiting access to a single S3 bucket, you can reduce the risk of data breaches or unauthorized access to sensitive information. This makes it easier to control who has access to your data, and minimizes the potential for data leakage or other security issues.

2. Easier Management: Managing permissions for multiple S3 buckets can be time-consuming and complex. By limiting access to just one bucket, you can simplify the process of managing permissions and reduce the risk of errors or misconfigurations.

3. Cost Savings: AWS charges for storage, data transfer, and other S3-related services based on usage. By limiting access to a single bucket, you can reduce the amount of data stored and transferred, potentially saving money on your AWS bill.

4. Better Performance: Accessing multiple S3 buckets can be slower than accessing a single bucket. By limiting access to just one bucket, you can improve performance and reduce latency for your applications and services.

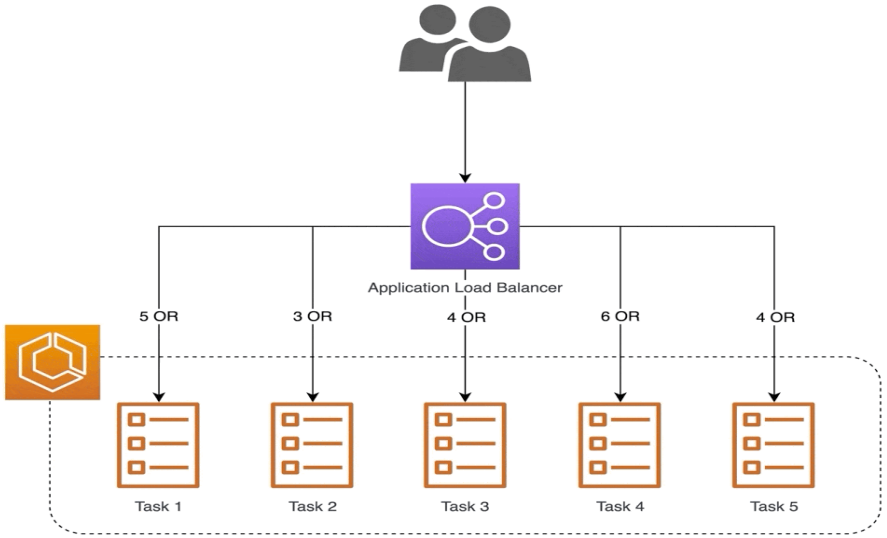
5. Simplified Access Management: Limiting access to a single S3 bucket can simplify access management for third-party users or services. For example, if you are providing access to an external developer or contractor, limiting access to just one bucket can make it easier to manage their permissions and reduce the risk of accidental access to other data or services.

Implementing Application load Balancer

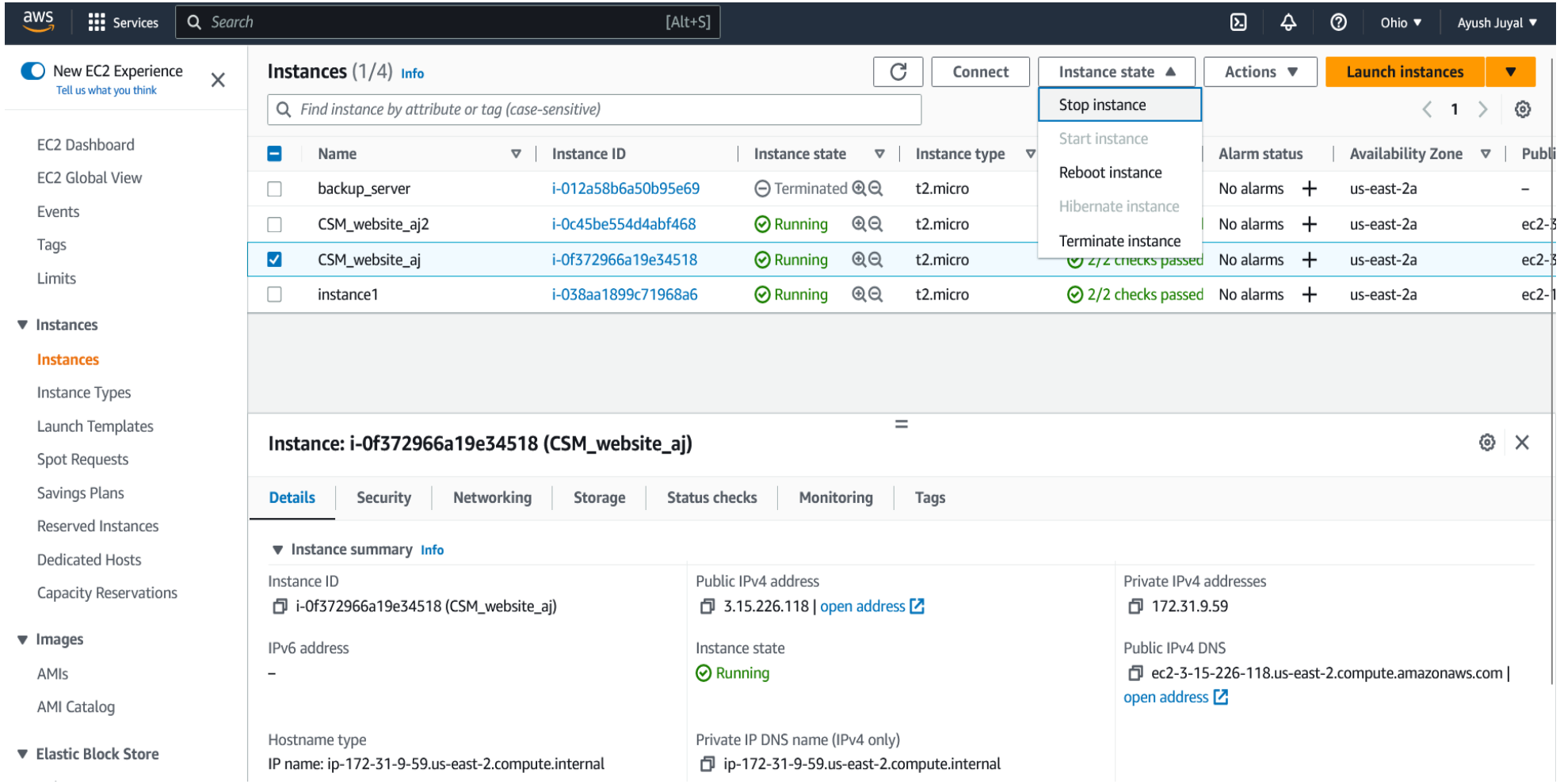
- An Application Load Balancer (ALB) is a service provided by Amazon Web Services (AWS) that distributes incoming traffic to multiple targets, such as EC2 instances, containers, and IP addresses, in one or more Availability Zones.

- It is designed to optimize application availability and scalability by evenly distributing traffic to healthy targets, monitoring the health of targets, and automatically routing traffic away from unhealthy targets.

- The ALB is managed through the AWS Management Console, CLI, or SDKs and can be integrated with other AWS services such as Elastic Compute Cloud (EC2), Elastic Container Service (ECS), Elastic Kubernetes Service (EKS), and Auto Scaling.



Stopping the initial instance on which website was hosted.



- Wait for the load balancer health check to pass.
- After that lets try weather load balancer working or not.
- Stop the initial instance on which the website was hosted .

- Though the instance has been stopped then also the website is perfectly working.
- This is because now the load balancer had transferred all the request to the second instance

Link: https://drive.google.com/file/d/1AwvtmGOsxf8vGBmB7Mp-yI4f1JCaBn7m/view?usp=share_link

Advantages:

- Improved Availability: An ALB can distribute traffic across multiple instances, which improves availability of the application. If one instance fails, traffic is automatically rerouted to another instance, helping to ensure that the application remains available.

- Increased Scalability: ALBs are designed to handle a large number of concurrent connections and can scale horizontally by adding additional instances to the load balancer target group. This allows the application to handle increased traffic without any interruption in service.

- Enhanced Security: ALBs provide SSL/TLS termination, which means that they can decrypt incoming traffic and then re-encrypt it before sending it to the target instances. This helps to secure the application and prevent attacks like SSL stripping.

- Simplified Network Configuration: ALBs can be used to route traffic to instances in multiple availability zones, which simplifies network configuration and improves the application's resiliency.

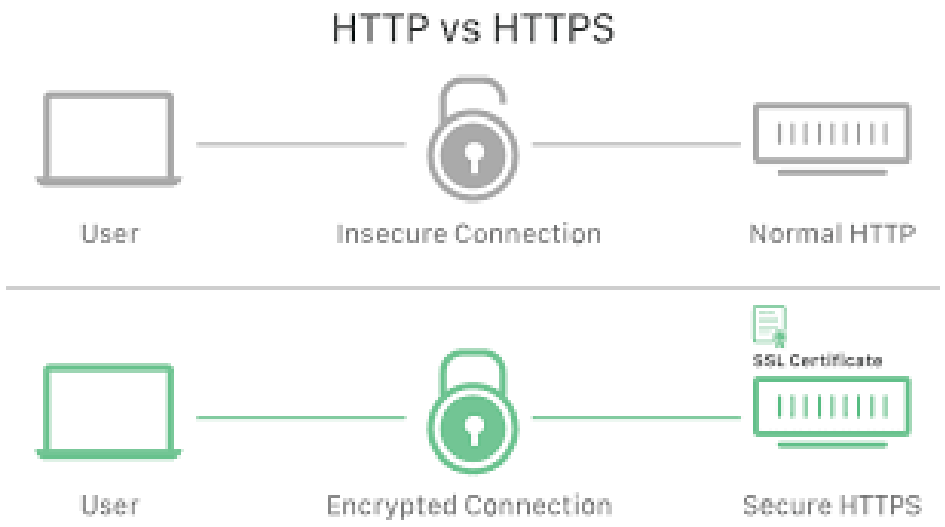
- Integration with AWS Services: ALBs can integrate with other AWS services such as Auto Scaling, Amazon ECS, and AWS Lambda, which can help to automate scaling and simplify application deployment.

Securing Site using ACM

- AWS Certificate Manager (ACM) is a service provided by Amazon Web Services (AWS) that enables users to manage SSL/TLS certificates for their web applications and websites deployed on AWS.

- ACM simplifies the process of obtaining, deploying, and managing SSL/TLS certificates for websites and web applications running on AWS services

- With ACM, users can easily request and manage SSL/TLS certificates, automate the renewal of certificates, and integrate with other AWS services to enhance the security of their applications.



3.15.226.118

Site information for 3.15.226.118

Connection not secure

- Home
- About
- Products
- Shop
- Blog
- Contact

- Search
- Login / Register
- Wishlist 5
- Basket: **rs318.00** 4

New Summer Shoes Collection

Competently expedite alternative benefits whereas leading-edge catalysts for change. Globally leverage existing an expanded array of leadership.

