BLACKBUCKS INTERNSHIP REPORT

Global Request Distribution Architecture for Webpage Access with Autoscaling, EFS and SNS Integration.

SUBMITTED BY

SAIPRANAY GANTA(20B91A05Q8)
SAMPANGI VENKATA MANIKANTA(20B91A05R0)
SIRIGIRI SAI MAHESH (20B91A05S6)

UNDER THE GUIDANCE OF MR. AASHU DEV

Blackbuck Engineers Pvt. Ltd
Road No 36, Jubilee Hills, Hyderabad

BLACKBUCK INTERNSHIP WORK

Team Members:

- SAIPRANAY GANTA (20B91A05Q8)
- SAMPANGI VENKATA MANIKANTA (20B91A05R0)
- SIRIGIRI SAI MAHESH (20B91A05S6)

Title:

Global Request Distribution Architecture for Webpage Access with Autoscaling, EFS and SNS Integration.

Abstract:

The architecture described above focuses on distributing user requests in a proportional manner using an Elastic Load Balancer (ELB) across multiple instances created by an Auto Scaling group. Additionally, an Elastic File System (EFS) is set up to store web pages. The instances created by the Auto Scaling group are directly connected to the EFS, allowing them to access and serve the web pages. The ELB manages the incoming requests and directs them to the appropriate instances hosting the web pages stored in the EFS. Moreover, the architecture utilizes the SNS service to send email notifications to the administrator whenever an instance is created, terminated, or encounters failures during creation or termination within the Auto Scaling group. This comprehensive solution ensures efficient distribution of user requests, seamless web page hosting, and enables timely notifications to the administrator regarding instance management events.

AWS PROJECT WORK

Team Members:

- Sai Pranay Ganta (20B91A05Q8)
- Sampangi Venkata Manikanta (20B91A05R0)
- Sirigiri Sai Mahesh (20B91A05S6)

Title:

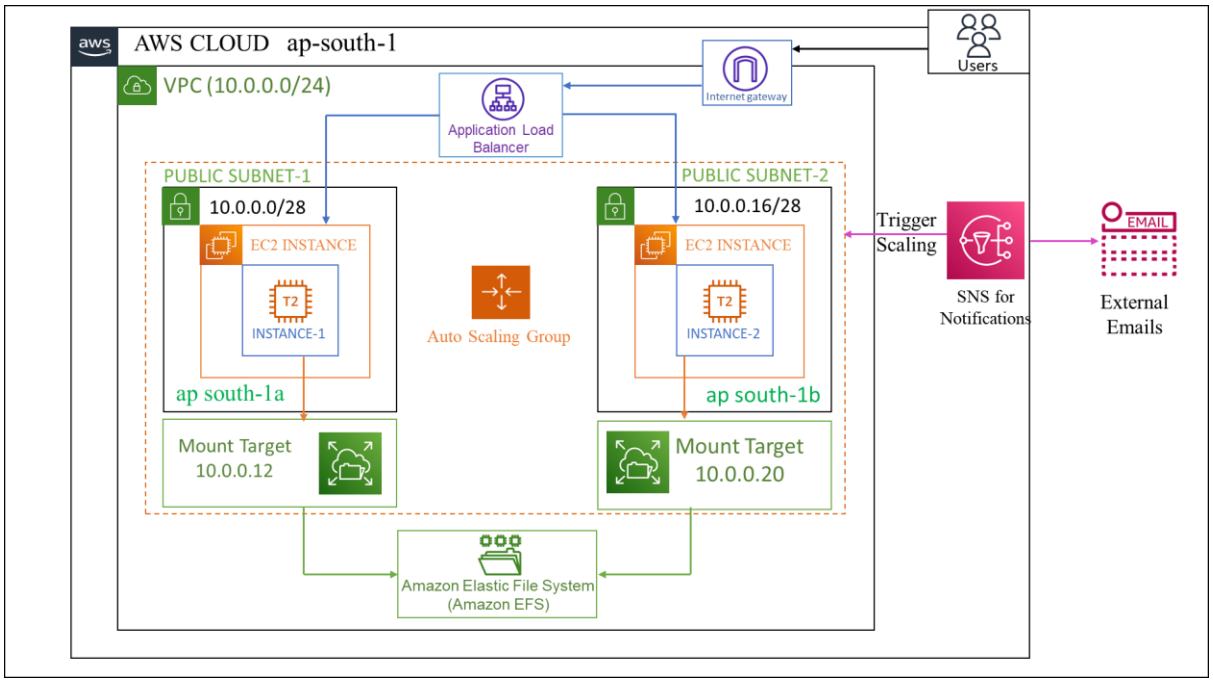
Global Request Distribution Architecture for Webpage Access with Autoscaling, EFS and SNS Integration.

Services used:

- EC2 (Elastic Compute Cloud)
- VPC (Virtual Private Cloud)
- Amazon Elastic File System (Amazon EFS)
- Auto Scaling Group
- Application Load Balancer
- Simple Notification service (SNS)

Rough architecture:

Final architecture:



Abstract:

The architecture described above focuses on distributing user requests in a proportional manner using an Elastic Load Balancer (ELB) across multiple instances created by an Auto Scaling group. Additionally, an Elastic File System (EFS) is set up to store web pages. The instances created by the Auto Scaling group are directly connected to the EFS, allowing them to access and serve the web pages. The ELB manages the incoming requests and directs them to the appropriate instances hosting the web pages stored in the EFS. Moreover, the architecture utilizes the SNS service to send email notifications to the administrator whenever an instance is created, terminated, or encounters failures during creation or termination within the Auto Scaling group. This comprehensive solution ensures efficient distribution of user requests, seamless web page hosting, and enables timely notifications to the administrator regarding instance management events.

Cloud computing

Cloud computing is on-demand access, via the internet, to computing resources—applications, servers (physical servers and virtual servers), data storage, development tools, networking capabilities, and more—hosted at a remote data center managed by a cloud services provider (or CSP). The CSP makes these resources available for a monthly subscription fee or bills them according to usage.

Compared to traditional on-premises IT, and depending on the cloud services you select, cloud computing helps do the following:

- **Lower IT costs:** Cloud lets you offload some or most of the costs and effort of purchasing, installing, configuring, and managing your own on-premises infrastructure.
- **Improve agility and time-to-value:** With cloud, your organization can start using enterprise applications in minutes, instead of waiting weeks or months for IT to respond to a request, purchase and configure supporting hardware, and install software. Cloud also lets you empower certain users—specifically developers and data scientists.
- **Scale more easily and cost-effectively:** Cloud provides elasticity—instead of purchasing excess capacity that sits unused during slow periods, you can scale capacity up and down in response to spikes and dips in traffic. You can also take advantage of your cloud provider's global network to spread your applications closer to users around the world.

The term 'cloud computing' also refers to the technology that makes cloud work. This includes some form of virtualized IT infrastructure—servers, operating system software, networking, and other infrastructure that's abstracted, using special software, so that it can be pooled and divided irrespective of physical hardware boundaries. For example, a single hardware server can be divided into multiple virtual servers.

Cloud Computing Services:

- IaaS (Infrastructure-as-a-Service)
- PaaS (Platform-as-a-Service)
- SaaS (Software-as-a-service)

are the three most common models of cloud services, and it's not uncommon for an organization to use all three.

IaaS (Infrastructure-as-a-Service)

IaaS provides on-demand access to fundamental computing resources—physical and virtual servers, networking, and storage—over the internet on a pay-as-you-go basis. IaaS enables end users to scale and shrink resources on an as-needed basis, reducing the need for high, up-front capital expenditures or unnecessary on-premises or 'owned' infrastructure and for overbuying resources to accommodate periodic spikes in usage.

In contrast to SaaS and PaaS (and even newer PaaS computing models such as containers and serverless), IaaS provides the users with the lowest-level control of computing resources in the cloud.

IaaS was the most popular cloud computing model when it emerged in the early 2010s. While it remains the cloud model for many types of workloads, use of SaaS and PaaS is growing at a much faster rate.

PaaS (Platform-as-a-service)

PaaS provides software developers with on-demand platform—hardware, complete software stack, infrastructure, and even development tools—for running, developing, and managing applications without the cost, complexity, and inflexibility of maintaining that platform on-premises.

With PaaS, the cloud provider hosts everything—servers, networks, storage, operating system software, middleware, databases—at their data center. Developers simply pick from a menu to 'spin up' servers and environments they need to run, build, test, deploy, maintain, update, and scale applications.

Today, PaaS is often built around *containers*, a virtualized compute model one step removed from virtual servers. containers virtualize the operating system, enabling developers to package the application with only the operating system services it needs to run on any platform, without modification and without need for middleware.

SaaS (Software-as-a-Service)

SaaS—also known as cloud-based software or cloud applications—is application software that's hosted in the cloud, and that user's access via a web browser, a dedicated desktop client, or an API that integrates with a desktop or mobile operating system. In most cases, SaaS users pay a monthly or annual subscription fee; some may offer 'pay-as-you-go' pricing based on your actual usage.

In addition to the cost savings, time-to-value, and scalability benefits of cloud, SaaS offers the following:

- **Automatic upgrades:** With SaaS, users take advantage of new features as soon as the provider adds them, without having to orchestrate an on-premises upgrade.

- **Protection from data loss:** Because SaaS stores application data in the cloud with the application, users don't lose data if their device crashes or breaks.

SaaS is the primary delivery model for most commercial software today—there are hundreds of thousands of SaaS solutions available, from the most focused industry and departmental applications to powerful enterprise software database and AI (artificial intelligence) software.

Cloud Service Providers:

- Amazon Web Services
- Microsoft Azure
- Google Cloud Platform
- Oracle
- IBM cloud
- Salesforce

Amazon Web Services:

Amazon Web Services, Inc. (AWS) is a subsidiary of Amazon that provides on-demand cloud computing platforms and APIs to individuals, companies, and governments, on a metered, pay-as-you-go basis. Oftentimes, clients will use this in combination with autoscaling (a process that allows a client to use more computing in times of high application usage, and then scale down to reduce costs when there is less traffic). These cloud computing web services provide various services related to networking, computing, storage, middleware, IoT and other processing capacity, as well as software tools via AWS server farms. This frees clients from managing, scaling, and patching hardware, and operating systems.

One of the foundational services is Amazon Elastic Compute Cloud (EC2), which allows users to have at their disposal a virtual cluster of computers, with extremely high availability, which can be interacted with over the internet via REST APIs, a CLI or the AWS console. AWS's virtual computers emulate most of the attributes of a real computer, including hardware central processing units (CPUs) and graphics processing units (GPUs) for processing; local/RAM memory; hard disk /SSD storage; a choice of operating systems; networking; and

pre-loaded application software such as web servers, databases, and customer relationship management (CRM).

AWS services are delivered to customers via a network of AWS server farms located throughout the world. Fees are based on a combination of usage (known as a "Pay-as-you-go" model), hardware, operating system, software, or networking features chosen by the subscriber required availability, redundancy, security, and service options. Subscribers can pay for a single virtual AWS computer, a dedicated physical computer, or clusters of either.

Amazon provides select portions of security for subscribers (e.g., physical security of the data centers) while other aspects of security are the responsibility of the subscriber (e.g., account management, vulnerability scanning, patching). AWS operates for many global geographical regions including seven in North America.

Amazon markets AWS to subscribers as a way of obtaining large-scale computing capacity more quickly and cheaply than building an actual physical server farm. All services are billed based on usage, but each service measures usage in varying ways. As of 2021 Q4, AWS has 33% market share for cloud infrastructure while the next two competitors Microsoft Azure and Google Cloud have 21%, and 10% respectively, according to Synergy Group.

Why AWS?

- **Easy to use:**

  AWS is designed to allow application providers, ISVs, and vendors to host your applications quickly and securely – whether an existing application or a new SaaS-based application. You can use the AWS Management Console or well-documented web services APIs to access AWS's application hosting platform**.**

- **Flexible:**

  AWS enables you to select the operating system, programming language, web application platform, database, and other services you need. With AWS, you receive a virtual environment that lets you load the software and services your application requires. This eases the migration process for existing applications while preserving options for building new solutions.

- **Cost-effective:**

  You pay only for the compute power, storage, and other resources you use, with no long-term contracts or up-front commitments. For more information on comparing the costs of other hosting alternatives with AWS, see the AWS Economics Center.

- **Reliable:**

  With AWS, you take advantage of a scalable, reliable, and secure global computing infrastructure, the virtual backbone of Amazon.com's multi-billion-dollar online business that has been honed for over a decade**.**

- **Scalable and High performance:**

  Using AWS tools, Auto Scaling, and Elastic Load Balancing, your application can scale up or down based on demand. Backed by Amazon's massive infrastructure, you have access to compute and storage resources when you need them.

- **Secure:**

  Using AWS tools, Auto Scaling, and Elastic Load Balancing, your application can scale up or down based on demand. Backed by Amazon's massive infrastructure, you have access to compute and storage resources when you need them.

List of AWS Services:

Amazon, the preeminent cloud vendor, broke new ground by establishing the first cloud computing service, Amazon EC2, in 2008. AWS offers more solutions and features than any other provider and has free tiers with access to the AWS Console, where users can centrally control their ministrations.

Designed around ease-of-use for various skill sets, AWS is tailored for those unaccustomed to software development utilities. Web applications can be deployed in minutes with AWS facilities, without provisioning servers or writing additional code.

- Amazon EC2 (Elastic Compute Cloud)
- Amazon RDS (Relational Database Services)
- Amazon S3 (Simple Storage Service)
- Amazon Lambda

- Amazon Cognito
- Amazon Glacier
- Amazon SNS (Simple Notification Service)
- Amazon VPC (Virtual Private Cloud)
- Amazon Lightsail
- Amazon CloudWatch
- Amazon Cloud9
- Amazon Elastic Beanstalk
- Amazon CodeCommit
- Amazon IAM (Identity and Access Management)
- Amazon Inspector
- Amazon Kinesis
- Amazon Dynamo DB
- Amazon Codecatalyst
- AWS Athena
- AWS Amplify
- AWS Quicksight
- AWS Cloudformation

Amazon EC2:

Amazon Elastic Compute Cloud (EC2) is a part of Amazon.com's cloud-computing platform, Amazon Web Services (AWS), that allows users to rent virtual computers on which to run their own computer applications. EC2 encourages scalable deployment of applications by providing a web service through which a user can boot an Amazon Machine Image (AMI) to configure a virtual machine, which Amazon calls an "instance", containing any software desired. A user can create, launch, and terminate server-instances as needed, paying by the second for active servers – hence the term "elastic". EC2 provides users with control over the geographical location of instances that allows for latency optimization and high levels of redundancy. In November 2010, Amazon switched its own retail website platform to EC2 and AWS.

Amazon announced a limited public beta test of EC2 on August 25, 2006, offering access on a first-come, first-served basis. Amazon added two new instance types (Large and Extra-Large)

on October 16, 2007. On May 29, 2008, two more types were added, High-CPU Medium and High-CPU Extra Large. There were twelve types of instances available.

Amazon added three new features on March 27, 2008, static IP addresses, availability zones, and user selectable kernels. On August 20, 2008, Amazon added Elastic Block Store (EBS) This provides persistent storage, a feature that had been lacking since the service was introduced.

Instance types:

Initially, EC2 used Xen virtualization exclusively. However, on November 6, 2017, Amazon announced the new C5 family of instances that were based on a custom architecture around the KVM hypervisor, called Nitro. Each virtual machine, called an "instance", functions as a virtual private server. Amazon sizes instances based on "Elastic Compute Units". The performance of otherwise identical virtual machines may vary. On November 28, 2017, AWS announced a bare-metal instance type offering marking a remarkable departure from exclusively offering virtualized instance types.

As of January 2019, the following instance types were offered:

- General Purpose: A1, T3, T2, M5, M5a, M4, T3a
- Compute Optimized: C5, C5n, C4
- Memory Optimized: R5, R5a, R4, X1e, X1, High Memory, z1d
- Accelerated Computing: P3, P2, G3, F1
- Storage Optimized: H1, I3, D2

As of April 2018, the following payment methods by instance were offered:

- On-demand: pay by the hour without commitment.
- Reserved: rent instances with one-time payment receiving discounts on the hourly charge.
- Spot: bid-based service runs the jobs only if the spot price is below the bid specified by bidder. The spot price is claimed to be supply-demand based, however a 2011 study concluded that the price was generally not set to clear the market but was dominated by an undisclosed reserve price.