Shop Now →

- **Men Collections**

  Explore All →

- **Women Collections**

  Explore All →

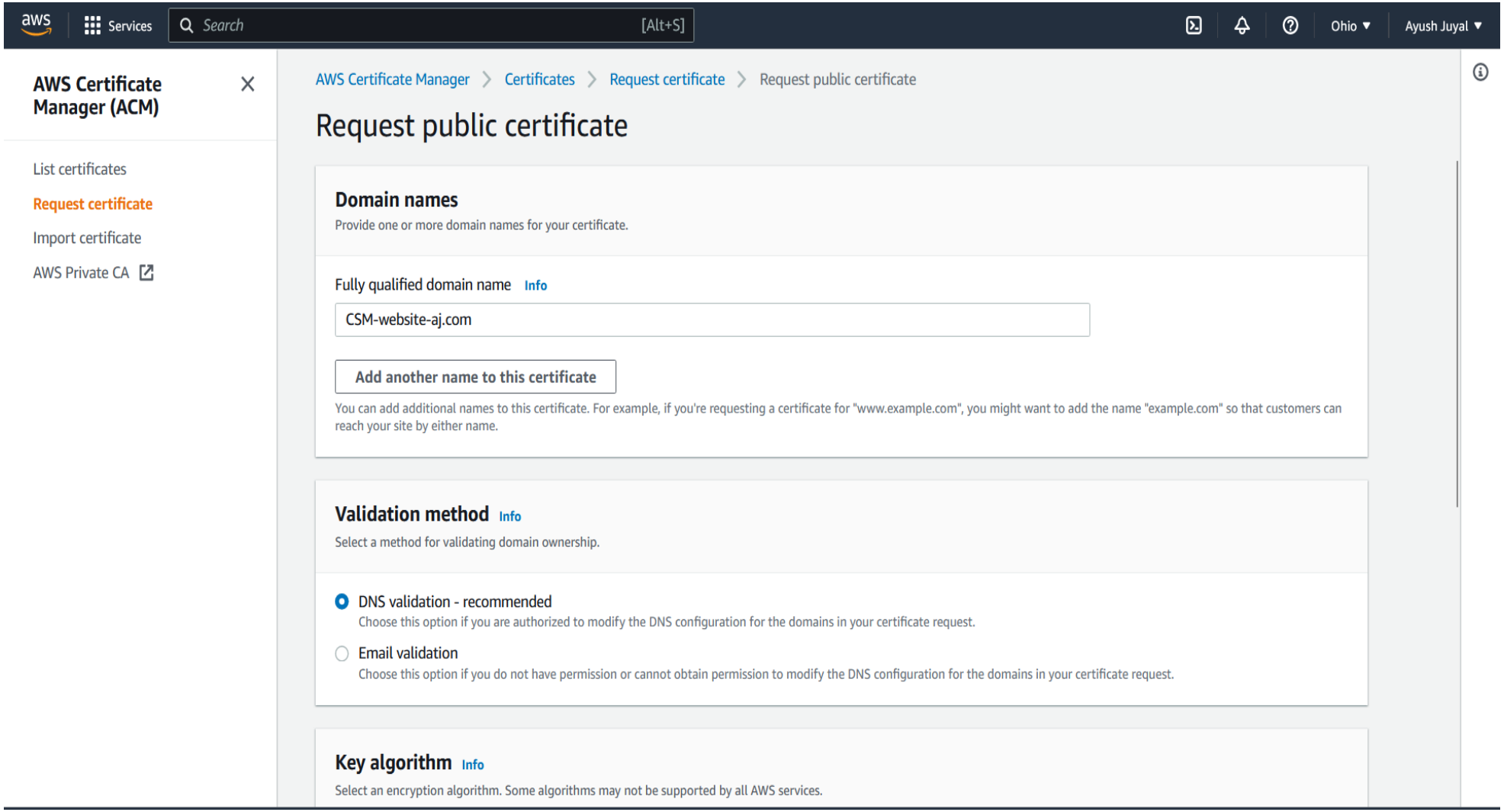
- **Sports Collections**

  Explore All →

Bestsellers Products

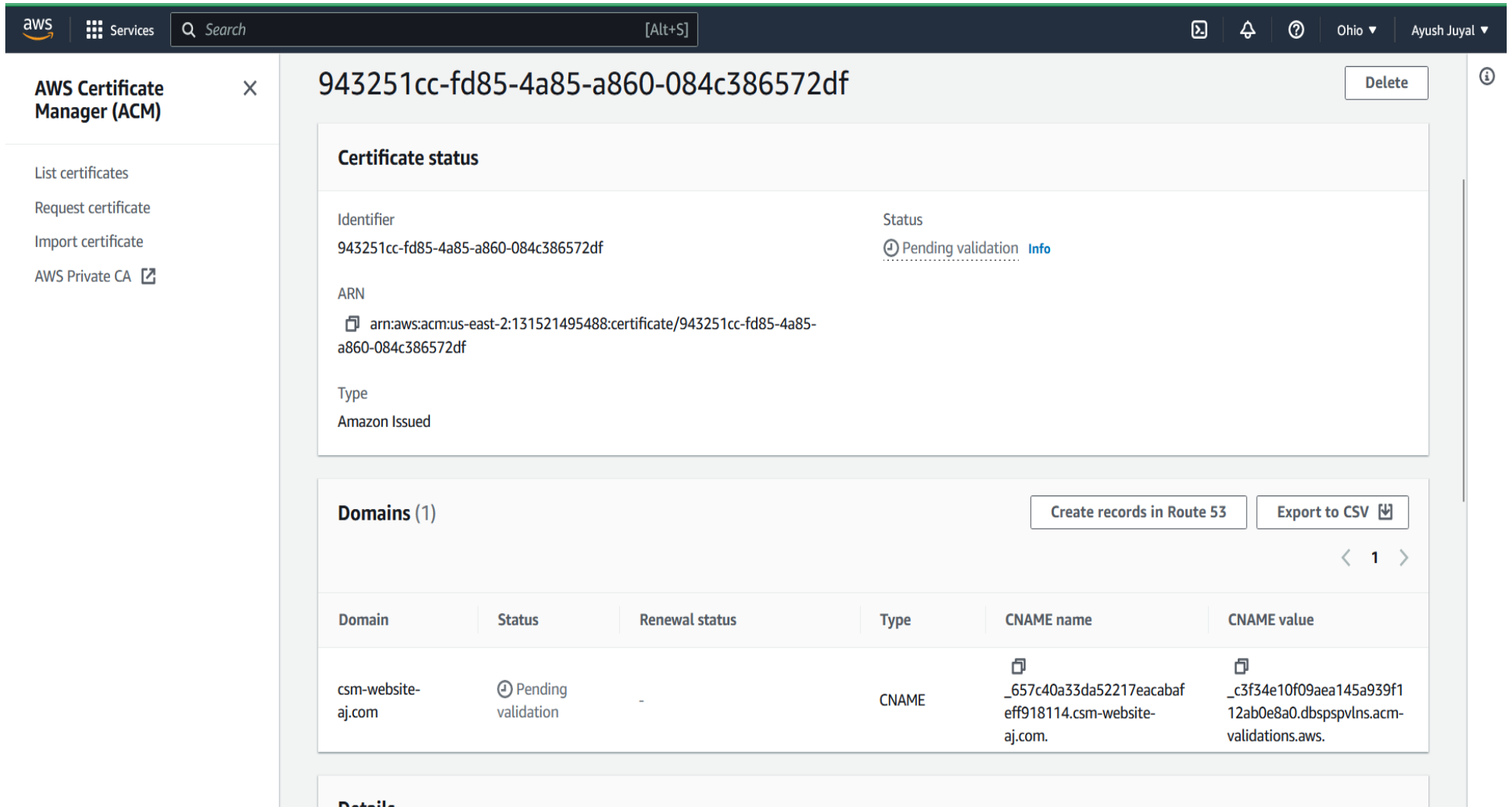
- All
- Nike
- Adidas

1. Request for the certificate



- Go to the AWS certificate manager.
- Click on the request a certificate
- Provide the domain name.
- Also provide the mode of validate either DNS or the email verification.

2. Validating using the DNS validation



- Go to the site where you have registered you deployed site.
- There click on manage DNA and this will divert to the admin area.
- There add the cname provided by the aws as well as cname value.
- Then click on route53 and click on create.
- Will take about 30 min to validate.

https://drive.google.com/file/d/149eoLt16au0pp0agneVlHXiWaL_W6kqa/view?usp=share_link

Advantage:

- Automated certificate renewal: AWS Certificate Manager can automate the renewal and installation of SSL/TLS certificates, eliminating the need for manual intervention and reducing the risk of certificate expiry.

- Simplified certificate management: With ACM, you can manage all your SSL/TLS certificates in a centralized location, making it easier to keep track of certificates, reduce errors and improve security.

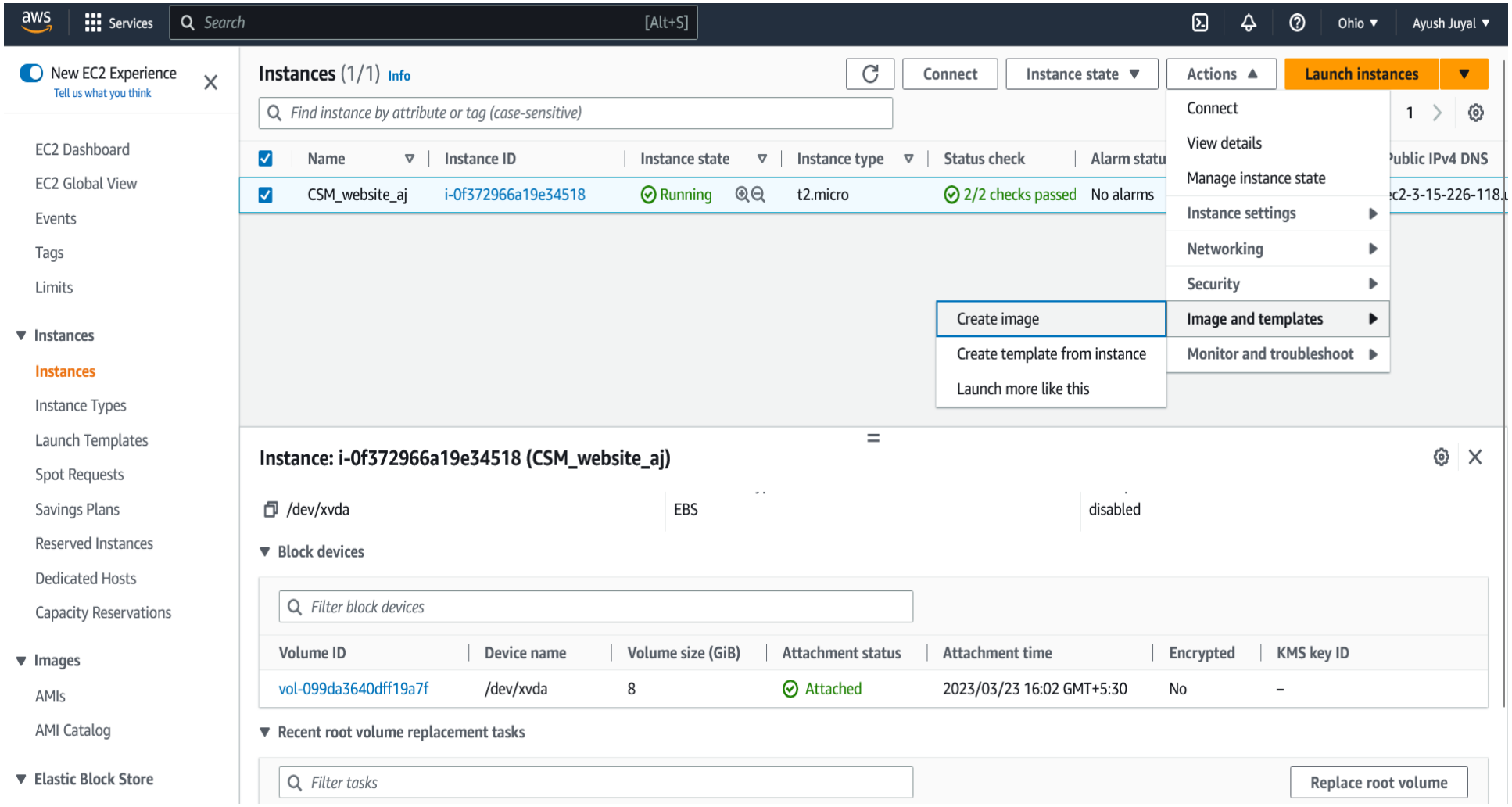
- Free SSL/TLS certificates: AWS Certificate Manager provides free SSL/TLS certificates for use with AWS services like Elastic Load Balancer, CloudFront, and API Gateway, eliminating the need for additional costs.

- Enhanced security: ACM provides a highly secure and reliable way to manage SSL/TLS certificates, with features such as certificate transparency monitoring, certificate revocation, and certificate private key protection.

Create Backup of the website Using AMI

- Amazon Machine Image (AMI) is a pre-configured virtual machine image used to create an Amazon Elastic Compute Cloud (Amazon EC2) instance.
- An AMI contains all the necessary information to launch an EC2 instance, including the operating system, application server, and any additional software needed to run an application.
- An AMI can be thought of as a snapshot of an EC2 instance at a particular point in time. It can be customized with different configurations, software, and settings, then saved and shared with others.
- This makes it easy to replicate an environment and deploy it quickly, without having to go through the entire setup process every time.

1. Select the instance and create Image.



- Select the image need to backup.
- Go to the actions then on images and template and click on create images.
- The EBS volume attached to the instance will also backed up.
- Can see under backup section.
- Make sure to enable no reboot.

aws | Services | Search [Alt+S] | Ohio | Ayush Juyal

New EC2 Experience
Tell us what you think

- EC2 Dashboard
- EC2 Global View
- Events
- Tags
- Limits
- **Instances**
  - Instances
  - Instance Types
  - Launch Templates
  - Spot Requests
  - Savings Plans
  - Reserved Instances
  - Dedicated Hosts
  - Capacity Reservations
- **Images**
  - AMIs
  - AMI Catalog
- **Elastic Block Store**

Instances (1/2) Info

Connect | Instance state | Actions | Launch instances

Find instance by attribute or tag (case-sensitive)

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
backup_server	i-012a58b6a50b95e69	Running	t2.micro	2/2 checks passed	No alarms	us-east-2a	ec2-18-118-139-77
CSM_website_aj	i-0f372966a19e34518	Running	t2.micro	2/2 checks passed	No alarms	us-east-2a	ec2-3-15-226-118.

Instance: i-012a58b6a50b95e69 (backup_server)

Details | Security | Networking | Storage | Status checks | Monitoring | Tags

Instance summary Info

Instance ID
i-012a58b6a50b95e69 (backup_server)

IPv6address
–

Hostname type
IP name: ip-172-31-6-199.us-east-2.compute.internal

Public IPv4 address
18.118.139.77 | open address

Instance state
Running

Private IP DNS name (IPv4 only)
ip-172-31-6-199.us-east-2.compute.internal

Private IPv4 addresses
172.31.6.199

Public IPv4 DNS
ec2-18-118-139-77.us-east-2.compute.amazonaws.com | open address

Link: https://drive.google.com/file/d/1Uj21WTPB2ZCl-eIyR52pPp3AasMMBNCr/view?usp=share_link

Advantage:

- Time-saving: AMIs provide a quick and easy way to launch pre-configured instances, saving time and effort in setting up and configuring new instances.
- Consistency: Using AMIs ensures consistency in the configuration and setup of instances, which reduces the risk of errors and improves the reliability of the application.
- Flexibility: AMIs can be customized to meet specific requirements, allowing for greater flexibility and adaptability in managing instances.
- Scalability: AMIs can be used to create identical instances, making it easy to scale up or down the number of instances as needed to meet changing demand.
- Security: AMIs can be configured to include security settings, such as firewalls and encryption, to help protect the instance and the application running on it.

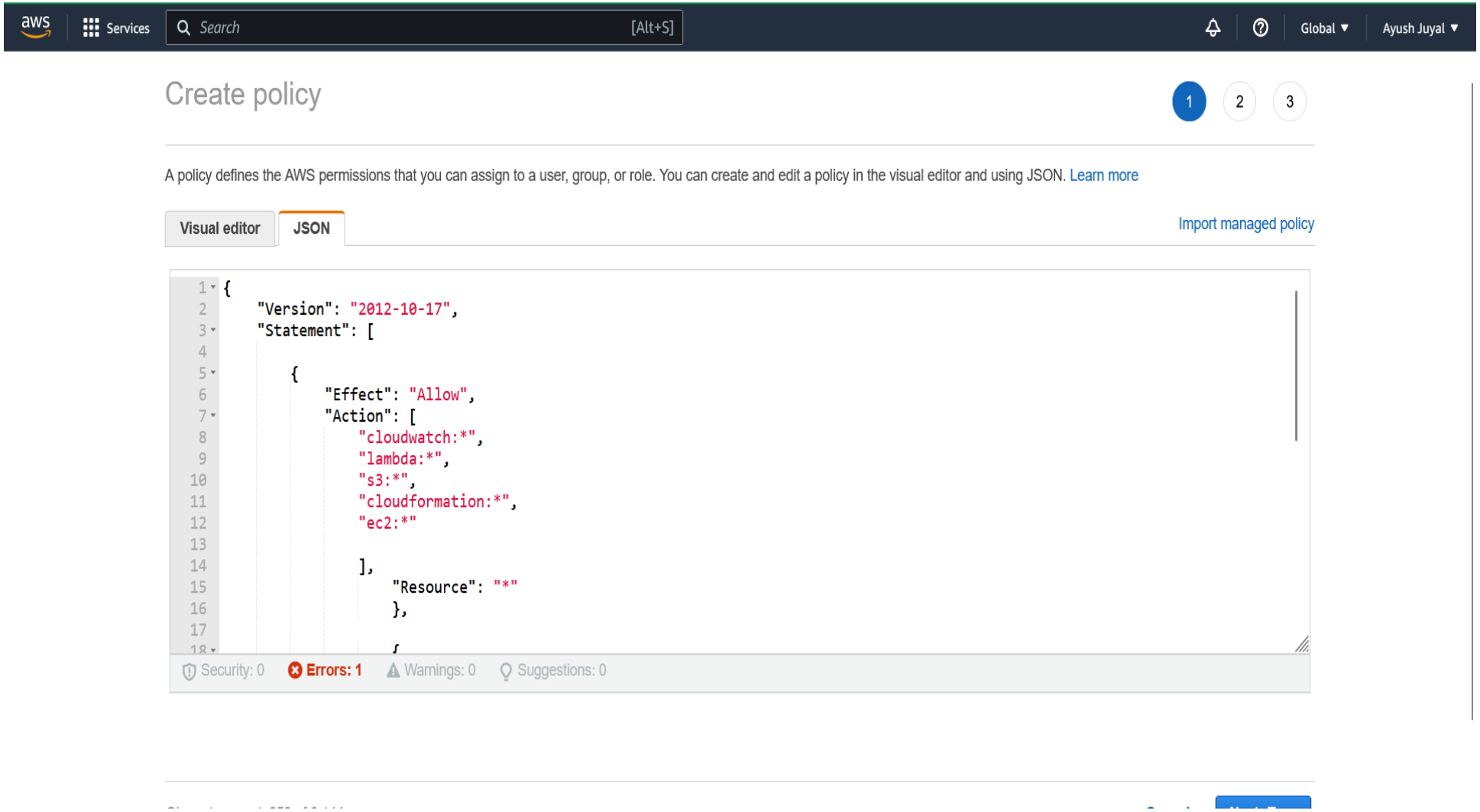
Implemented Limit Allowed EC2 Instance with IAM policy

1. Cost control: Limiting the user's ability to create specific EC2 instances can help control costs by preventing users from launching expensive instances unnecessarily. By restricting the types of instances that users can launch, you can ensure that they only use the resources that are necessary for their workloads, and avoid wasting resources on unnecessary or oversized instances.

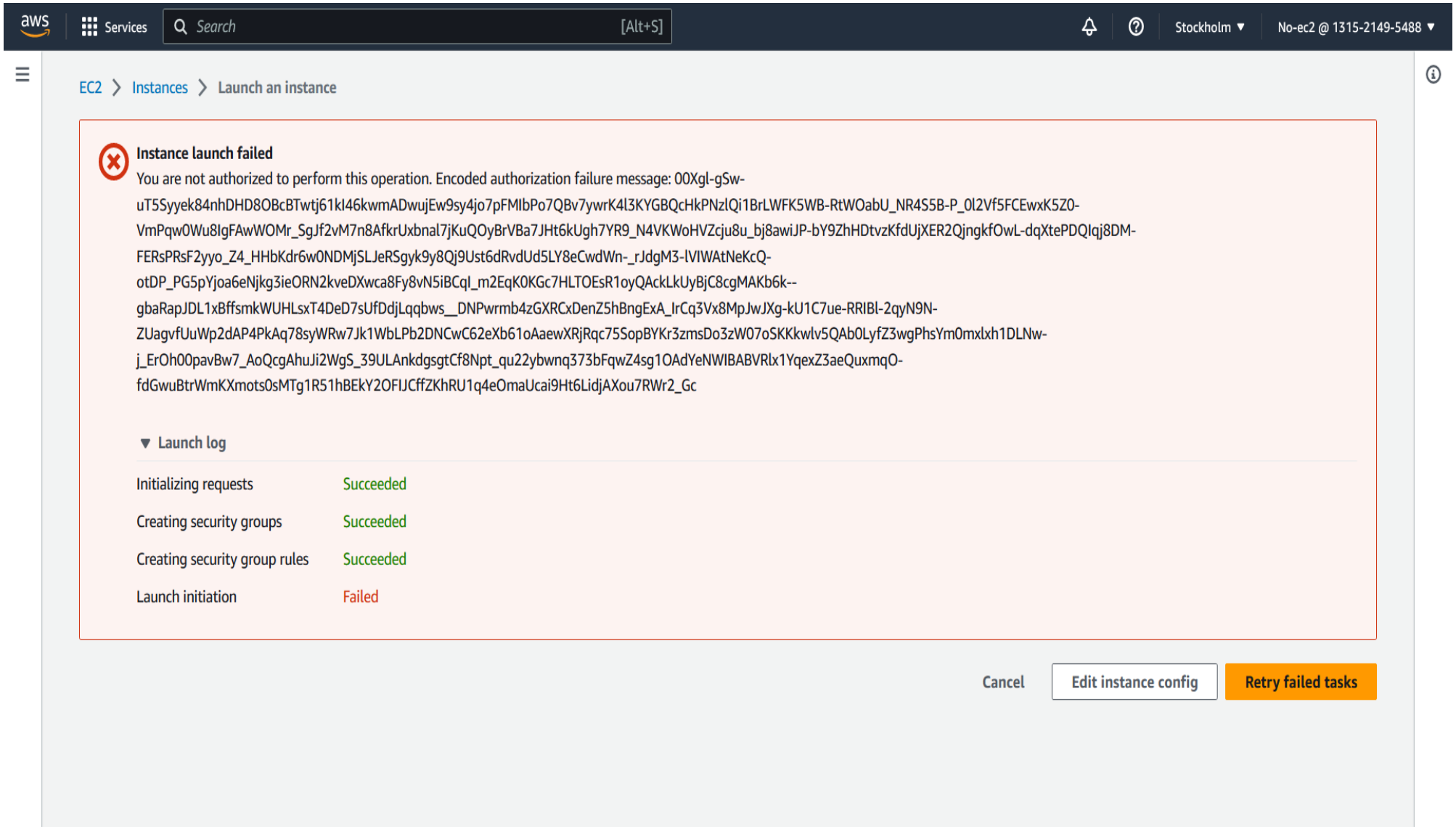
2. Security: Limiting the user's ability to create specific EC2 instances can also help enhance security by preventing users from launching instances with excessive privileges or permissions. By controlling the types of instances that users can launch, you can ensure that they only have access to the resources that they need to perform their tasks, and avoid exposing sensitive data or resources to unauthorized users.

3. Consistency: Limiting the user's ability to create specific EC2 instances can help ensure consistency and standardization in your environment. By controlling the types of instances that users can launch, you can ensure that all instances are built to your organization's standards, with appropriate security controls, networking configurations, and other settings.

1. Create Policy



- Go to the IAM dashboard.
- Click on the policy button.
- Create policy To limit user to create particular instance Type.



- If the user try to launch any of the instance other than which we had allowed then will get the following error.

https://drive.google.com/file/d/1dfVGtIbUk0e2I4QR2WxzBMgbF_F7zTIw/view?usp=share_link

Implementing AWS Code commit Service.

- AWS CodeCommit is a fully-managed source control service that makes it easy for companies to host private Git repositories. Some of the advantages of AWS CodeCommit include:

1.Secure: AWS CodeCommit uses encryption to secure your code, and it integrates with AWS Identity and Access Management (IAM) to allow you to manage access to your repositories.