Amazon EBS (Elastic Block Store):

Amazon Elastic Block Store (Amazon EBS) provides block level storage volumes for use with EC2 instances. EBS volumes behave like raw, unformatted block devices. You can mount these volumes as devices on your instances. EBS volumes that are attached to an instance are exposed as storage volumes that persist independently from the life of the instance. You can create a file system on top of these volumes or use them in any way you would use a block device (such as a hard drive). You can dynamically change the configuration of a volume attached to an instance.

We recommend Amazon EBS for data that must be quickly accessible and requires long-term persistence. EBS volumes are particularly well-suited for use as the primary storage for file systems, databases, or for any applications that require fine granular updates and access to raw, unformatted, block-level storage. Amazon EBS is well suited to both database-style applications that rely on random reads and writes, and to throughput-intensive applications that perform long, continuous reads and writes.

With Amazon EBS, you pay only for what you use. For more information about Amazon EBS pricing, see the Projecting Costs Section of the Amazon Elastic Block Store page.

Features of Amazon EBS

- You create an EBS volume in a specific Availability Zone, and then attach it to an instance in that same Availability Zone. To make a volume available outside of the Availability Zone, you can create a snapshot and restore that snapshot to a new volume anywhere in that Region. You can copy snapshots to other Regions and then restore them to new volumes there, making it easier to leverage multiple AWS Regions for geographical expansion, data center migration, and disaster recovery.

- Amazon EBS provides the following volume types: General Purpose SSD, Provisioned IOPS SSD, Throughput Optimized HDD, and Cold HDD. For more information, see EBS volume types.
  The following is a summary of performance and use cases for each volume type.
  - General Purpose SSD volumes (`gp2` and `gp3`) balance price and performance for a wide variety of transactional workloads. These volumes are ideal for use cases such as boot volumes, medium-size single instance databases, and development and test environments.
  - Provisioned IOPS SSD volumes (`io1` and `io2`) are designed to meet the needs of I/O-intensive workloads that are sensitive to storage performance and consistency. They provide a consistent IOPS rate that you specify when you create the volume. This enables you to predictably scale to tens of thousands of IOPS per instance. Additionally, io2 volumes provide the highest levels of volume durability.
  - Throughput Optimized HDD volumes (`st1`) provide low-cost magnetic storage that defines performance in terms of throughput rather than IOPS. These volumes are ideal for large, sequential workloads such as Amazon EMR, ETL, data warehouses, and log processing.
  - Cold HDD volumes (`sc1`) provide low-cost magnetic storage that defines performance in terms of throughput rather than IOPS. These volumes are ideal for large, sequential, cold-data workloads. If you require infrequent access to your data and are looking to save costs, these volumes provide inexpensive block storage.

- You can create your EBS volumes as encrypted volumes, in order to meet a wide range of data-at-rest encryption requirements for regulated/audited data and applications. When you create an encrypted EBS volume and attach it to a supported instance type, data stored at rest on the volume, disk I/O, and snapshots created from the volume are all encrypted. Encryption occurs on the servers that host EC2 instances, providing encryption of data-in-transit from EC2 instances to EBS storage. For more information, see Amazon EBS encryption.

- Performance metrics, such as bandwidth, throughput, latency, and average queue length, are available through the AWS Management Console. These metrics, provided by Amazon CloudWatch, allow you to monitor the performance of your volumes to make sure that you are providing enough performance for your applications without paying for resources you don't need.

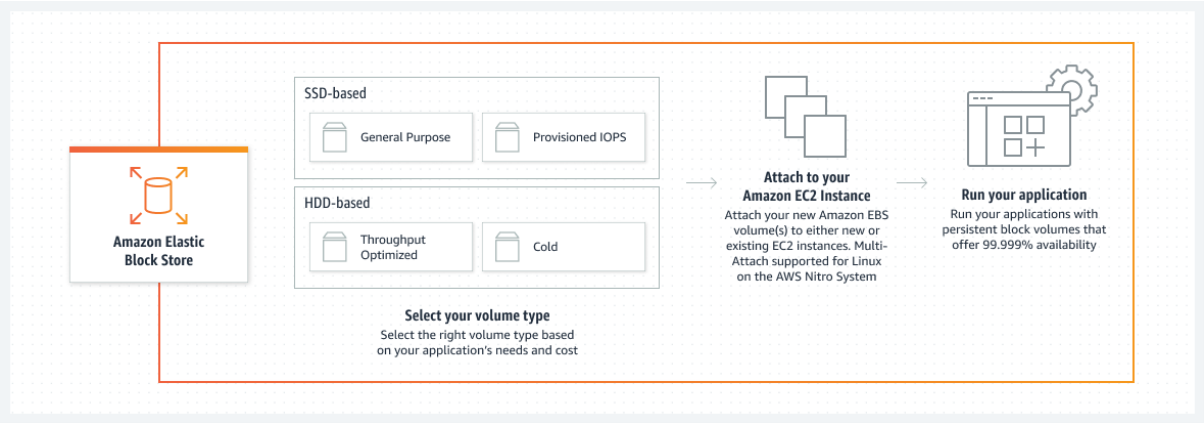
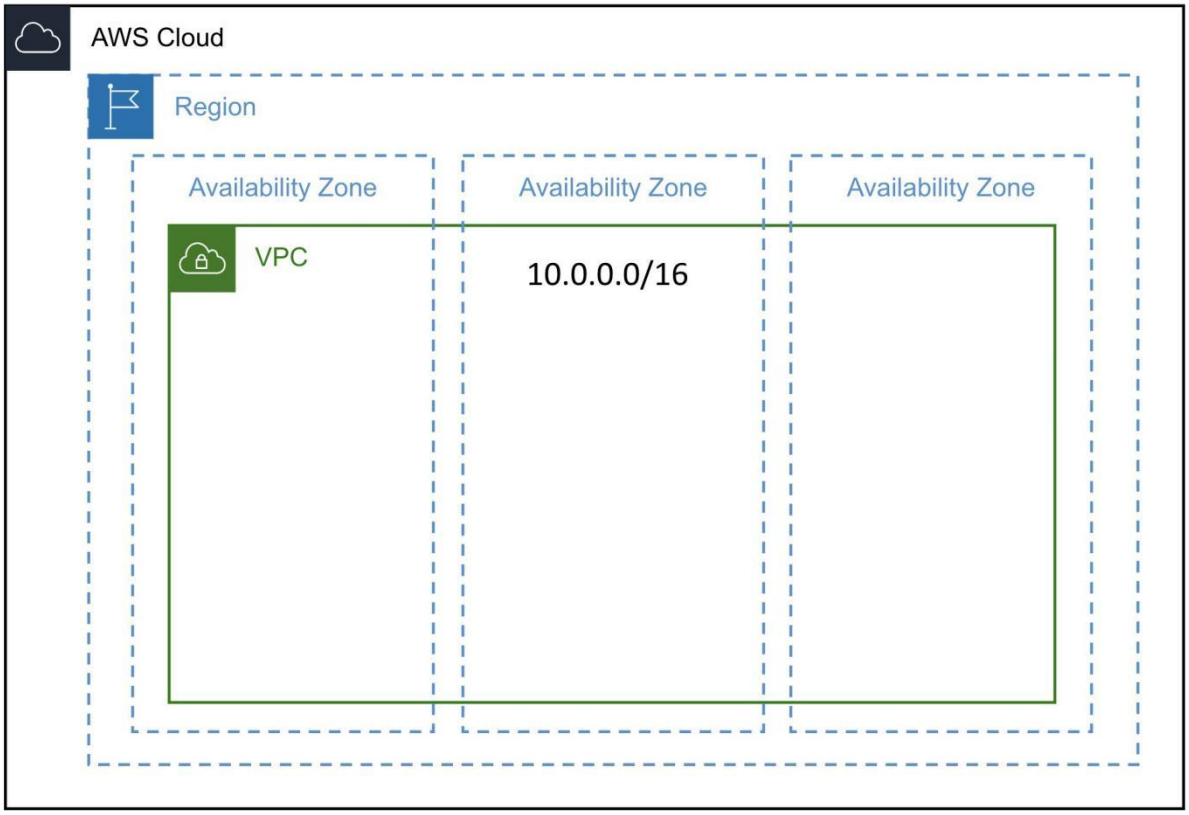


Fig. High-Performance Block Storage

Amazon VPC:

Amazon Virtual Private Cloud (VPC) is a commercial cloud computing service that provides a virtual private cloud, by provisioning a logically isolated section of Amazon Web Services (AWS) Cloud. Enterprise customers are able to access the Amazon Elastic Compute Cloud (EC2) over an IPsec based virtual private network. Unlike traditional EC2 instances which are allocated internal and external IP numbers by Amazon, the customer can assign IP numbers of their choosing from one or more subnets.



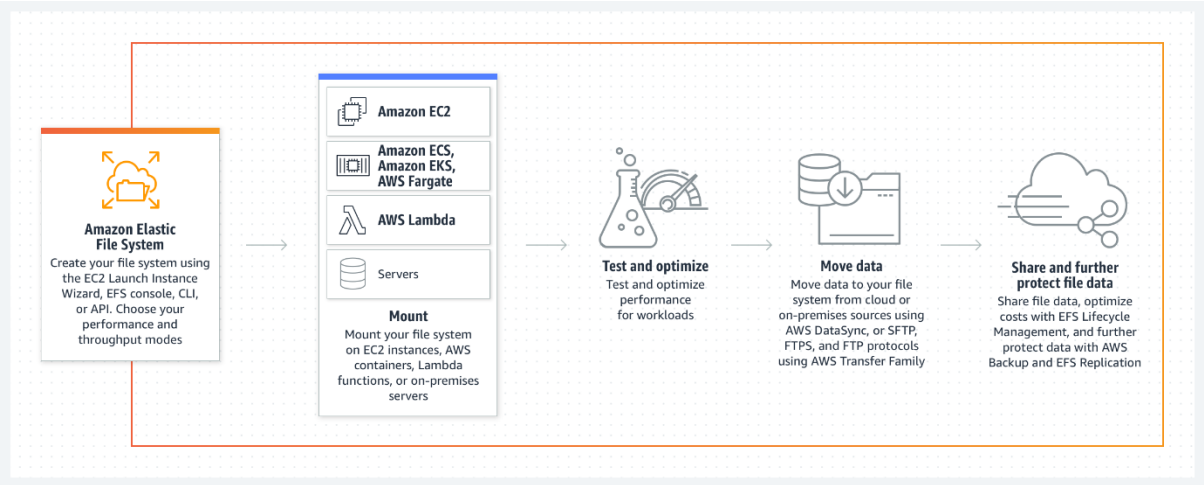
Amazon Web Services launched Amazon Virtual Private Cloud on 26 August 2009, which allows the Amazon Elastic Compute Cloud service to be connected to legacy infrastructure over an IPsec VPN.In AWS, the basic VPC is free to use, with users being charged by usage for additional features.EC2 and RDS instances running in a VPC can also be purchased using Reserved Instances, however will have a limitation on resources being guaranteed.[citation needed]

IBM Cloud launched IBM Cloud VPC on 4 June 2019, provides an ability to manage virtual machine-based compute, storage, and networking resources. Pricing for IBM Cloud Virtual Private Cloud is applied separately for internet data transfer, virtual server instances, and block storage used within IBM Cloud VPC.

Google Cloud Platform resources can be provisioned, connected, and isolated in a virtual private cloud (VPC) across all GCP regions. With GCP, VPCs are global resources and subnets within that VPC are regional resources. This allows users to connect zones and regions without the use of additional networking complexity as all data travels, encrypted in transit and at rest, on Google's own global, private network. Identity management policies and security rules allow for private access to Google's storage, big data, and analytics managed services. VPCs on Google Cloud Platform leverage the security of Google's data centers.

Elastic File System:

Amazon Elastic File System (EFS) is a fully managed, scalable, and highly available file storage service provided by Amazon Web Services (AWS). EFS is designed to provide shared file storage for EC2 instances, allowing multiple instances to access and modify files concurrently. It offers a simple and scalable solution for applications that require shared file storage, such as content management systems, web servers, and Big Data analytics. EFS automatically scales storage capacity as needed, without any upfront provisioning, ensuring that applications have the required storage resources to handle growing workloads. It supports the Network File System (NFS) protocol, making it compatible with a wide range of Linux-based applications and services. EFS provides high durability and availability, replicating data across multiple availability zones within a region, thereby protecting against data loss and ensuring continuous access to files. With its elastic scalability, ease of use, and strong reliability, EFS simplifies file storage management and provides a flexible solution for shared file access in AWS environments.

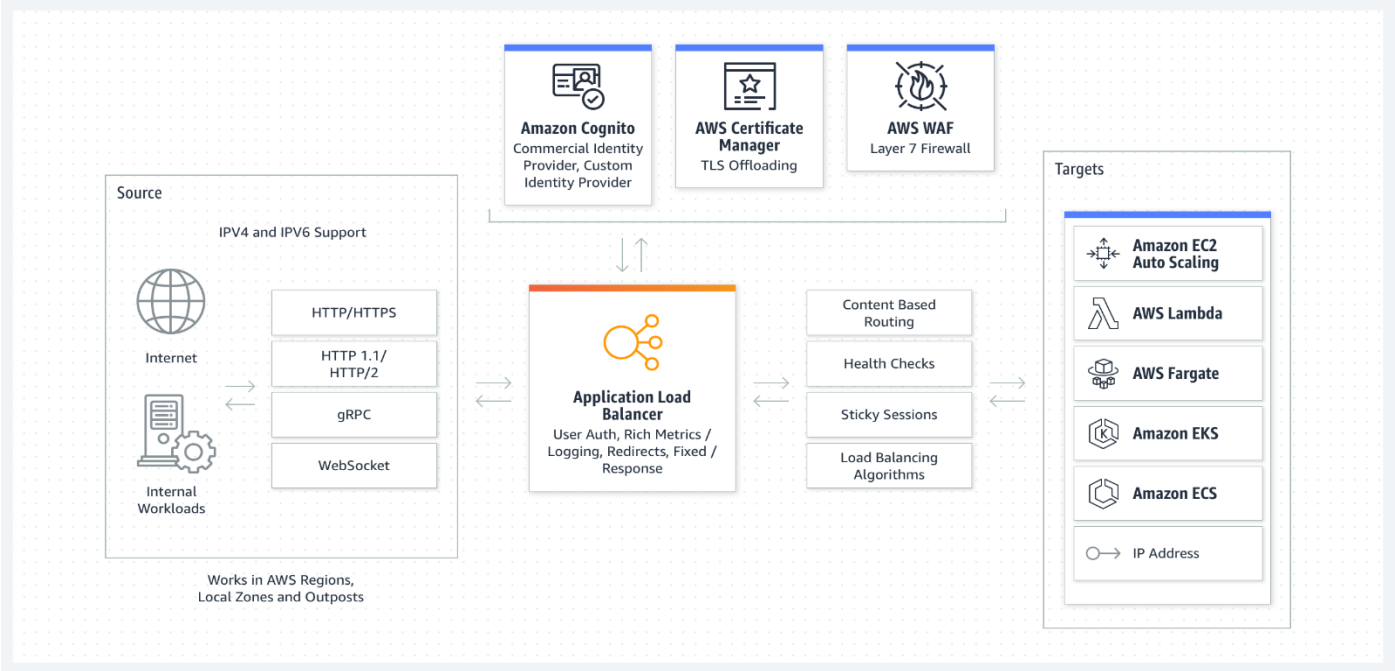


Load Balancer:

Load Balancer in AWS is a service that evenly distributes incoming network traffic across multiple instances or services, ensuring high availability and optimal resource utilization. With Classic Load Balancer (CLB) and Application Load Balancer (ALB) options, it can operate at the transport or application layer, respectively. Load balancers automatically handle traffic routing, detect and redirect requests from unhealthy instances to healthy ones, and provide advanced features like content-based routing and SSL termination. By distributing traffic and providing redundancy, load balancers enhance scalability, availability, and performance of applications deployed in AWS, making them a crucial component for achieving reliable and efficient workload management.