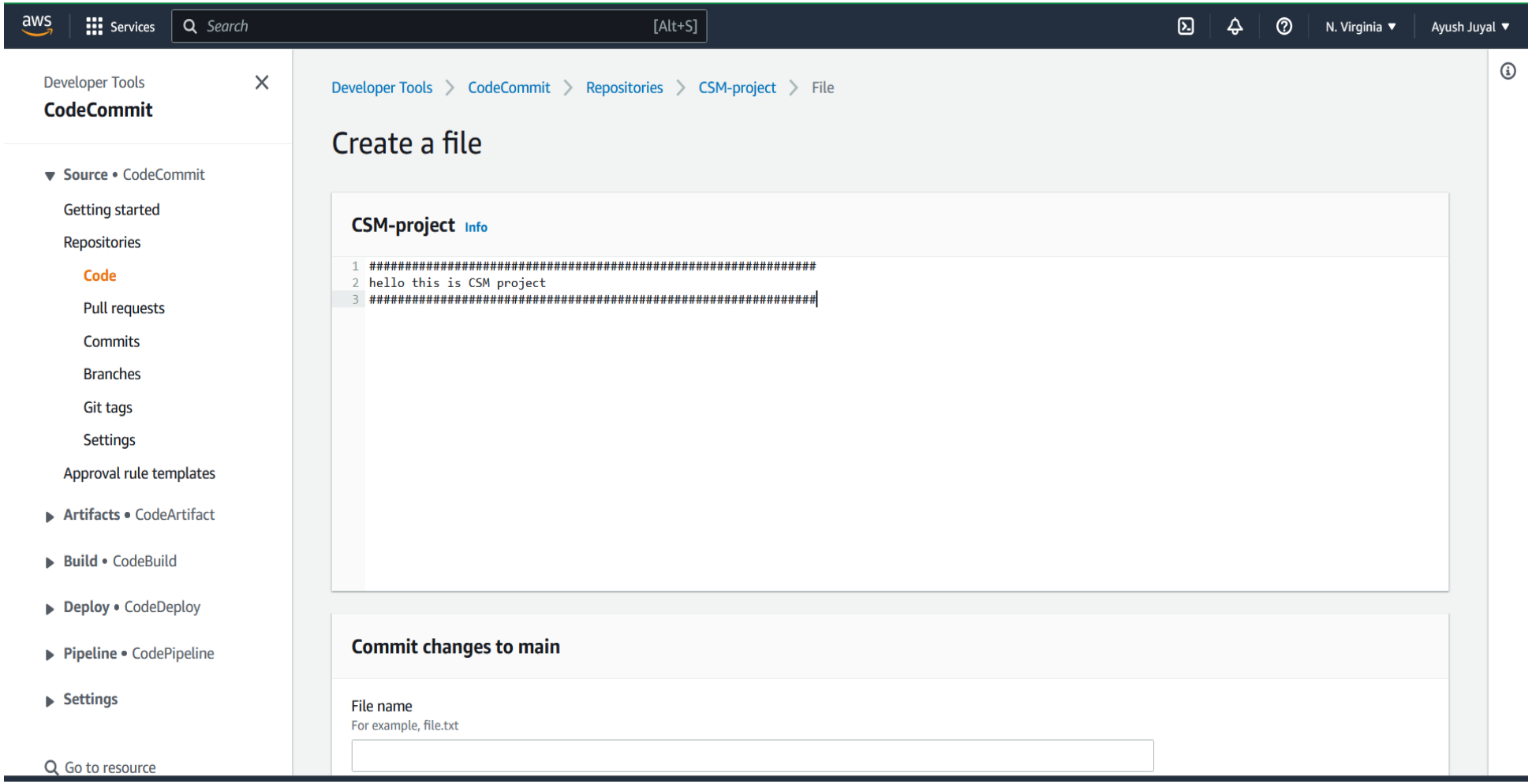
2.Scalable: AWS CodeCommit is designed to handle repositories of any size, and it can scale to meet your needs as your codebase grows.

3.Easy to use: AWS CodeCommit integrates with many popular development tools, including Git, the AWS CLI, and various IDEs.

4.Flexible: AWS CodeCommit supports both HTTPS and SSH protocols for repository access, and it can integrate with other AWS services like AWS CodeBuild and AWS CodePipeline.

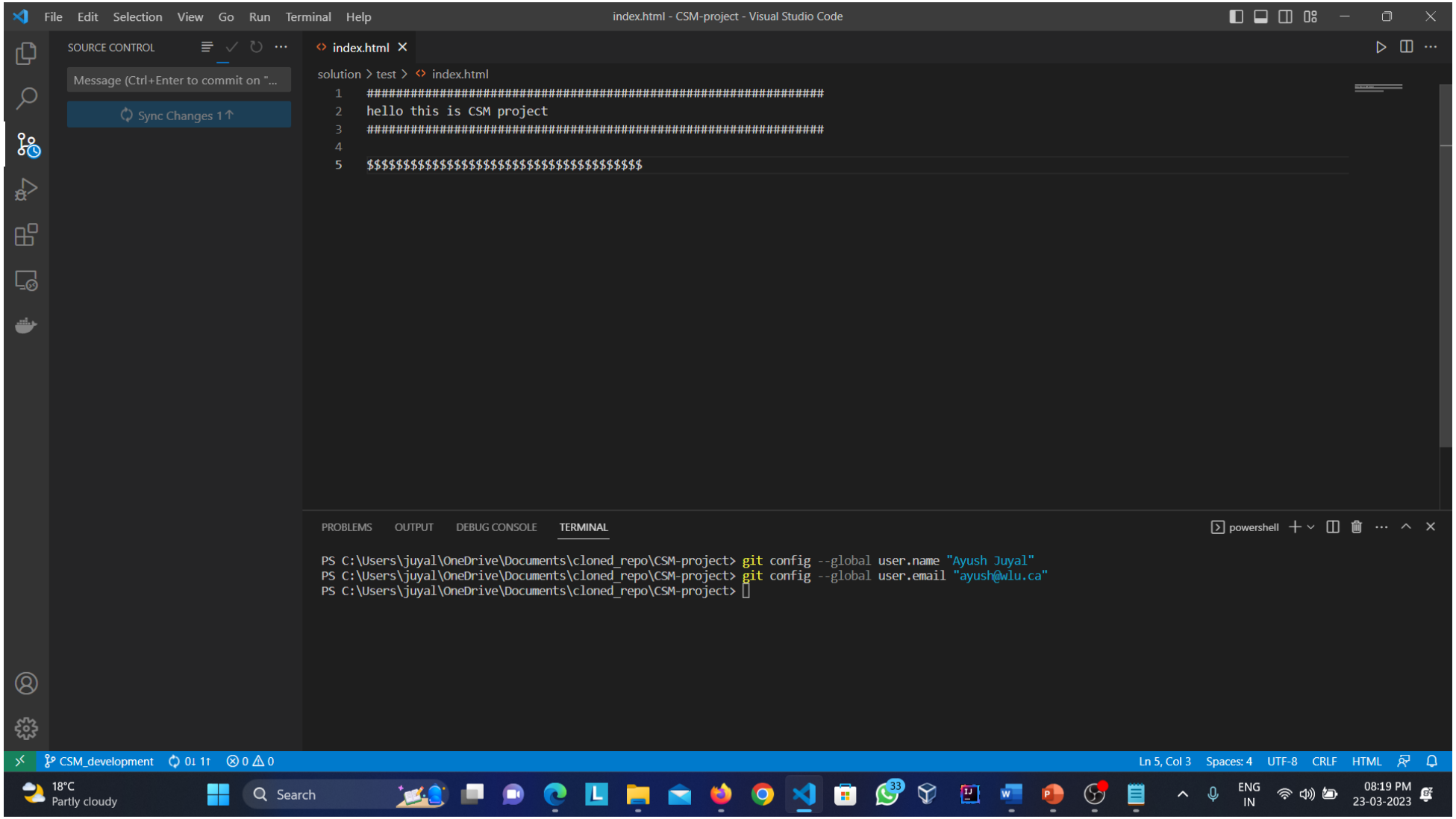
5.Cost-effective: AWS CodeCommit charges based on the size of your repositories and the number of users accessing them, making it a cost-effective option for hosting private Git repositories.

Create Repository For the Development purpose



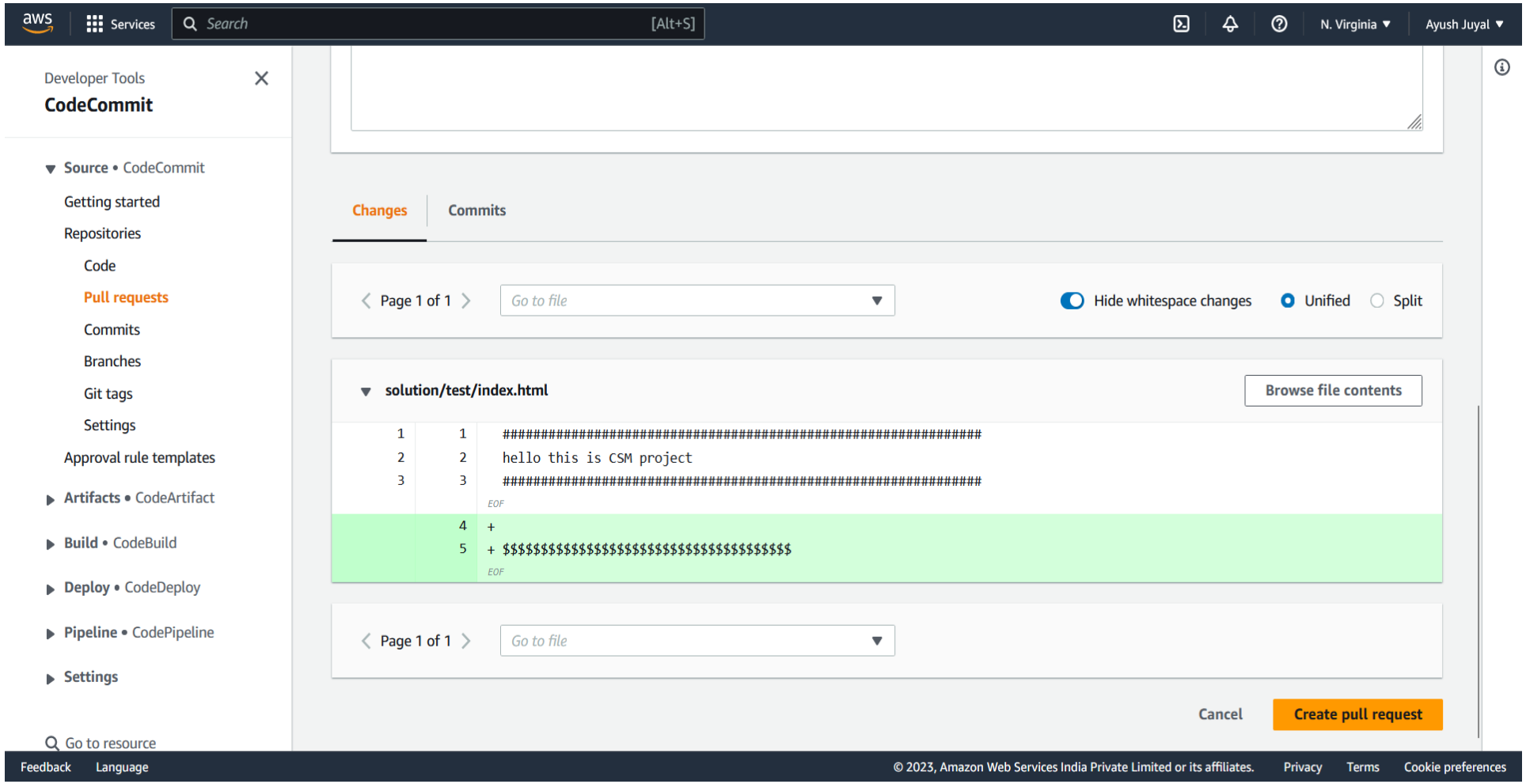
- Create the repository.
- In that repository create the sample file and commit the changes to it.
- This is done under main brand so create another branch where developer could upload their code.
- Then generate the http git credentials for the user and save it.

Cloning and Modifying the repository from the any IDE



- Then open any ide here used vs code.
- Click on cone button put the url by that you get from the clone https button of the repository.
- After entering that it will ask the username and password put both of them and the clone will start.
- The file will be added to workspace and modify It and then commit it.