Application Load Balancer:

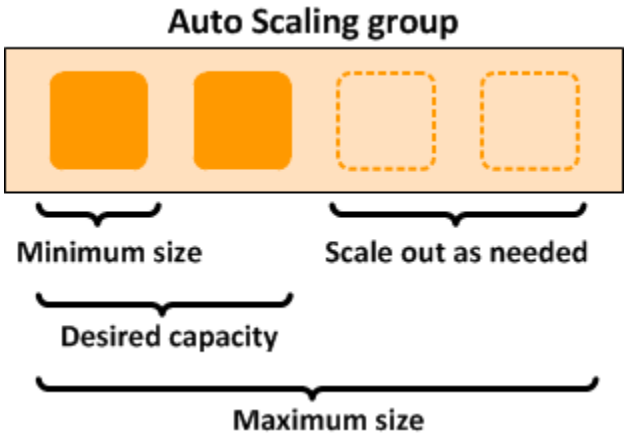
The Application Load Balancer (ALB) is an advanced load balancing solution provided by Amazon Web Services (AWS). It operates at the application layer (Layer 7) of the network stack, allowing it to intelligently route traffic based on content and application-specific requirements. ALB supports HTTP and HTTPS protocols, enabling it to handle web traffic and provide secure connections. It offers features like content-based routing, allowing requests to be directed to different backend services based on specific URL patterns or headers. ALB also supports advanced health checks, SSL/TLS termination, and sticky sessions for session persistence. With its flexible and powerful routing capabilities, ALB is an essential component for building highly scalable, fault-tolerant, and performant applications in AWS.



AWS Autoscaling:

Amazon EC2 Auto Scaling helps you ensure that you have the correct number of Amazon EC2 instances available to handle the load for your application. You create collections of EC2 instances, called *Auto Scaling groups*. You can specify the minimum number of instances in each Auto Scaling group, and Amazon EC2 Auto Scaling ensures that your group never goes below this size. You can specify the maximum number of instances in each Auto Scaling group, and Amazon EC2 Auto Scaling ensures that your group never goes above this size. If you specify the desired capacity, either when you create the group or at any time thereafter, Amazon EC2 Auto Scaling ensures that your group has this many instances. If you specify scaling policies, then Amazon EC2 Auto Scaling can launch or terminate instances as demand on your application increases or decreases.

For example, the following Auto Scaling group has a minimum size of one instance, a desired capacity of two instances, and a maximum size of four instances. The scaling policies that you define adjust the number of instances, within your minimum and maximum number of instances, based on the criteria that you specify.



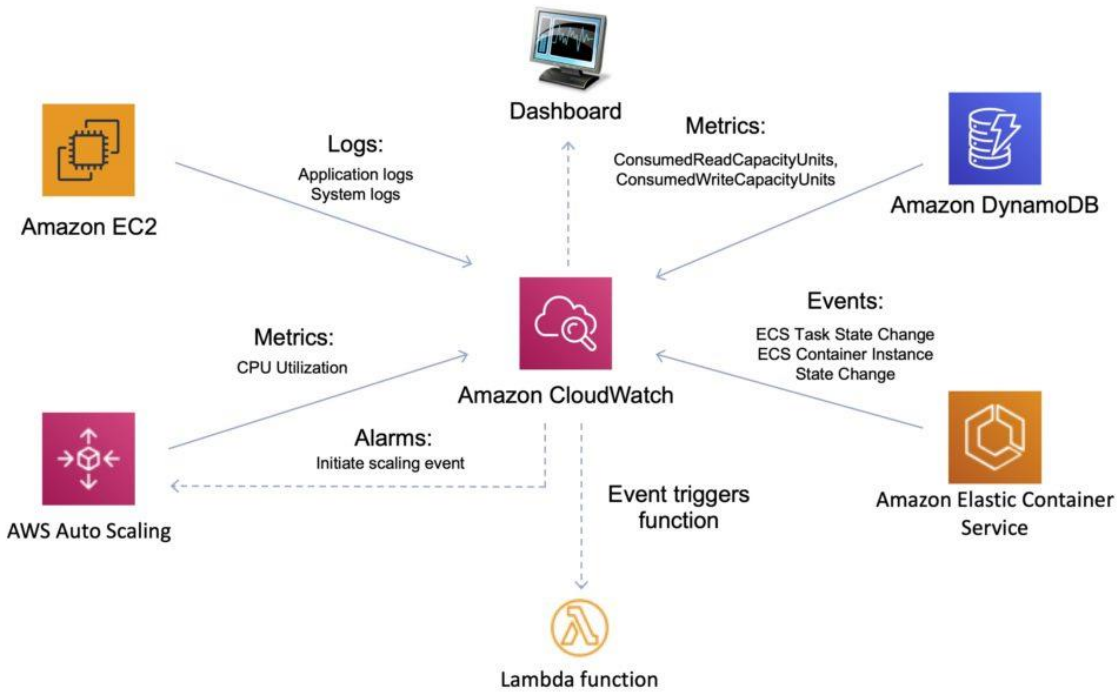
Amazon CloudWatch:

Amazon CloudWatch monitors your Amazon Web Services (AWS) resources and the applications you run on AWS in real time. You can use CloudWatch to collect and track metrics, which are variables you can measure for your resources and applications.

The CloudWatch home page automatically displays metrics about every AWS service you use. You can additionally create custom dashboards to display metrics about your custom applications and display custom collections of metrics that you choose.

You can create alarms that watch metrics and send notifications or automatically make changes to the resources you are monitoring when a threshold is breached. For example, you can monitor the CPU usage and disk reads and writes of your Amazon EC2 instances and then use that data to determine whether you should launch additional instances to handle increased load. You can also use this data to stop underused instances to save money.

With CloudWatch, you gain system-wide visibility into resource utilization, application performance, and operational health.



AWS Simple Notification Service :

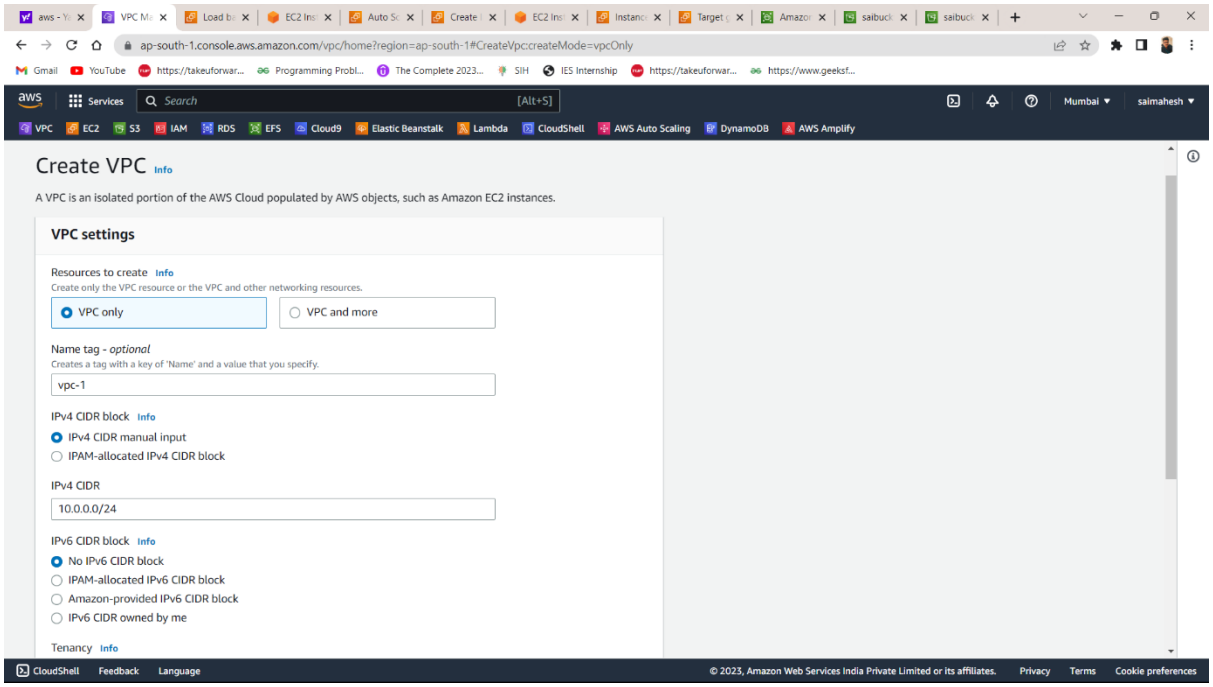
The Simple Notification Service (SNS) in AWS is a fully managed messaging service that enables the sending and receiving of notifications and messages. SNS provides a highly scalable and flexible solution for pub/sub messaging, allowing publishers to send messages to multiple subscribers. It supports various communication protocols, including HTTP, email, SMS, and mobile push notifications, ensuring seamless delivery across multiple channels. SNS simplifies the process of building distributed systems by decoupling components and enabling asynchronous communication. It offers features like message filtering, topic-based message filtering, and message attributes for granular control over message delivery. With its ease of use, scalability, and versatility, SNS is a reliable service for building event-driven architectures and real-time messaging applications in AWS.

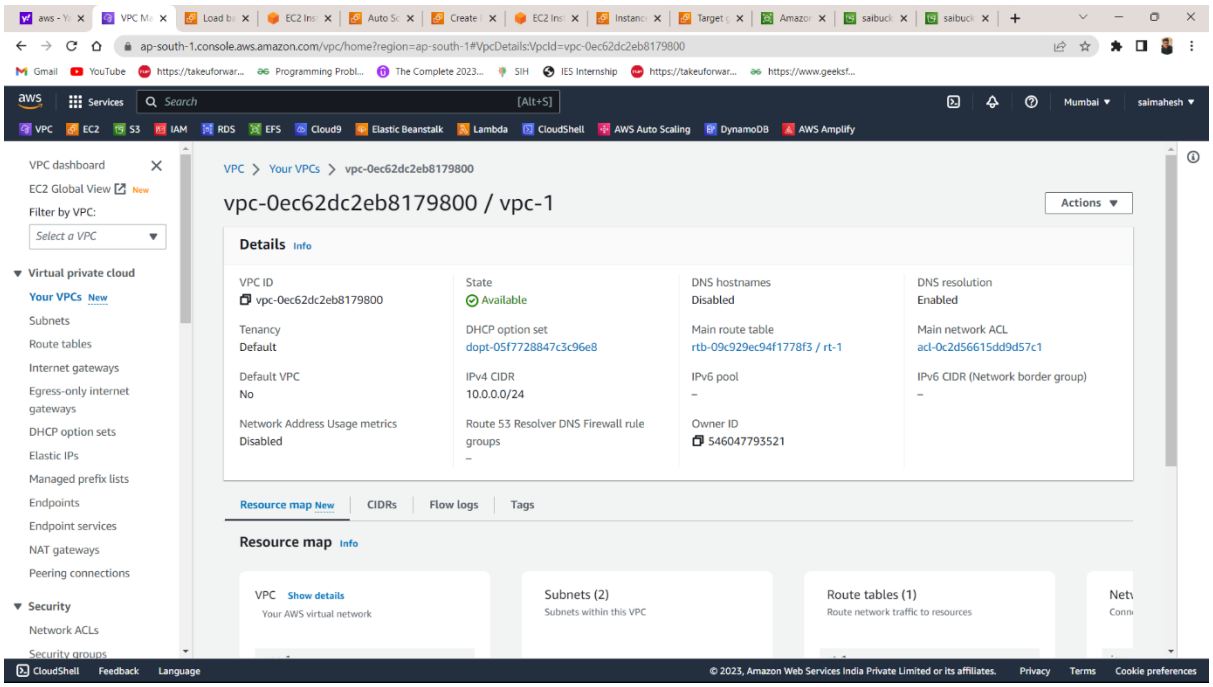
IMPLEMENTATION

Steps to perform:

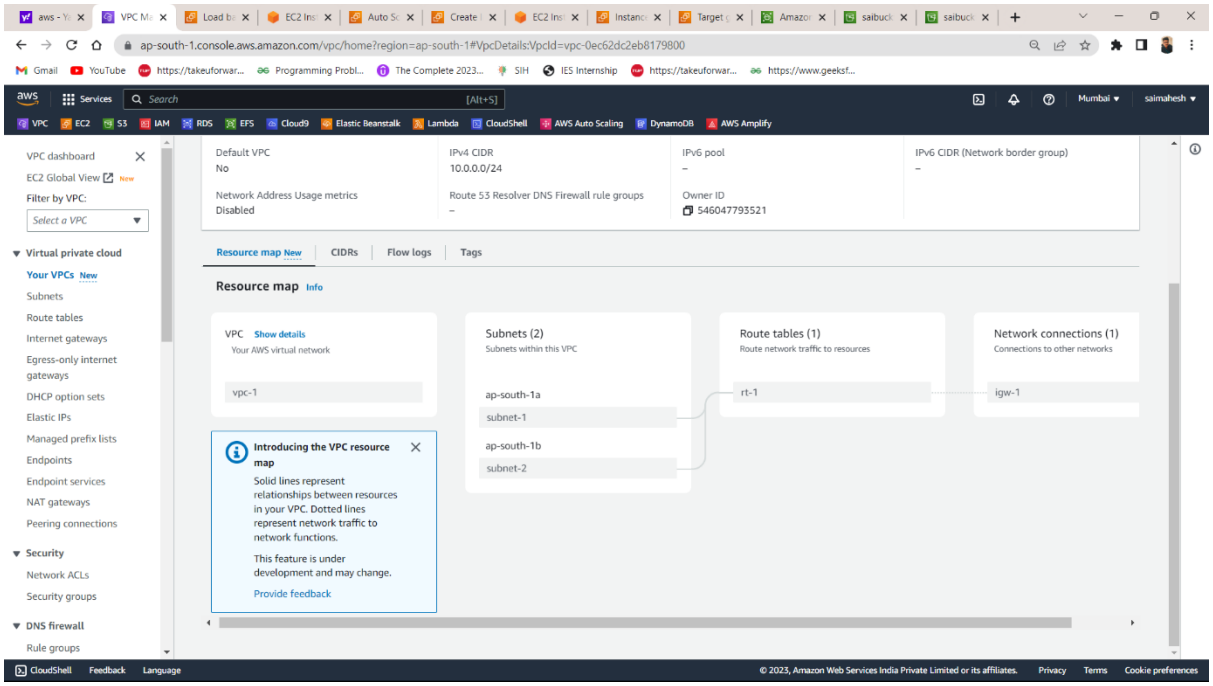
VPC: Creating a custom VPC

Give a name to the Custom VPC and an IPV4 address



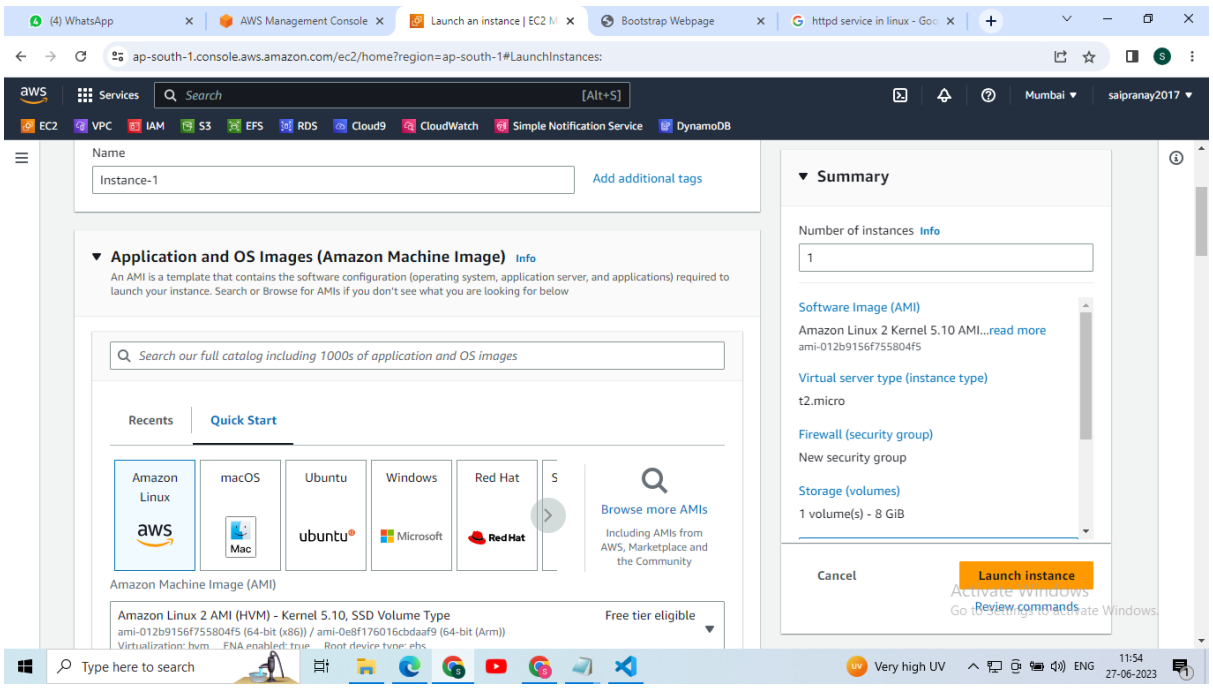


VPC created

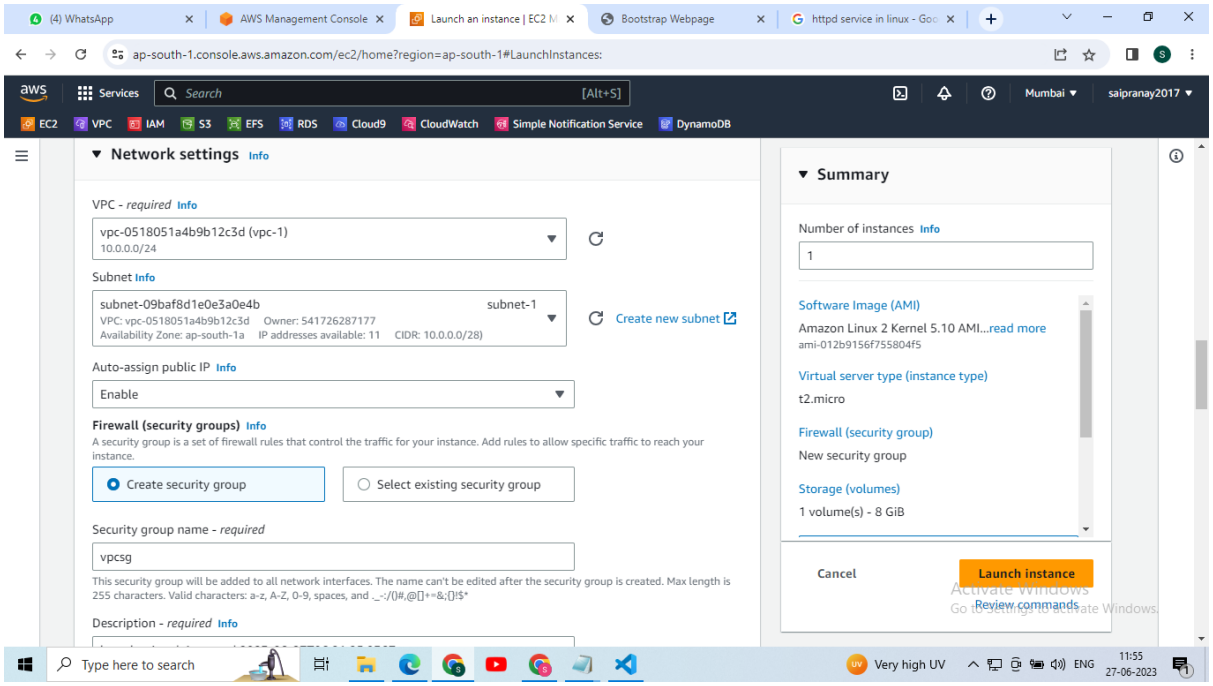


This is the flow diagram of how a vpc and the subnets created are connected to VPC and the VPC connected to the internet gateway.

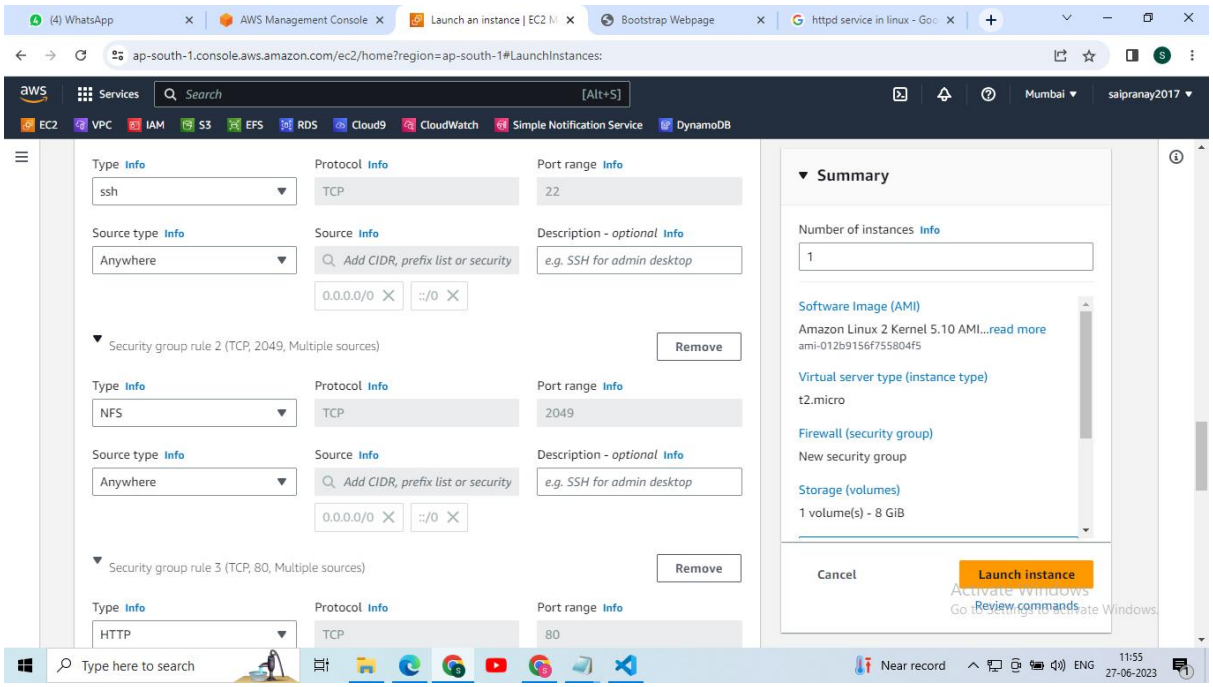
EC2:



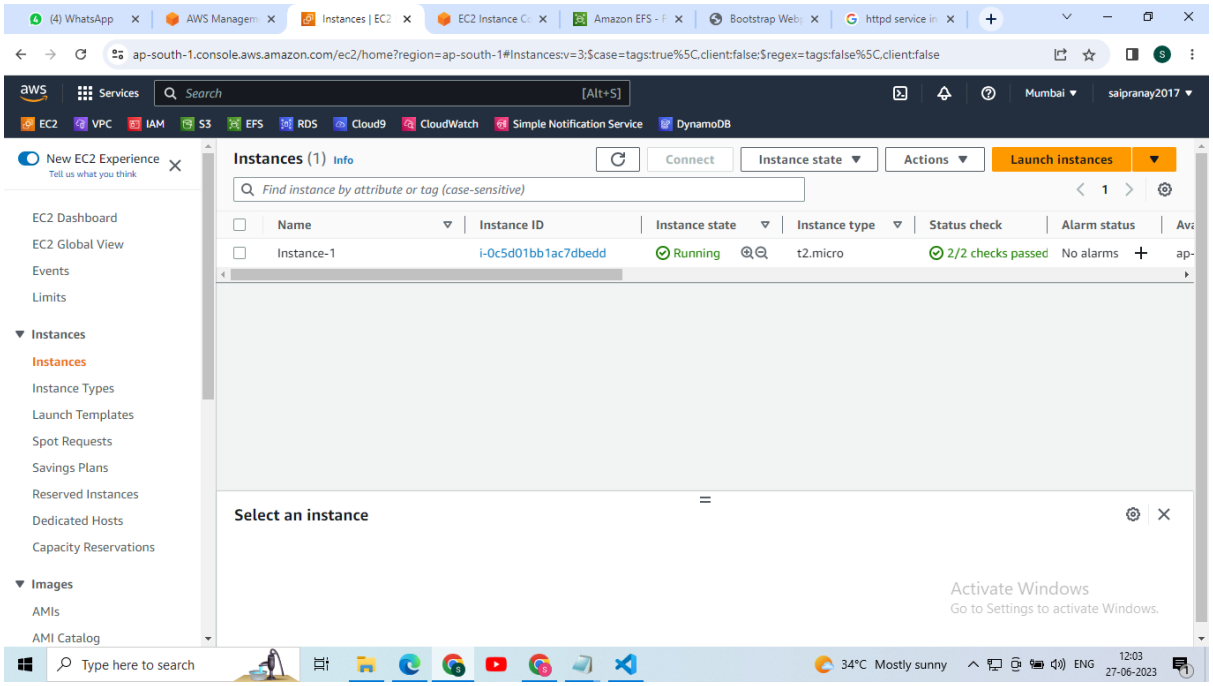
Giving name to the instance and choosing the AMI



Selecting the Custom VPC - the VPC we created in the above steps
Selecting the subnet-1 present in the availability zone 1a
Creating a new security group called - vpcsg



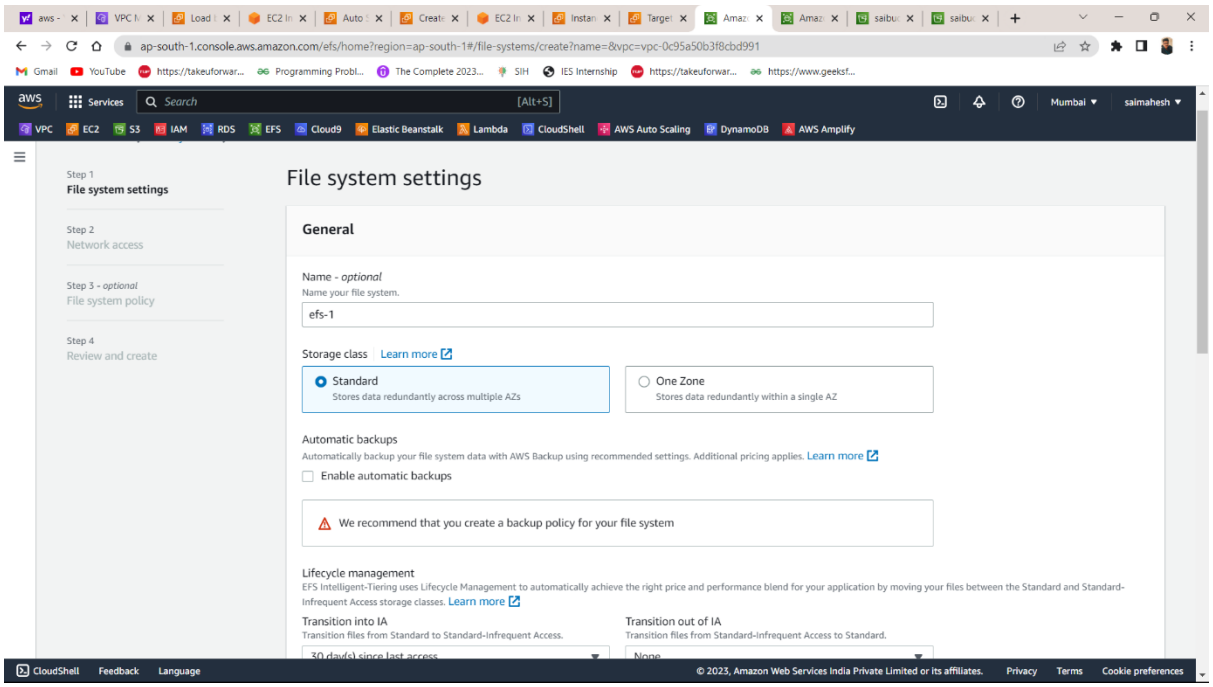
Adding rules:
1. SSH
2. All traffic
3. NFS - to mount the Elastic File System
4. HTTP - to host the web page



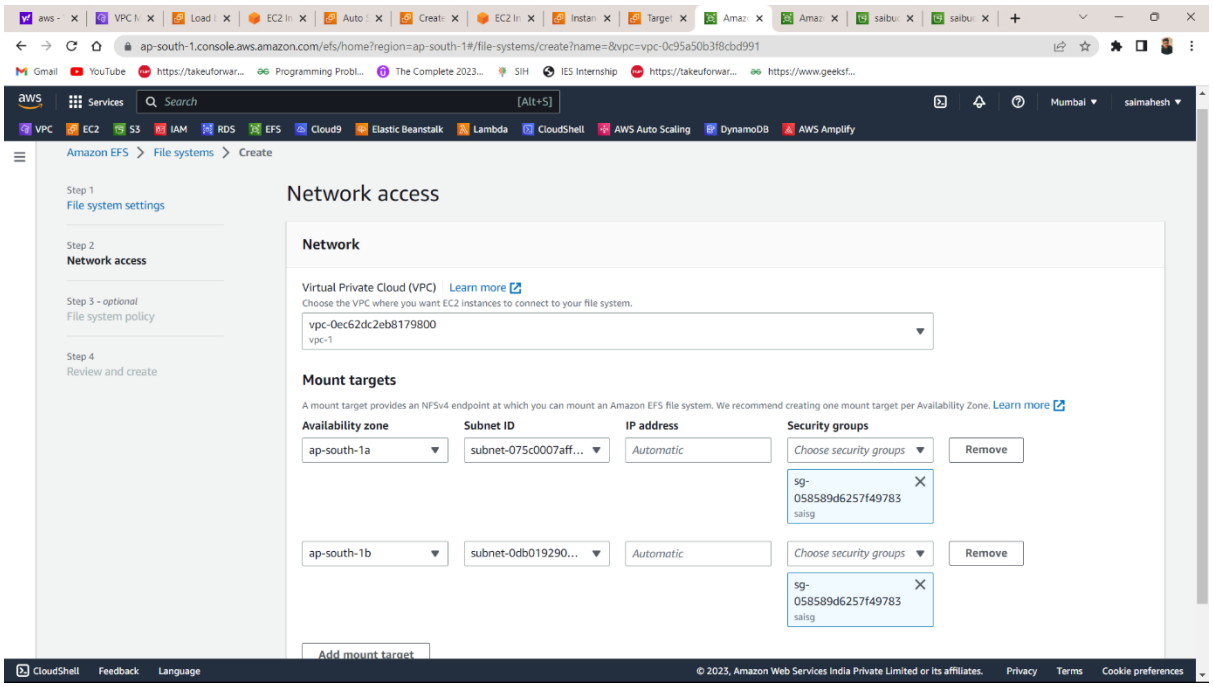
The instance is created

EFS:

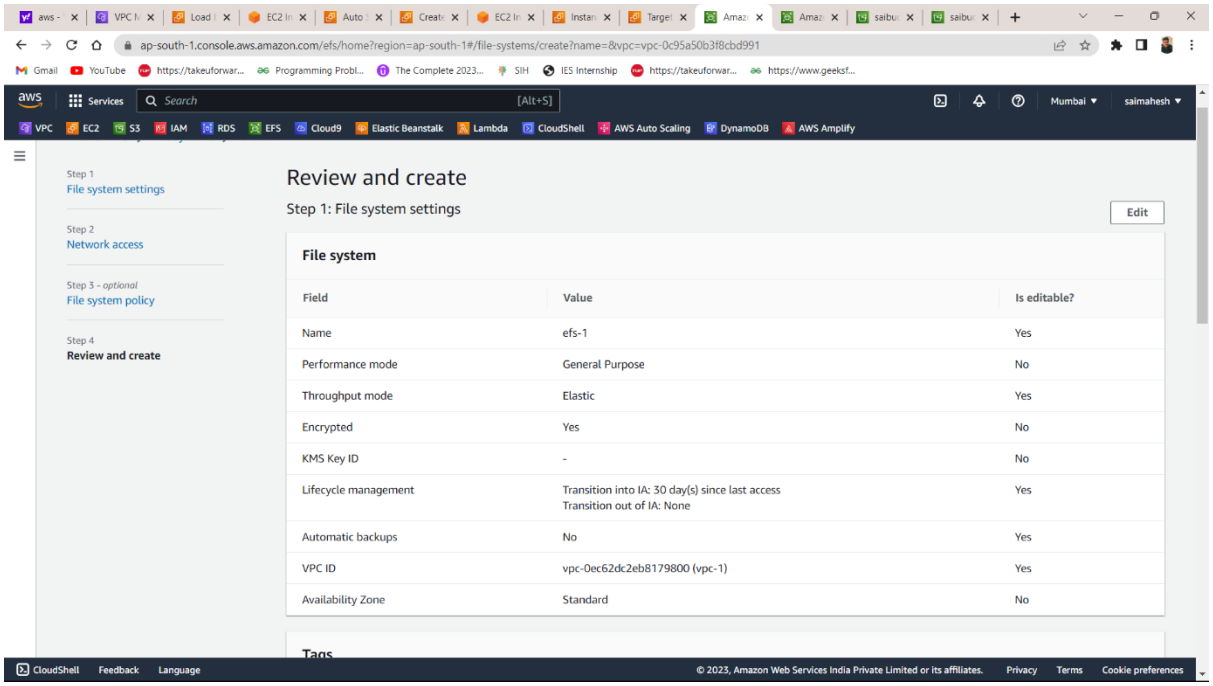
First we create name of efs and rest constrains as default.