See Changes in the repository



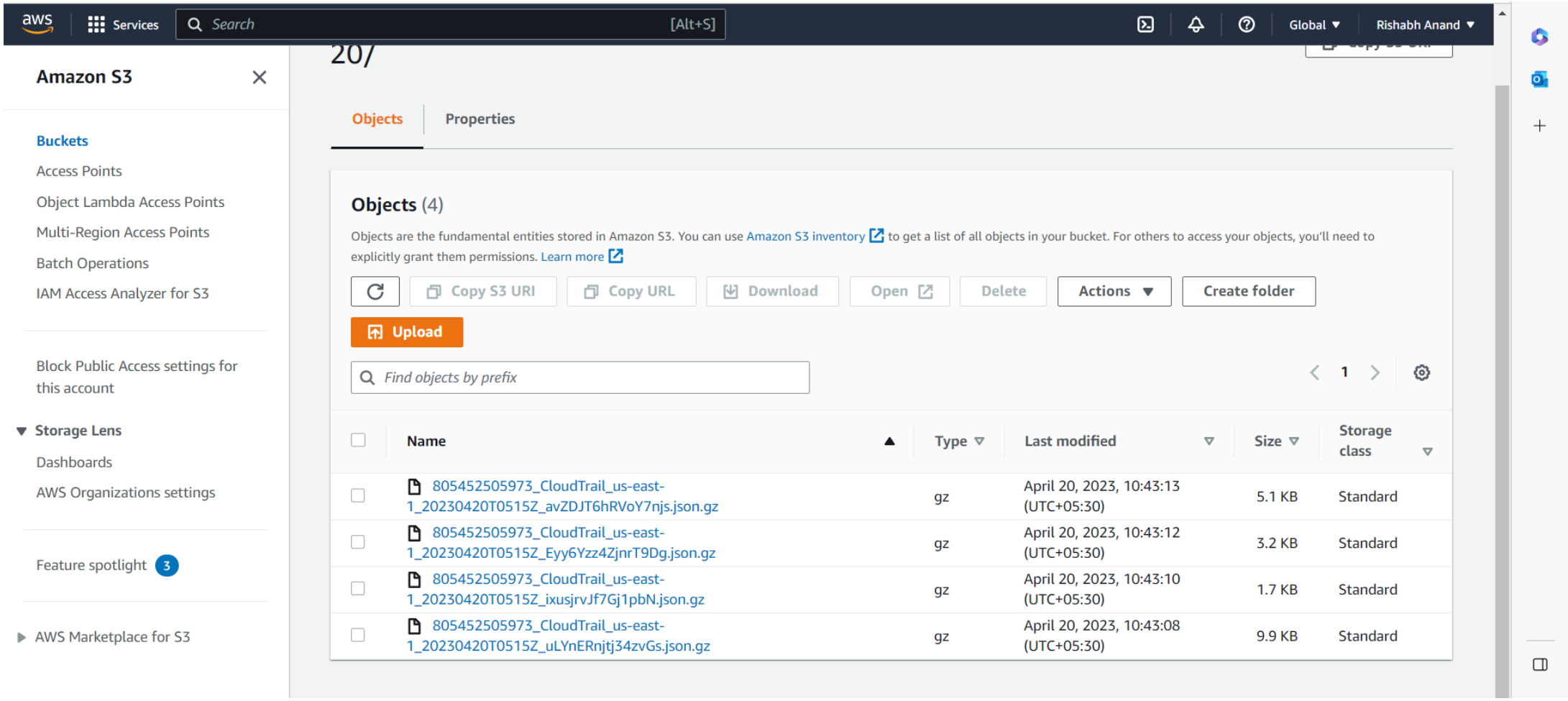
- When will open the main branch you will see there is no change but when you open the second branch will see change.
- Hence change has been done.
- Can also compare the main and user created branch and can commit changes to the main branch.

https://drive.google.com/file/d/10BrGvYiANopTEySYMkQromjub3zhdv9B/view?usp=share_link

Implementing AWS Cloud Trail

- CloudTrail is an AWS service that provides a record of all API calls made within an AWS account. It captures all API calls made through the AWS Management Console, SDKs, and command-line tools, as well as actions taken through other AWS services.
- CloudTrail logs are stored in an S3 bucket. The logs provide detailed information about who made the API call, which resources were involved, and what actions were taken. This makes it easier to troubleshoot issues, track changes, and ensure compliance with regulatory requirements.

Amazon
CloudTrail



Link: https://drive.google.com/file/d/1lFAYwCYIsi-0zJSRz_MmYMOvLrCkTxbh/view?usp=share_link

Advantage:

1.Visibility: CloudTrail provides visibility into all API calls made within an AWS account, including who made the call, which resources were involved, and what actions were taken. This makes it easier to troubleshoot issues, track changes, and ensure compliance with regulatory requirements.

2.Security: CloudTrail can be used to monitor activity in real-time, set up alerts for specific events, and automate responses to security threats. This helps to identify and respond to potential security risks and protect against unauthorized access.

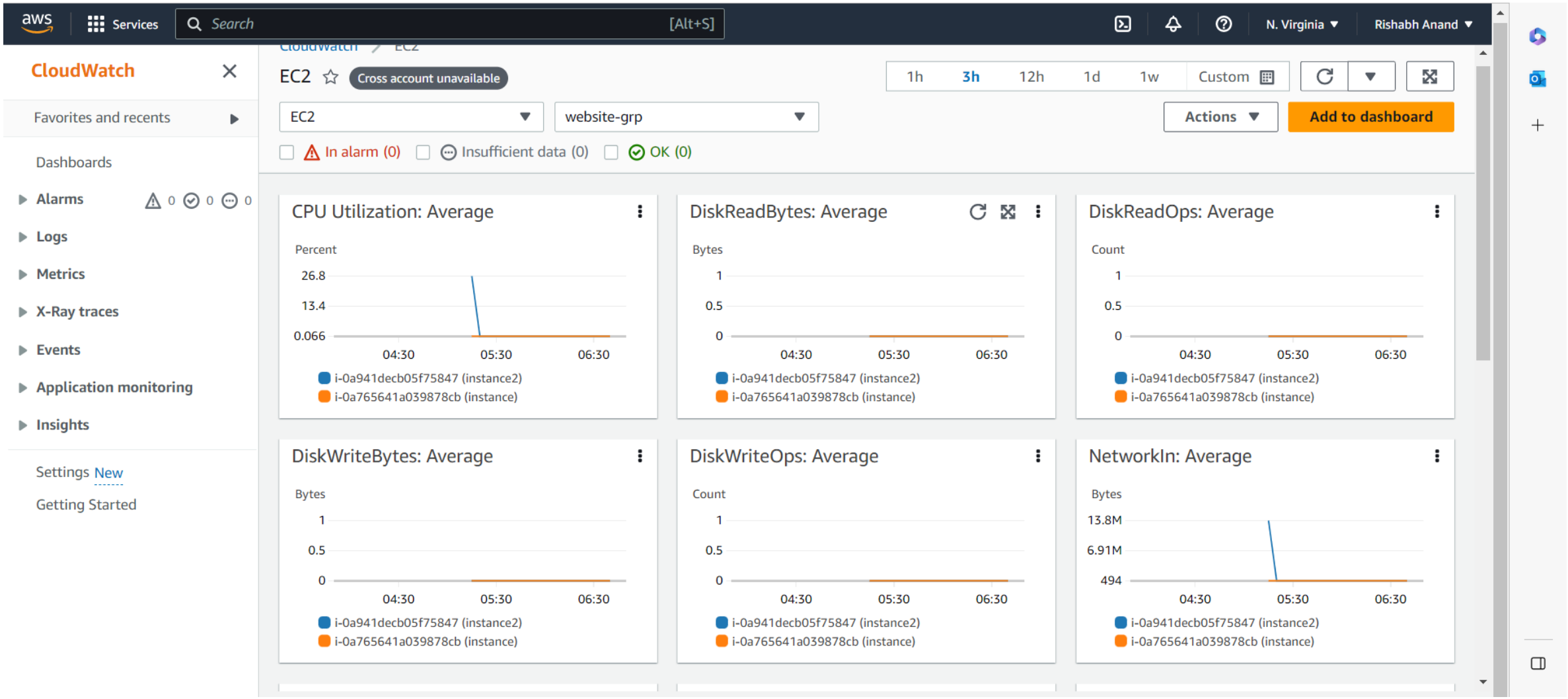
3.Compliance: CloudTrail logs can be used to provide detailed auditing and compliance reports, helping to meet regulatory requirements and ensure that the organization is adhering to industry best practices.

4.Troubleshooting: CloudTrail logs can be used to diagnose issues and troubleshoot problems within an AWS environment, providing valuable insights into usage patterns and identifying potential areas for improvement.

Implementing AWS Cloud Watch

- Amazon CloudWatch is a monitoring and management service provided by AWS that enables users to monitor resources and applications running on AWS in real-time. CloudWatch can be used to collect and track metrics, collect and monitor log files, and set alarms.
- CloudWatch can monitor AWS resources such as EC2 instances, RDS databases, and S3 buckets, as well as custom metrics generated by applications and services. It provides detailed insights into system-wide performance, resource utilization, and application-level performance.

AWS CloudWatch



Link: https://drive.google.com/file/d/1FuJYISKWmYiGMKaEOu0q4NL56iEbPpEG/view?usp=share_link

Advantage:

1.Real-time Monitoring: CloudWatch provides real-time monitoring of AWS resources, including EC2 instances, RDS databases, and S3 buckets, as well as custom metrics generated by applications and services.

2.Scalability: CloudWatch can scale to meet the needs of any AWS environment, from small-scale applications to large enterprise workloads.

3.Cost-Effective: CloudWatch is a cost-effective way to monitor AWS resources, with pay-as-you-go pricing and no upfront costs.

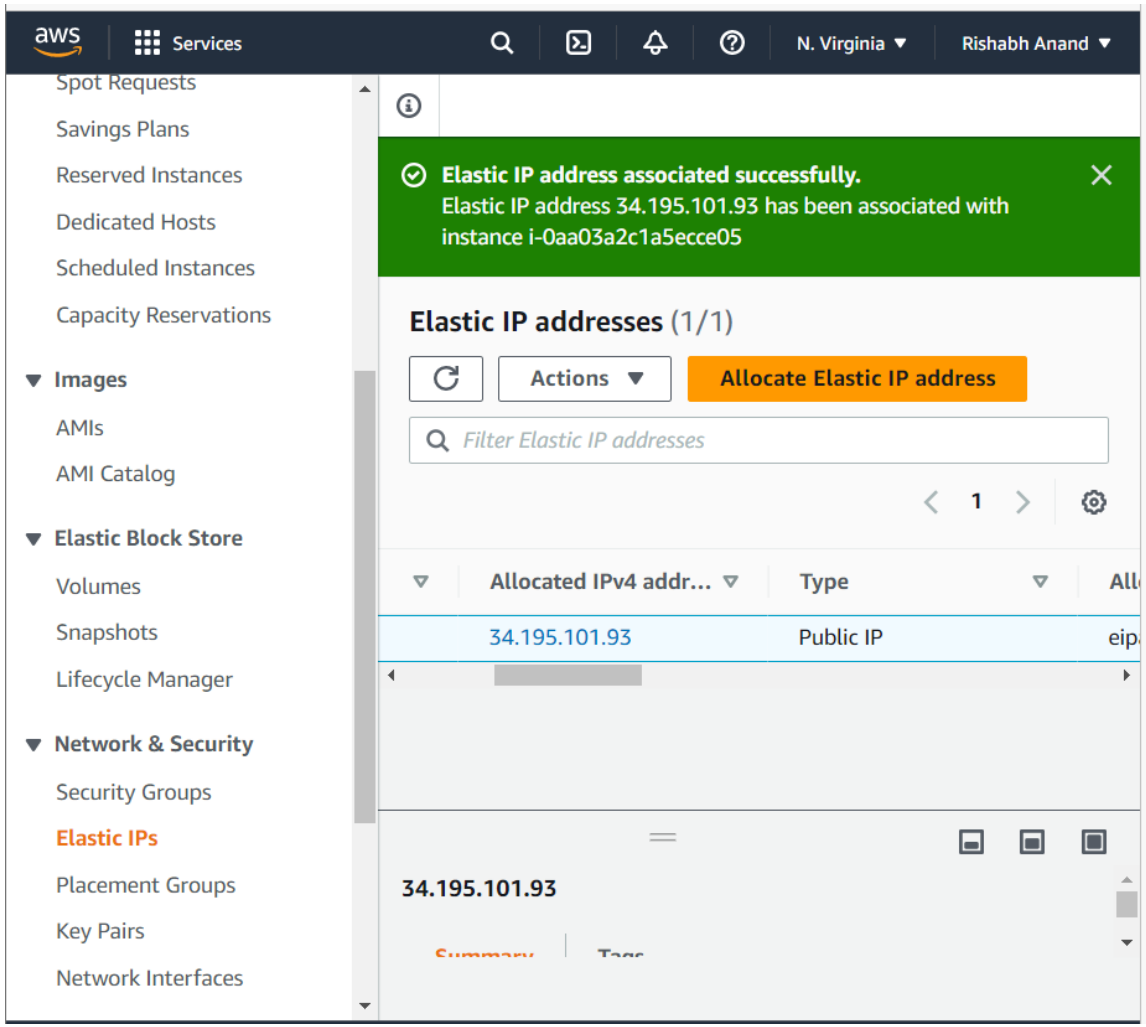
4.Automation: CloudWatch can be integrated with other AWS services to automate actions based on metrics or events, such as scaling up or down an EC2 instance based on CPU utilization.