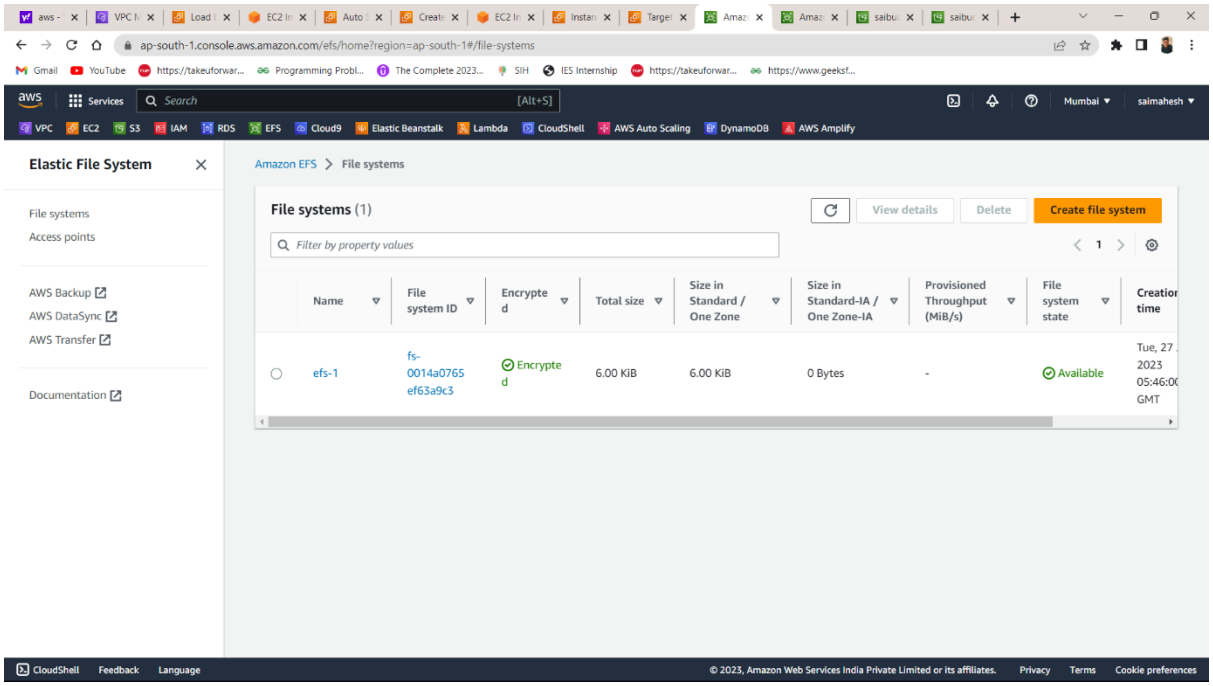
Now we Select Network accsess i.e VPC and Mount Target.



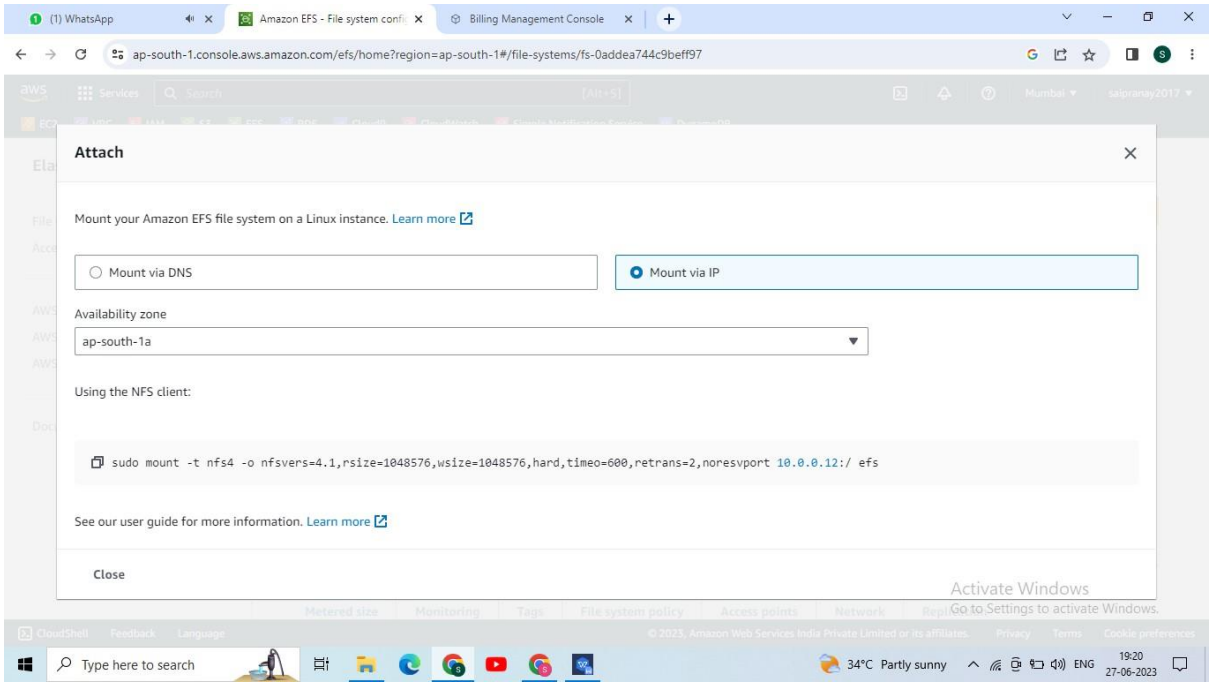
We selected policies as default and we create EFS.



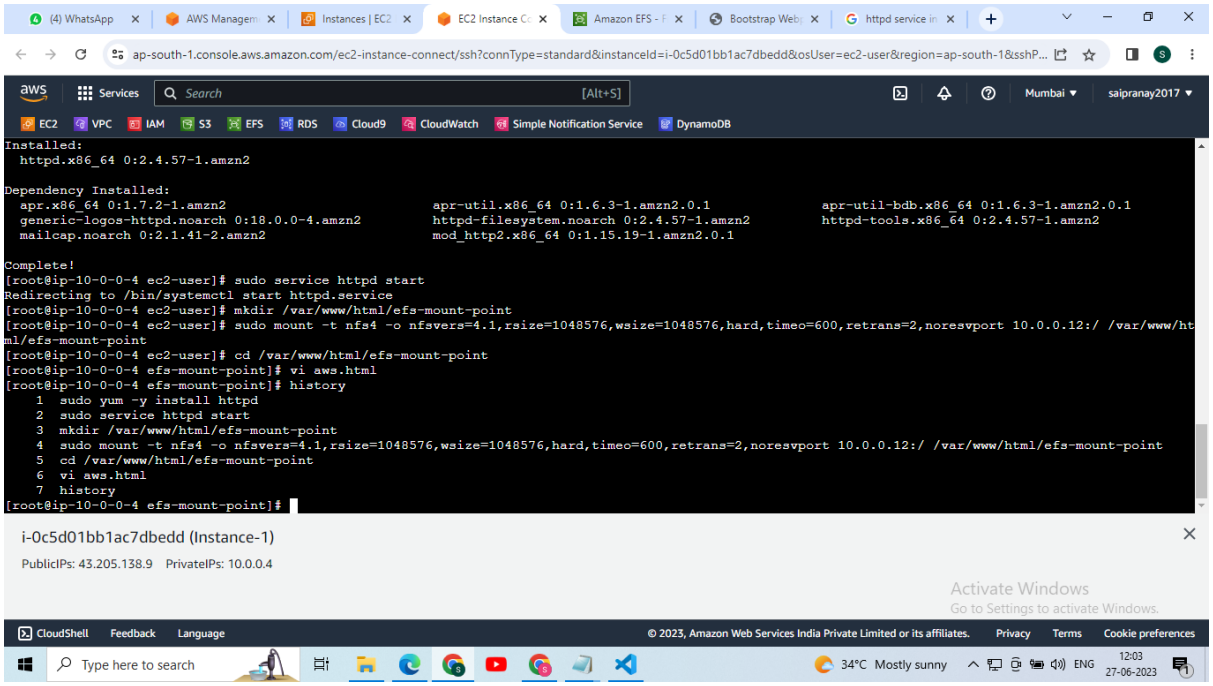
We Created EFS .



NFS url is copied from EC2 to mount with EFS

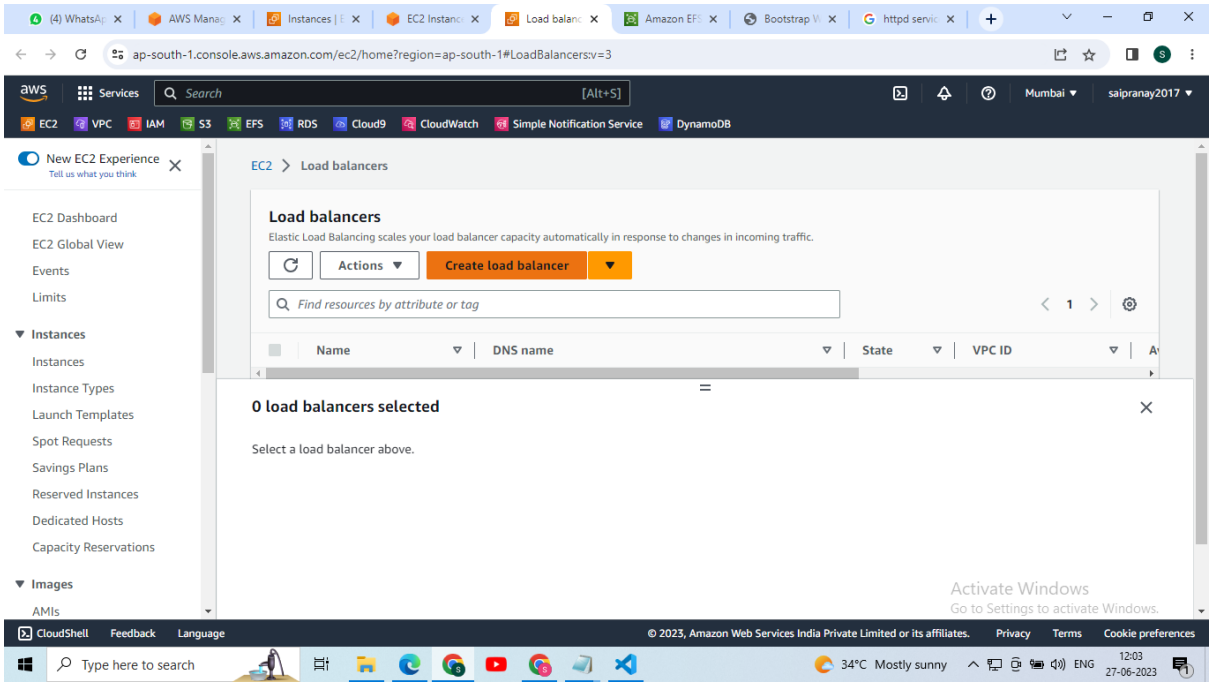


Mounting File System on EC2:

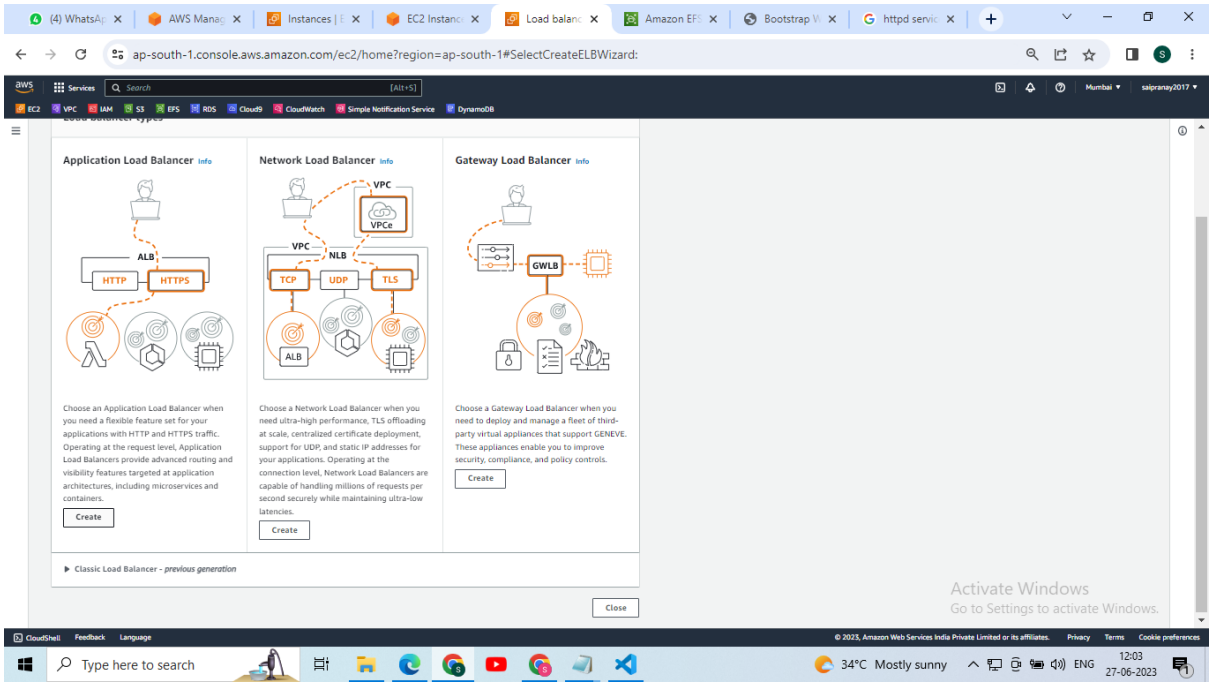


Using the above commands :
we can mount the file system we created to the ec2 instance.
Now a HTML file aws.html is created which is present in the file system we created.

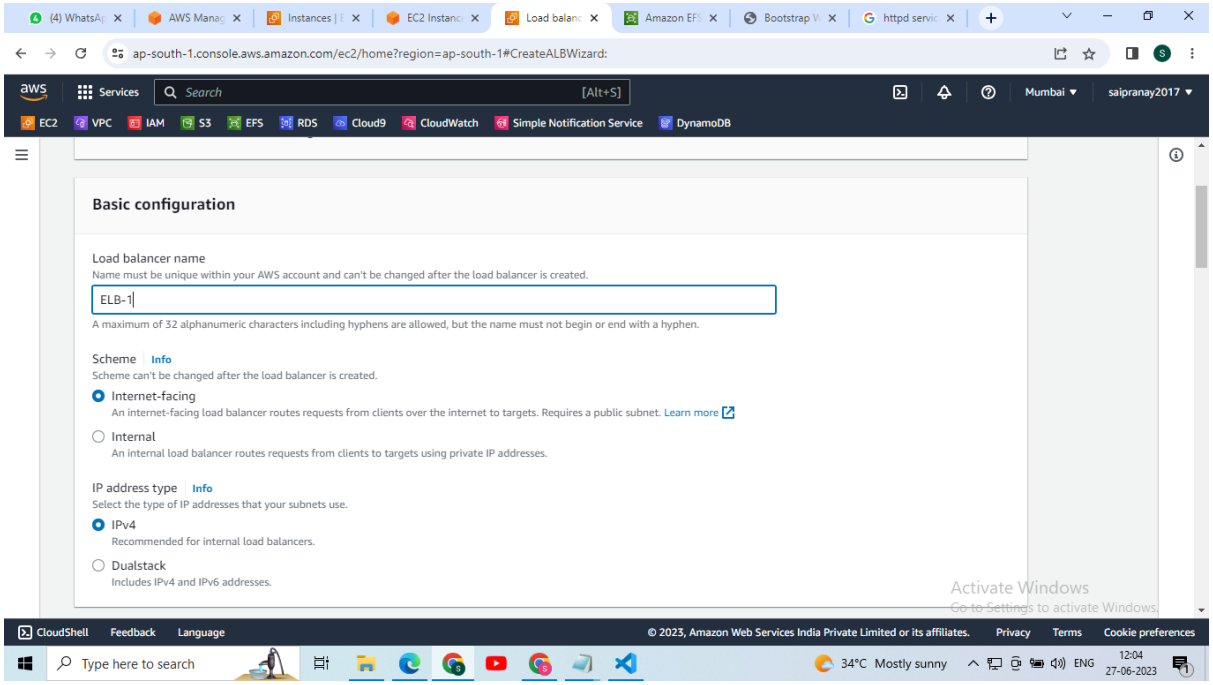
Load Balancer:



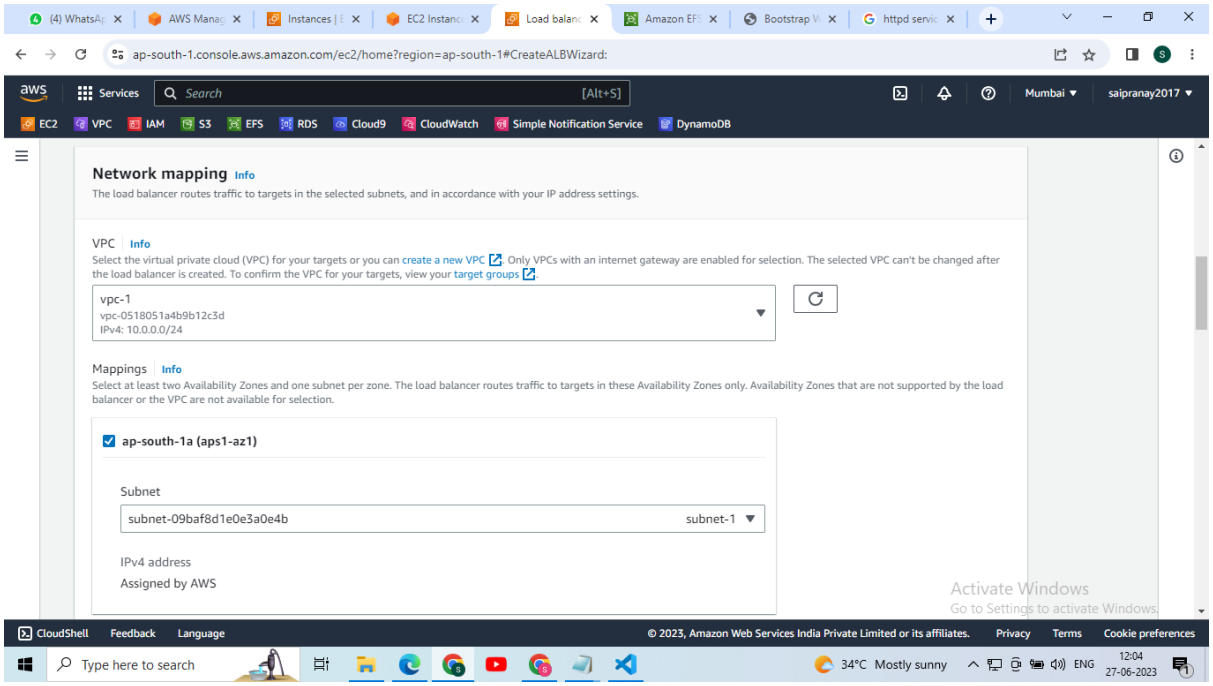
Now creating a load balancer.



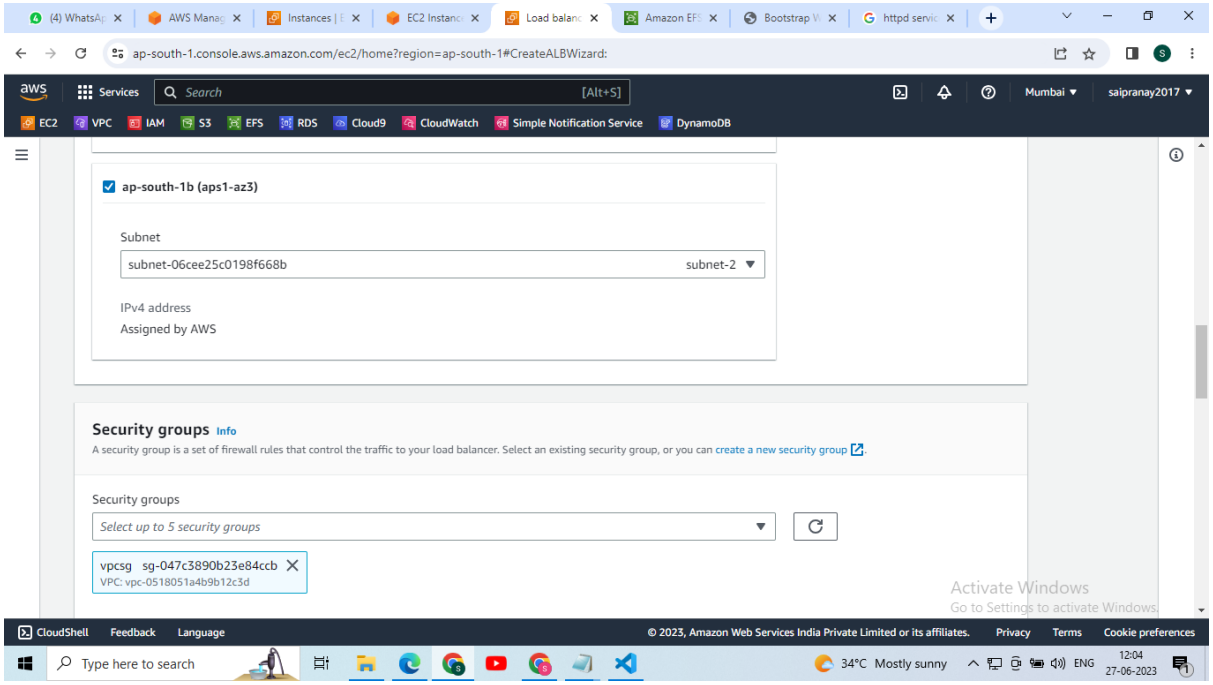
Creating an application load balancer as it is an elastic load balancer. It can divide the requests proportionally for also the new instances created by the auto scaling group.



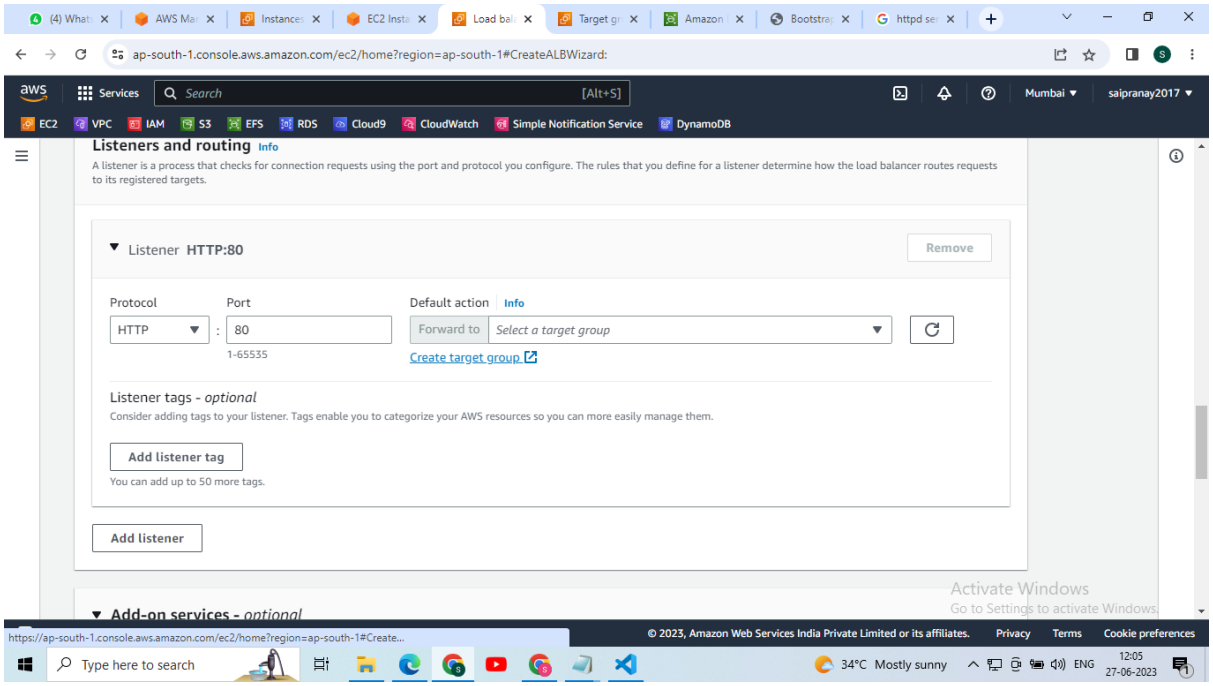
Giving name to the application load balancer



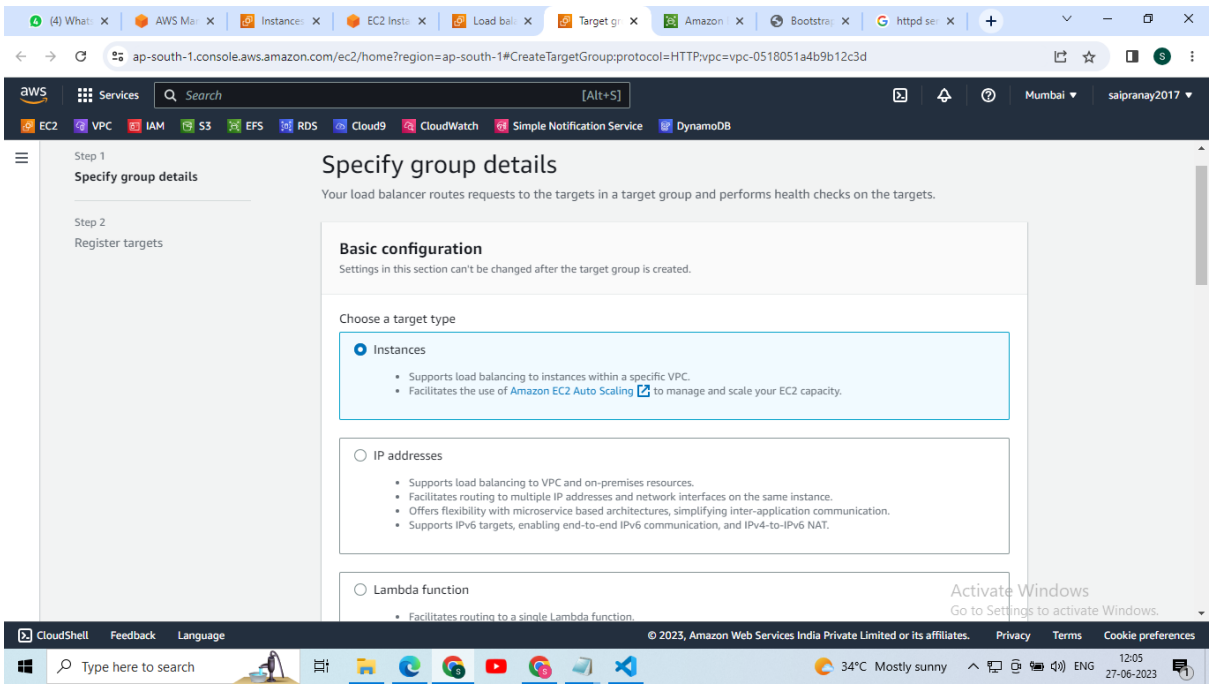
Selecting the custom VPC, sub-nets.