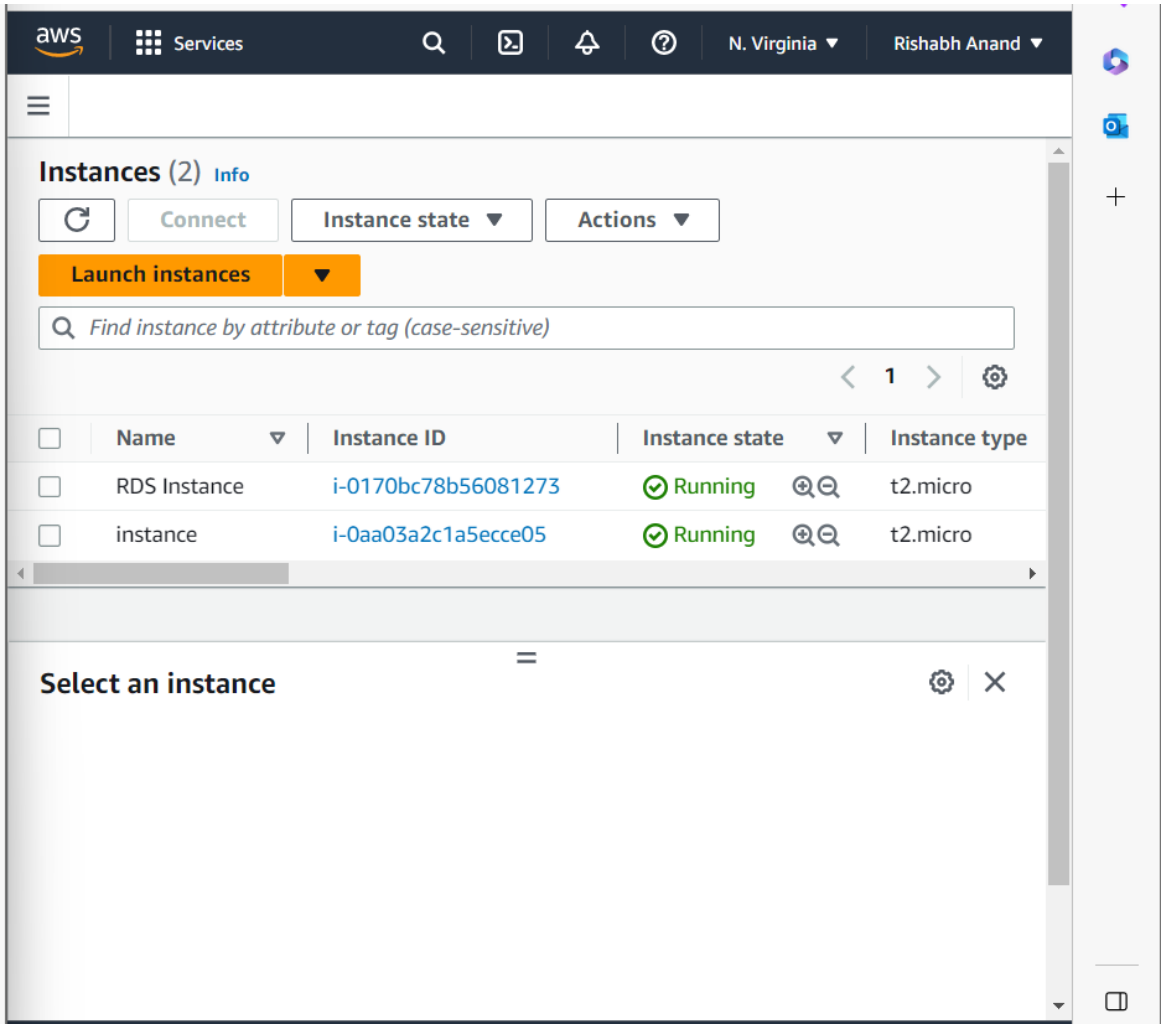
5.Alarms: CloudWatch allows users to set alarms based on specific metrics or events, such as when a certain threshold is exceeded or when a specific log event occurs. These alarms can be used to trigger automated actions or notify users of potential issues.

Implementing Security Group(firewall) and Elastic Ip

- Security Groups: Security groups act as virtual firewalls for instances, controlling inbound and outbound traffic to and from instances. They are used to define and manage network access to instances, allowing specific IP addresses or ranges to access resources within an AWS environment.

- Elastic IP (EIP): An Elastic IP address is a static, public IPv4 address that can be associated with an instance or a network interface in a VPC. It provides a fixed IP address that can be used to access resources within an AWS environment, making it easier to route traffic to a specific instance or service.





Link: https://drive.google.com/file/d/1yK6gwXDBdWn-jOHbWBJt61Y6n7TtjDWG/view?usp=share_link

Advantage:

1. Static IP Address: EIP provides a static, public IPv4 address that can be associated with an instance or a network interface in a VPC. This provides a fixed IP address that can be used to access resources within an AWS environment, making it easier to route traffic to a specific instance or service.
2. Easy to Reassign: EIP addresses can be easily reassigned between instances or services within an AWS environment, allowing for greater flexibility in managing resources.
3. Improved Availability: By associating an EIP address with an instance, the instance can be stopped and started without losing the public IP address. This can help to improve
4. Granular Control: Security groups enable granular control over traffic flow to and from instances, allowing for highly customized network access rules based on IP addresses, protocols, and ports.
5. Easy Management: Security groups are easy to manage and configure, with changes taking effect immediately. They can also be applied to multiple instances simultaneously, reducing the need for manual configuration.

Implementing Amazon Inspector

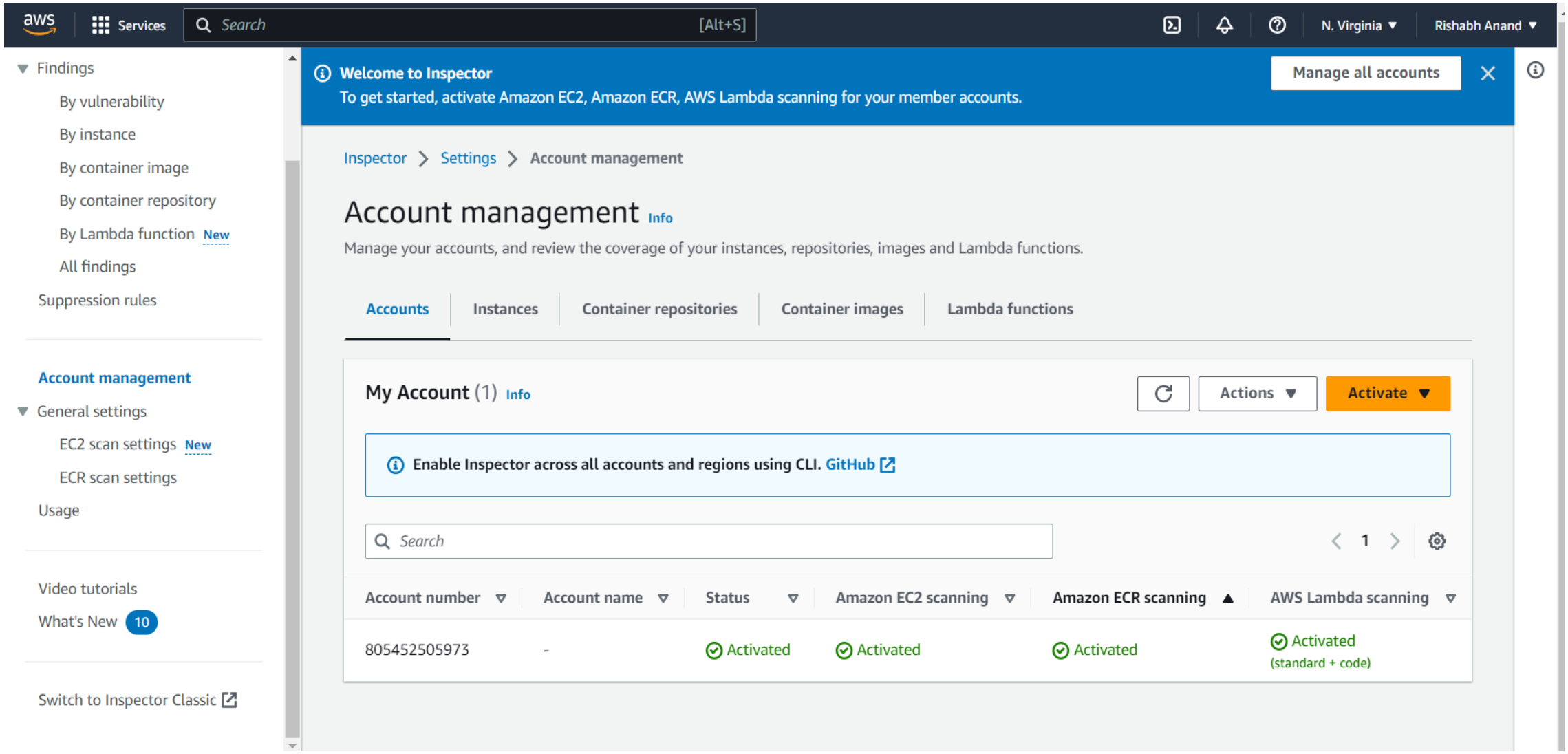
Amazon Inspector is an AWS security service that helps users improve the security and compliance of their applications deployed on AWS. It automatically assesses applications for vulnerabilities, exposures, and security issues, and provides users with detailed reports and recommendations for remediation.

Some advantages of Amazon Inspector include:

1.Automated Vulnerability Assessment: Amazon Inspector automatically discovers and assesses vulnerabilities in applications running on AWS, reducing the need for manual security assessments.

2.Continuous Monitoring: Amazon Inspector continuously monitors applications for security issues, helping users to detect and remediate security issues before they become major problems.

3.Integration with Other AWS Services: Amazon Inspector integrates with other AWS services such as AWS Identity and Access Management (IAM) and AWS CloudFormation, making it easier to manage and remediate security issues across an entire AWS environment.



Link: https://drive.google.com/file/d/1Fj8bV_0K2YbTgkK3lNNeBDxiAj-tJueL/view?usp=share_link

Implementing Standardize Tags

AWS Standardize Tags is a feature of Amazon Web Services (AWS) that allows users to apply consistent metadata to resources across their AWS account. AWS Standardize Tags feature provides a standardized set of tags that can be applied to resources in AWS, making it easier to manage and organize those resources.