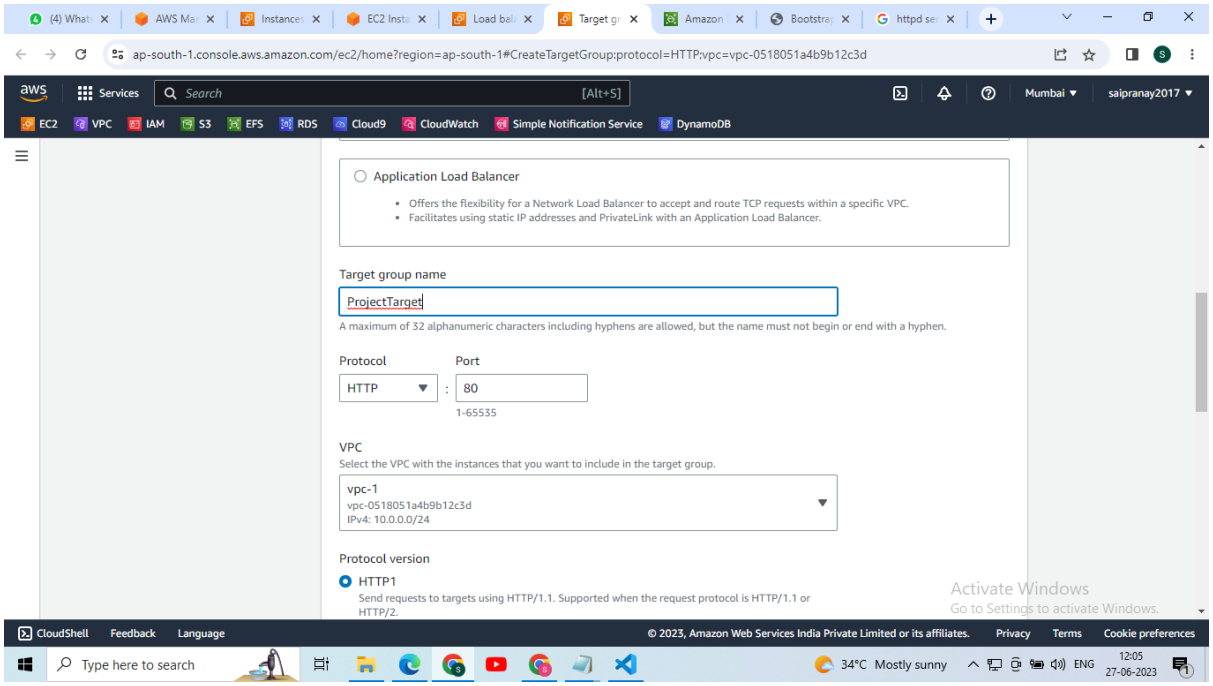
Selecting the security group



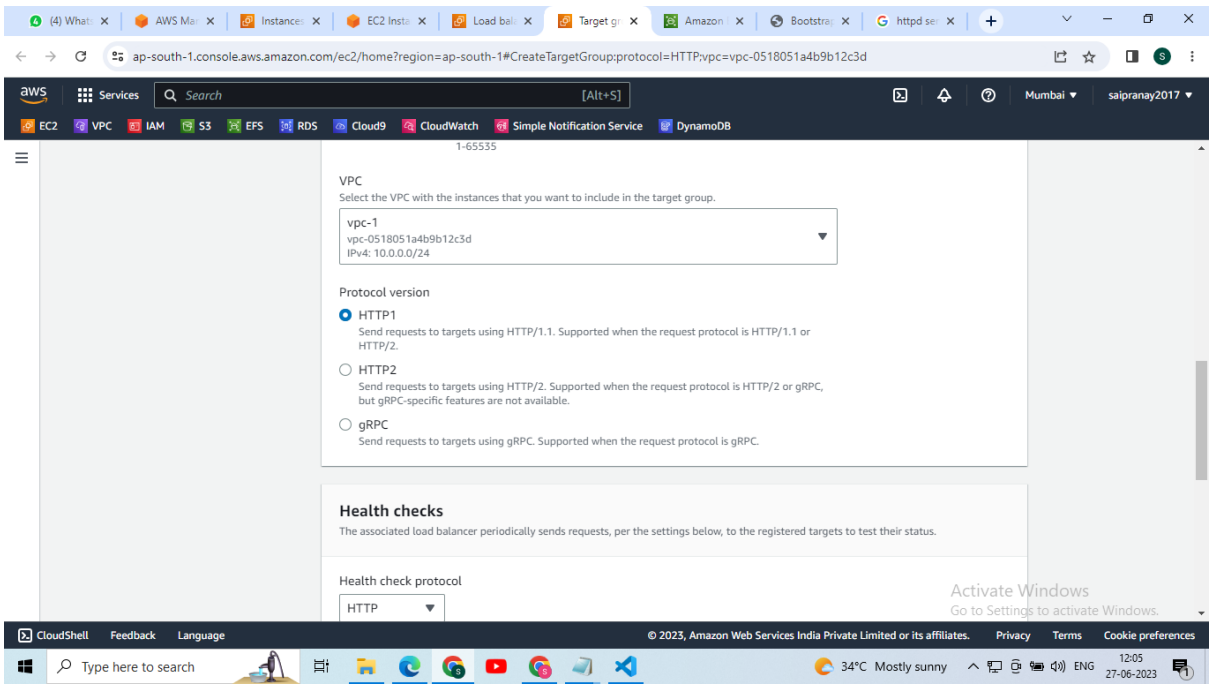
Creating a target group

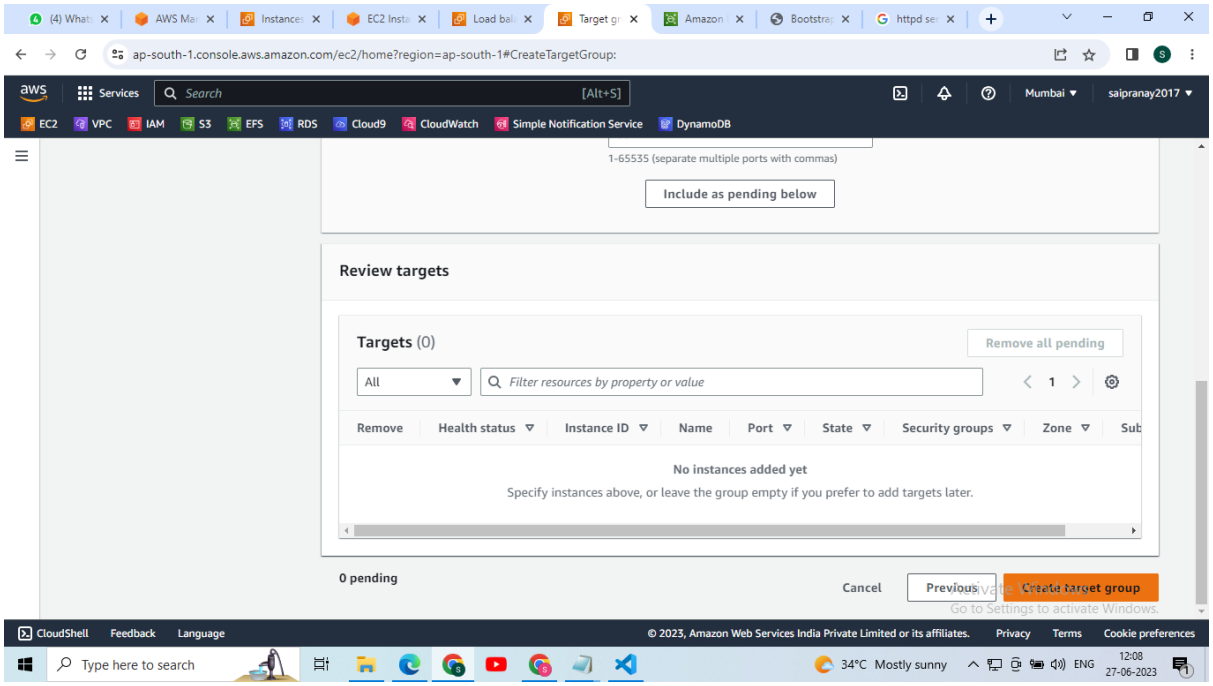


Selecting the target type - Instances

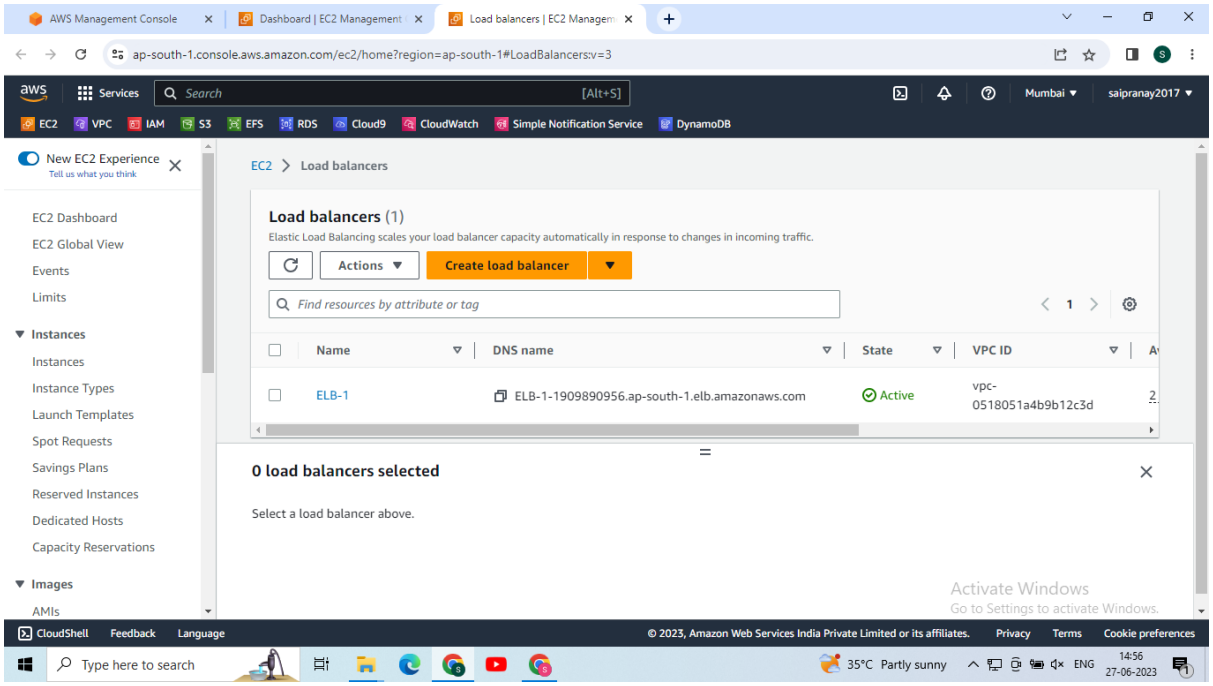


Giving the target group name
Selecting the custom VPC



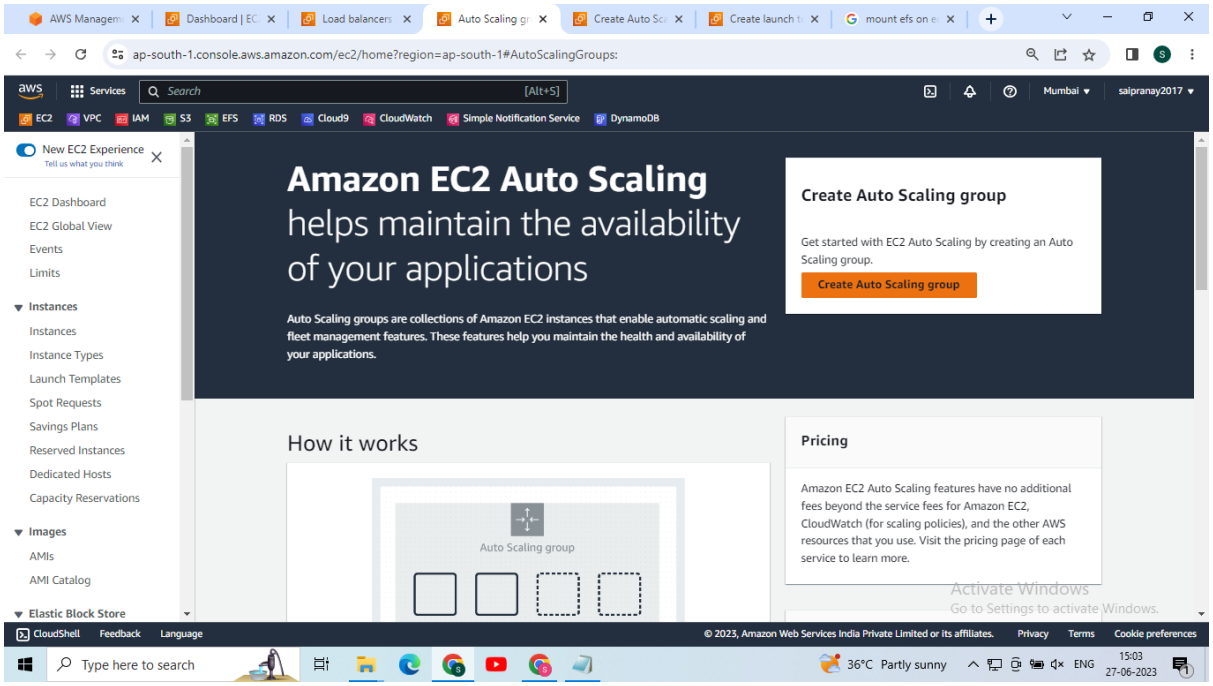


Now creating the target group .

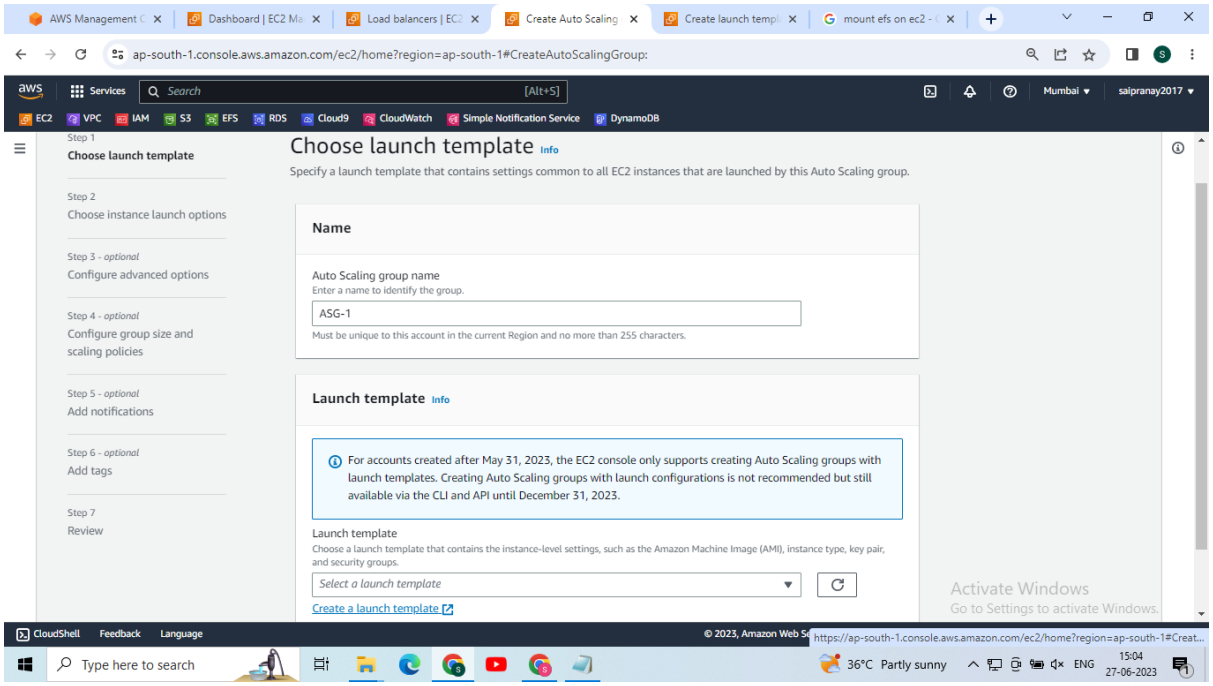


The application load balancer is created now.

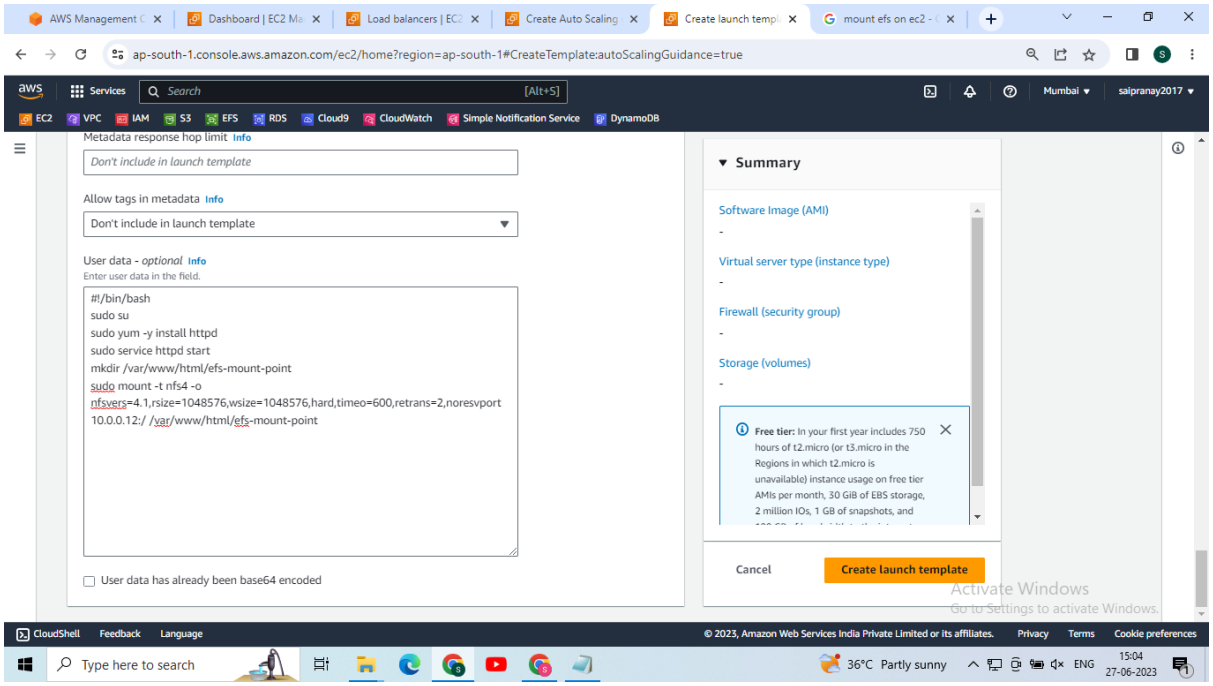
Auto Scaling Group:



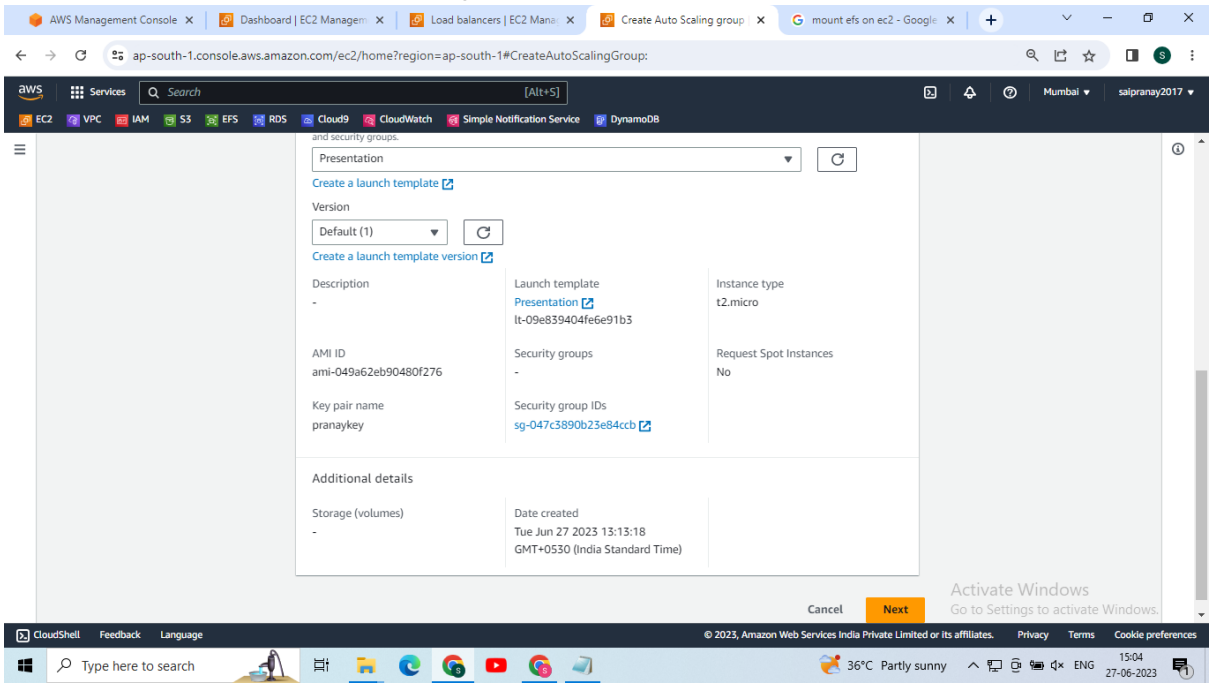
Now creating an auto scaling group to launch new instances.