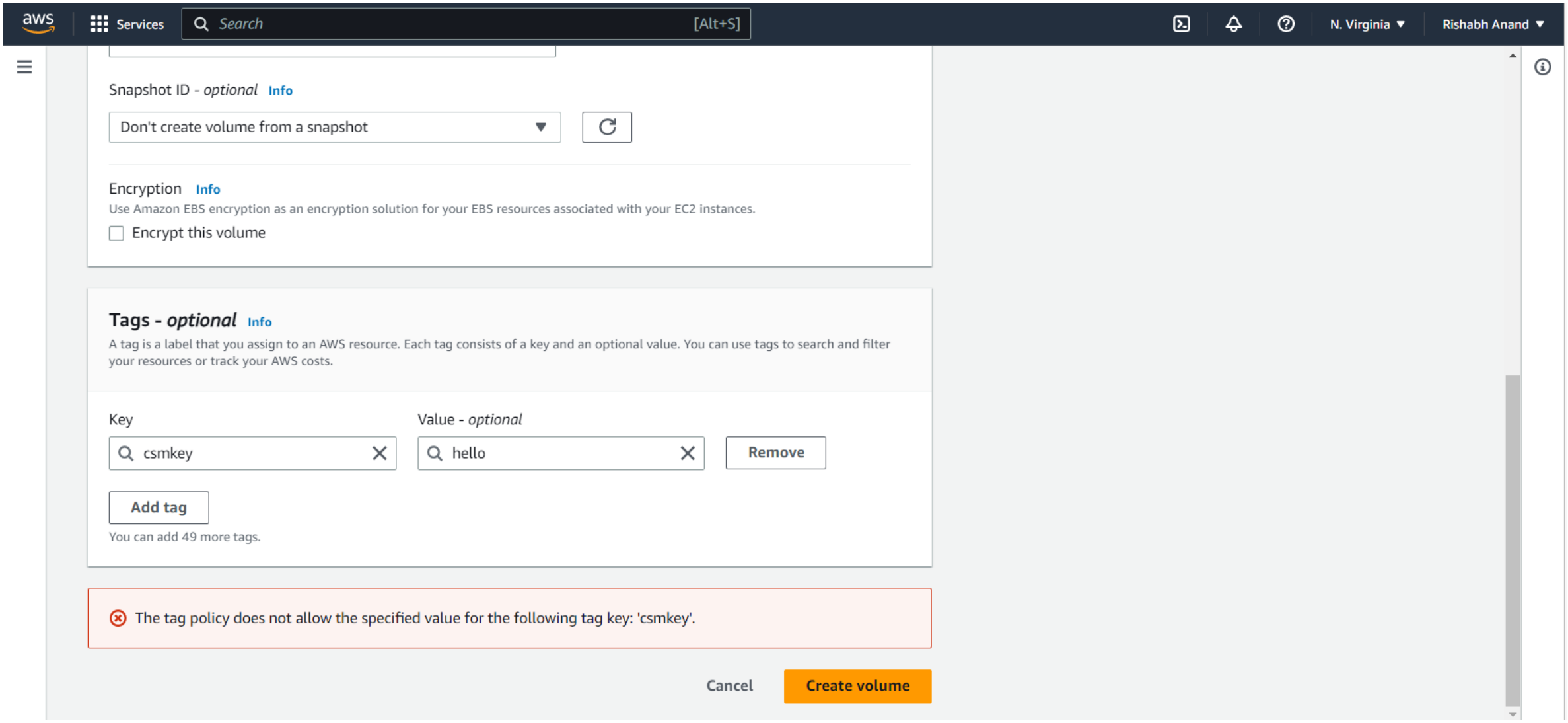
The advantages of using AWS Standardize Tags include:

1.Consistency: By using a standardized set of tags across all resources, it ensures that all resources are consistently labeled and organized.

2.Visibility: The standardized tags provide a consistent and organized view of all resources in AWS, making it easier to manage them and identify resources that need attention.

3.Automation: AWS Standardize Tags can be used in conjunction with AWS automation tools like AWS Lambda, AWS Config, AWS CloudFormation, and AWS Systems Manager, to automate resource tagging and simplify resource management.

4.Cost control: By applying standardized tags to resources, AWS customers can track and control their costs more effectively by identifying and managing resources based on their usage and cost.



Link: https://drive.google.com/file/d/1aMixffZamMhM7Ve1u-kfUGceHCVyLhuO/view?usp=share_link

Implementing Cloud front

Amazon CloudFront is a content delivery network (CDN) service provided by Amazon Web Services (AWS). It delivers content, including website assets and video streams, from Amazon's global network of data centers to end-users, reducing latency and improving performance.

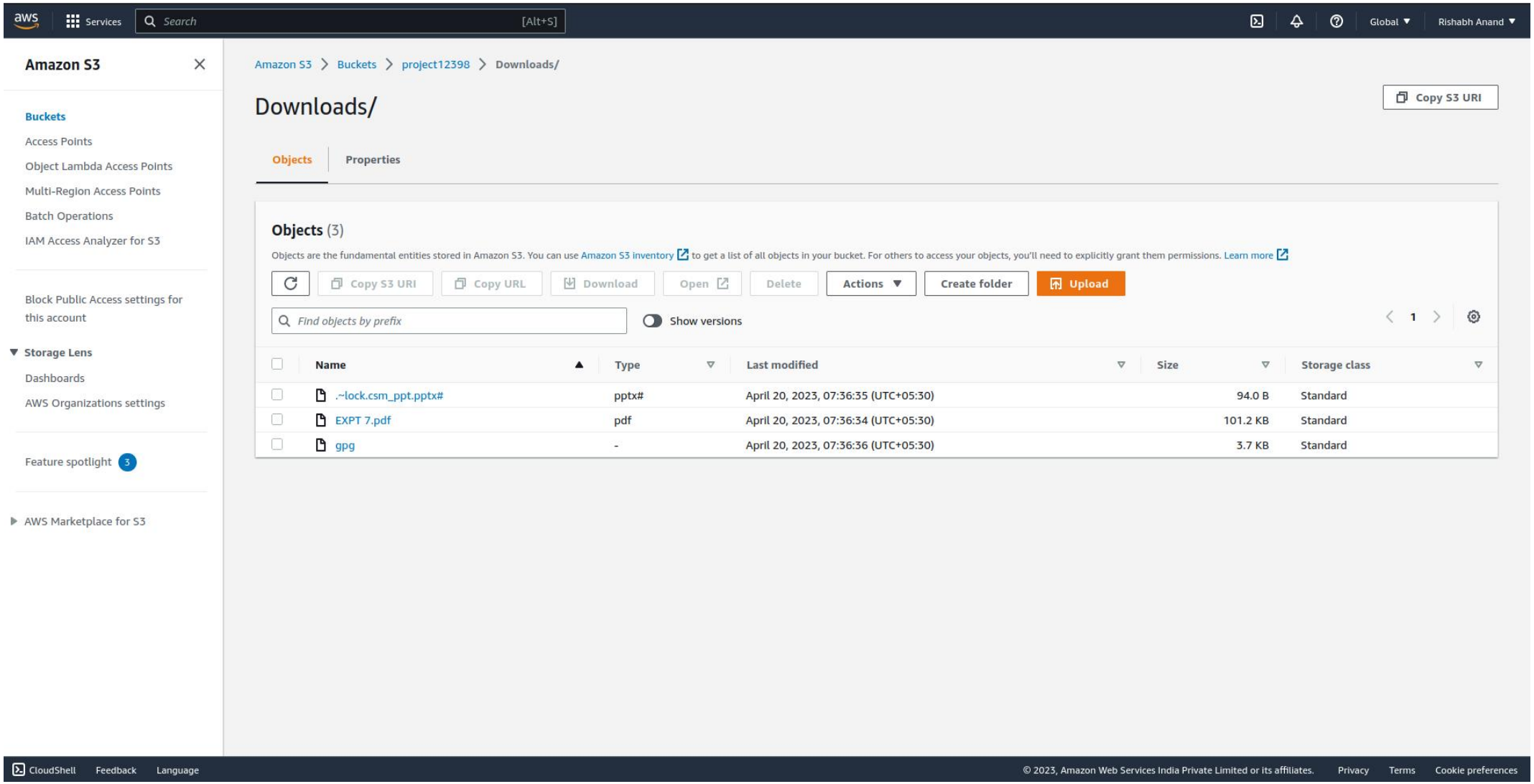
Some advantages of Amazon CloudFront include:

1.Improved Performance: Amazon CloudFront uses a global network of edge locations to deliver content to users with low latency, reducing the time it takes for content to reach end-users and improving overall performance.

2.Cost-Effective: Amazon CloudFront is a cost-effective way to deliver content globally, as it charges based on the amount of data transferred and requests processed, rather than fixed monthly fees.

3.Easy to Use: Amazon CloudFront is easy to set up and use, with simple configuration options and integration with other AWS services.

4.Scalability: Amazon CloudFront is highly scalable and can handle large volumes of traffic and content, ensuring that applications can handle spikes in traffic without impacting performance.



https://drive.google.com/file/d/1cYBhUiqBg9EPc3GF2pknq1tDSzKc5x5R/view?usp=share_link

Creating RDS & Implementing security Features

Amazon Relational Database Service (RDS) is a managed database service provided by Amazon Web Services (AWS) that allows users to easily deploy and manage relational databases in the cloud. RDS supports popular database engines such as MySQL, PostgreSQL, Oracle, SQL Server, and MariaDB.

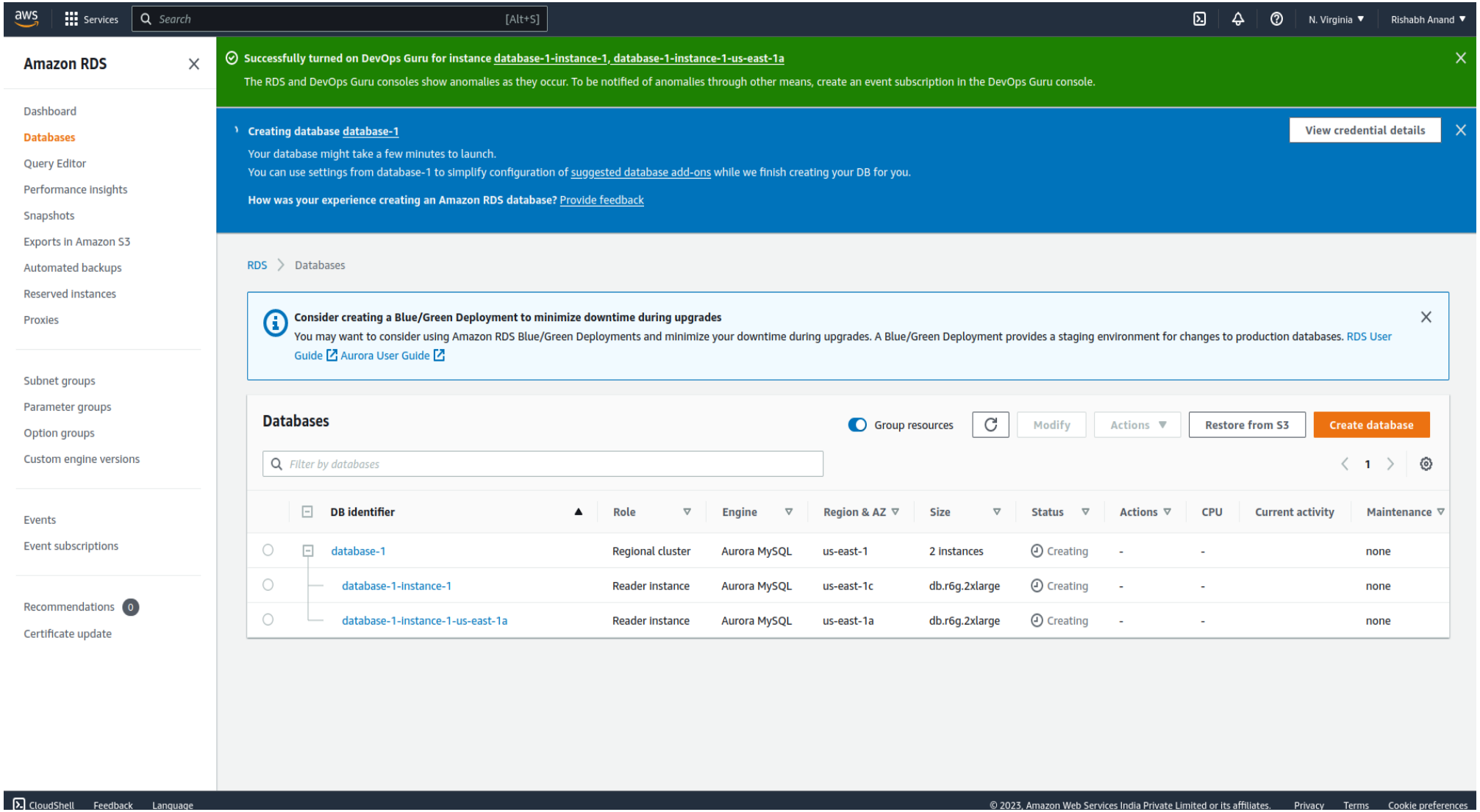
Advantages of implementing security features to RDS include:

1.Data Protection: Implementing security features to RDS can help protect the sensitive data stored in the database from unauthorized access, theft, or data breaches.

2.Compliance: Security features can help ensure that the RDS database meets regulatory and compliance requirements such as HIPAA, PCI DSS, and GDPR.

3.Access Control: Security features such as AWS Identity and Access Management (IAM), database authentication, and SSL/TLS encryption can help control access to the database and ensure that only authorized users can access it.

4.Monitoring and Logging: Security features can provide logging and monitoring capabilities to track database activity, detect potential security threats, and identify any suspicious behavior or unauthorized access attempts.



https://drive.google.com/file/d/187QCSrPDNbrH9Gj-mag4M8khPdOTAK9d/view?usp=share_link

Implementing KMS

AWS Key Management Service (KMS) is a managed service that allows users to easily create and control the encryption keys used to encrypt their data. It is a fully managed service that provides a highly secure way to manage encryption keys that can be used to encrypt data stored in AWS services such as Amazon S3, Amazon EBS, Amazon Redshift, Amazon RDS, and Amazon Elasticsearch Service.

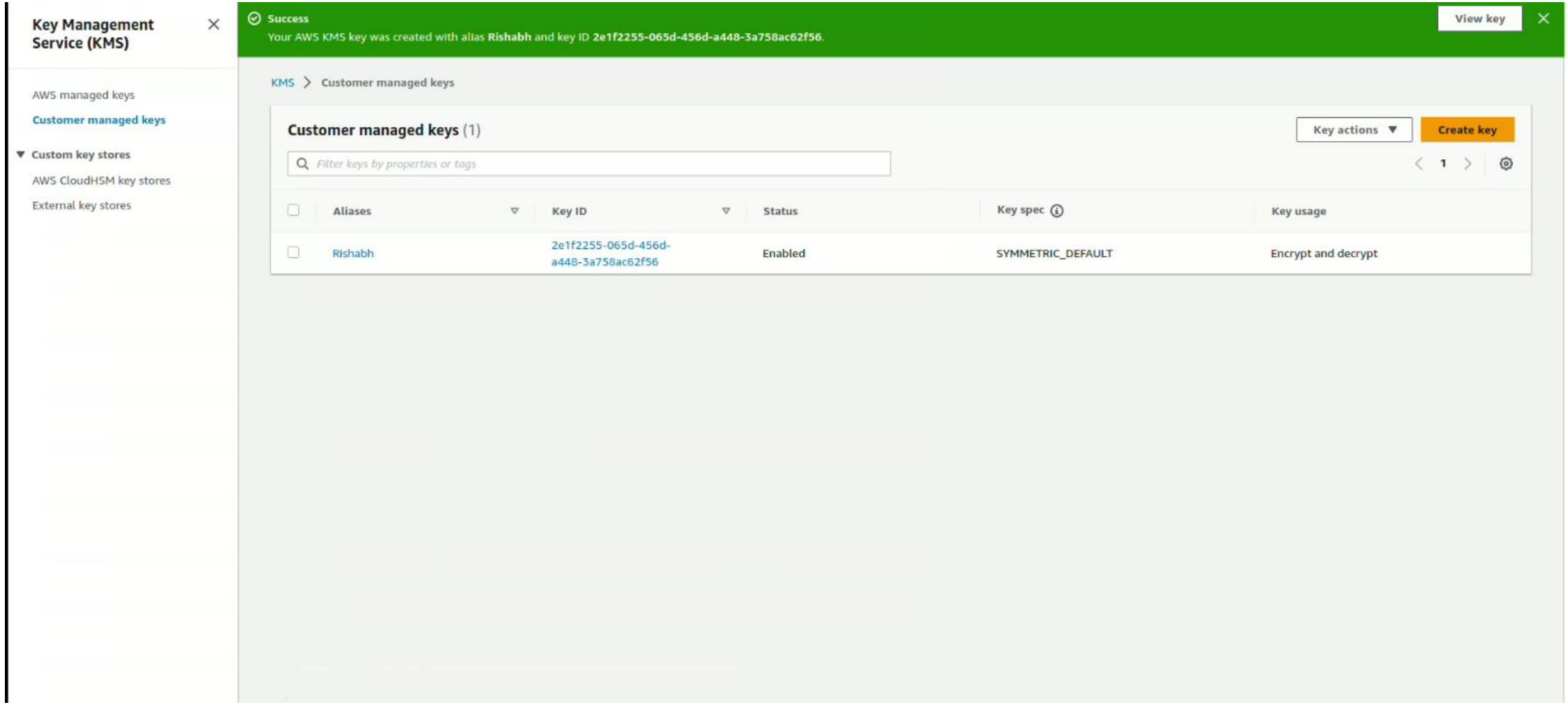
Some advantages of using AWS KMS include:

1.Ease of Use: AWS KMS is easy to set up and use, with a simple API and management console that allows users to create, manage, and audit their encryption keys.

2.Highly Secure: AWS KMS uses hardware security modules (HSMs) to protect the encryption keys, ensuring that they are stored and managed in a highly secure environment.

3.Integration with AWS Services: AWS KMS integrates seamlessly with other AWS services, allowing users to easily encrypt data stored in those services and ensuring that data is protected at all times.

4.Customizable: AWS KMS allows users to create and manage their own encryption keys, providing a highly customizable approach to data encryption.



https://drive.google.com/file/d/1Q9zKEp_LaqV4OxFuNpSfOHGdhp4r0mKS/view?usp=share_link

SWOT Analysis

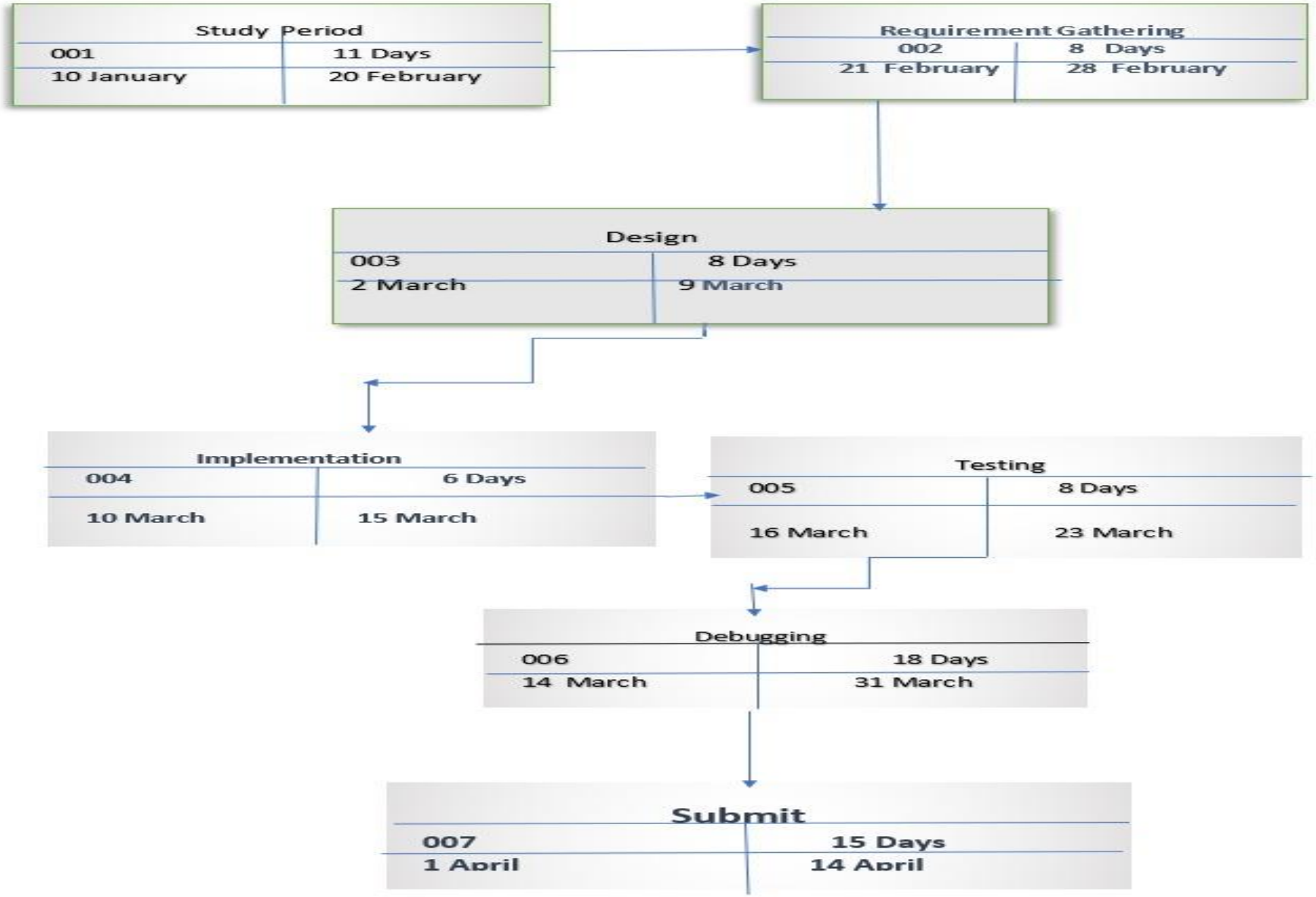
Strength: The strength of our project is that it provides security as well as maintenance to the website. We can access website deployed on it from anywhere all over interest and even configure it very easily. A very good user interface provided by aws through which user can easily modify the website

Weakness: The weakness of the project is that the website need to be registered then only aws could issue the SSL certificate. AWS provide the paid service so user need to pay to use the service.

Opportunities: It can have wide range of valid possibility that this project can be implemented over the big or small e commerce website to protect them from various attacks. Also helps it helps to manage various e commerce sites.

Threats: There is no threat for our project if the security credentials are secured. If it will be exposed any intruder can make changes to the website and take data.

Pert Chart


Study Period
001
11 Days
10 January
20 February
Requirement Gathering
002
8 Days
21 February
28 February
Design
003
8 Days
2 March
9 March
Implementation
004
6 Days
10 March
15 March
Testing
005
8 Days
16 March
23 March
Debugging
006
18 Days
14 March
31 March
Submit
007
15 Days
1 April
14 April

13. Objectives Covered

Objectives	Status
Creating the website	Completed
Deploying website on EC2	Completed
Implementing MFA	Completed
Granting access to only one s3 bucket	Completed
Implementing application load Balancer	Completed
Securing site using ACM	Completed
Creating Backup of the website using AMI	Completed
Implementing limit allowed EC2 instance with IAM policy	Completed
Implementing AWS Code Commit	Completed
Implementing AWS cloud watch	Completed
Implementing AWS cloud trail	Completed
Implementing Security Group(firewall) and Elastic IPs	Completed

Objectives	Status
Implementing AWS Inspector	Done
Implementing Standardize Tags	Done
Implementing CloudFront	Done
Implement KMS	Done
Creating RDS and implementing security Features	Done

UPES

UNIVERSITY OF TOMORROW

Thank You