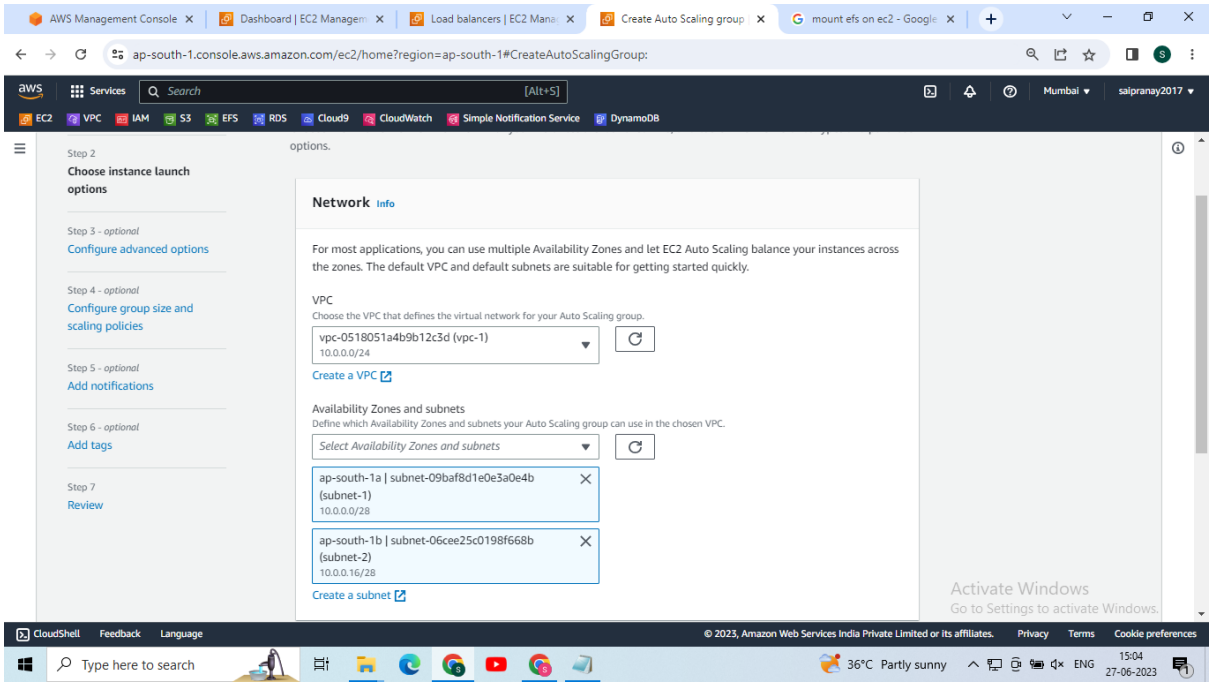
Give the autoscaling group name
Now giving a template to launch an instance.



In the user data section of launching a template the COMMANDS are given so that when the ASG launches an instance, the instance is directly mounted to the Elastic File System.

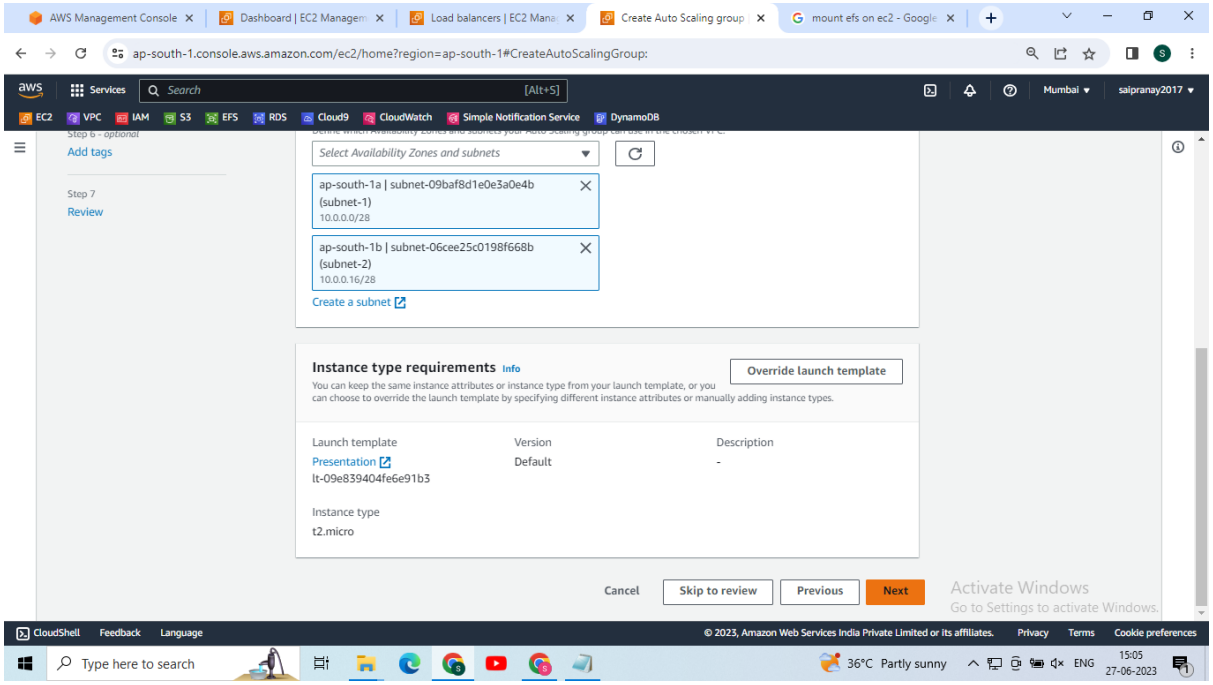


Choosing the instance template

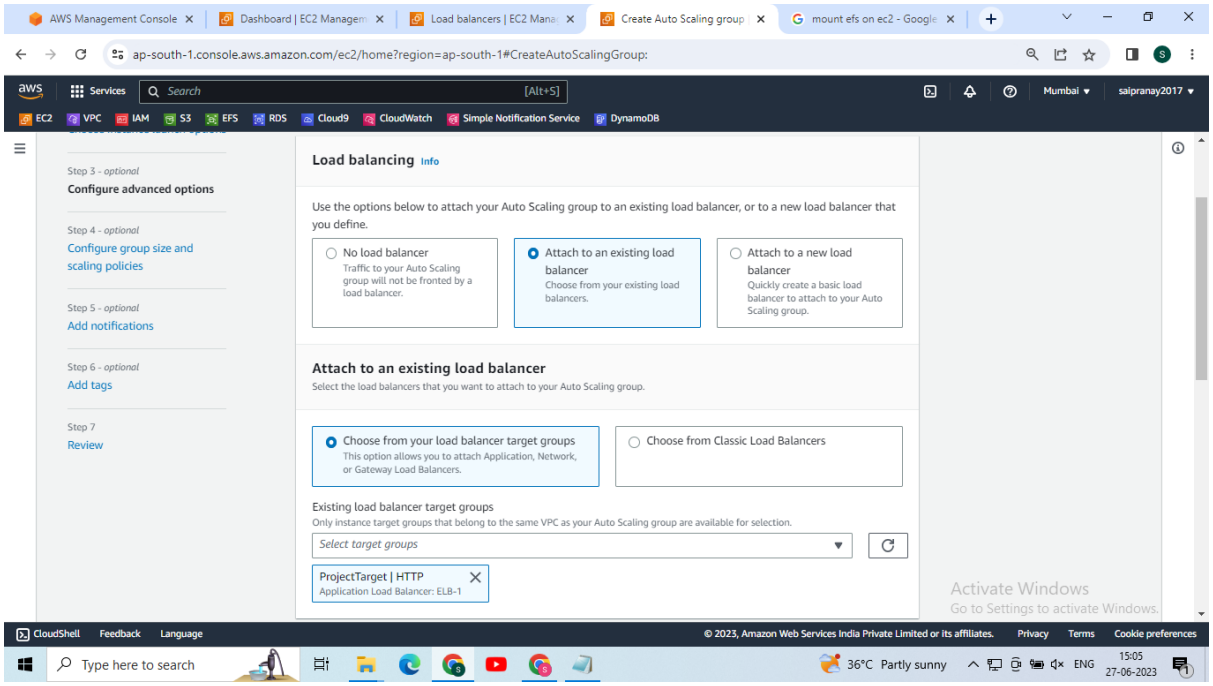


Selecting the custom VPC

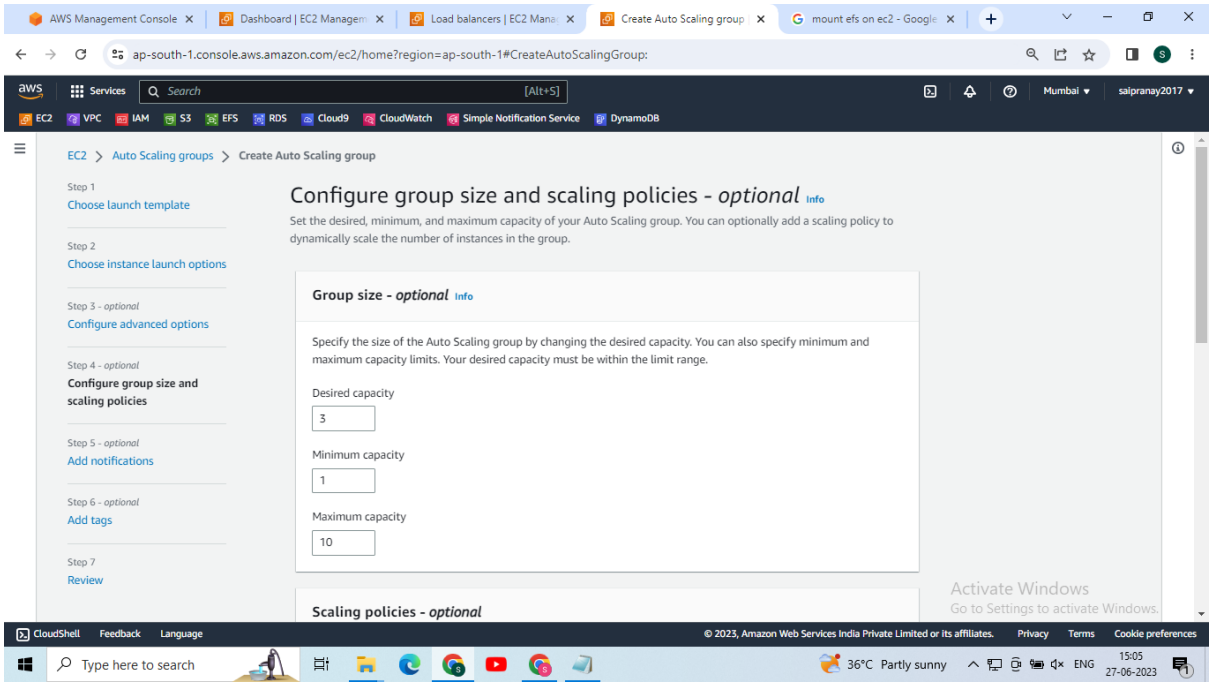
Selecting the Subnets present in different availability zones



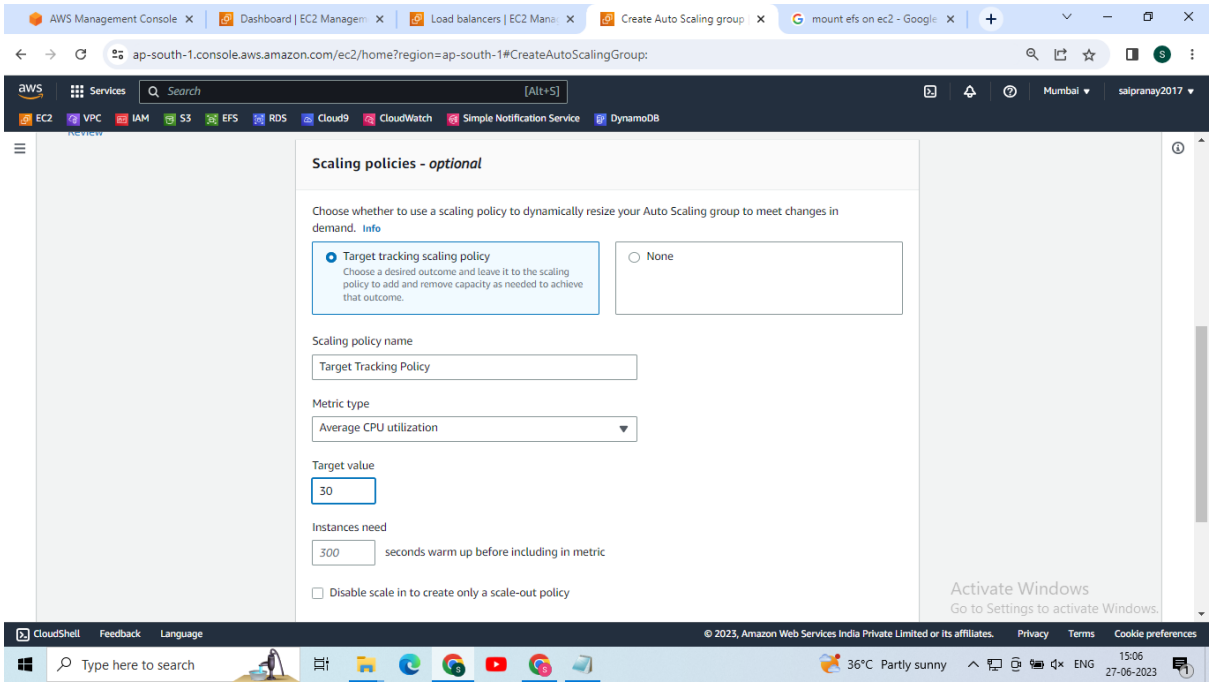
Click on Next



Attaching the auto scaling group to the elastic load balancer



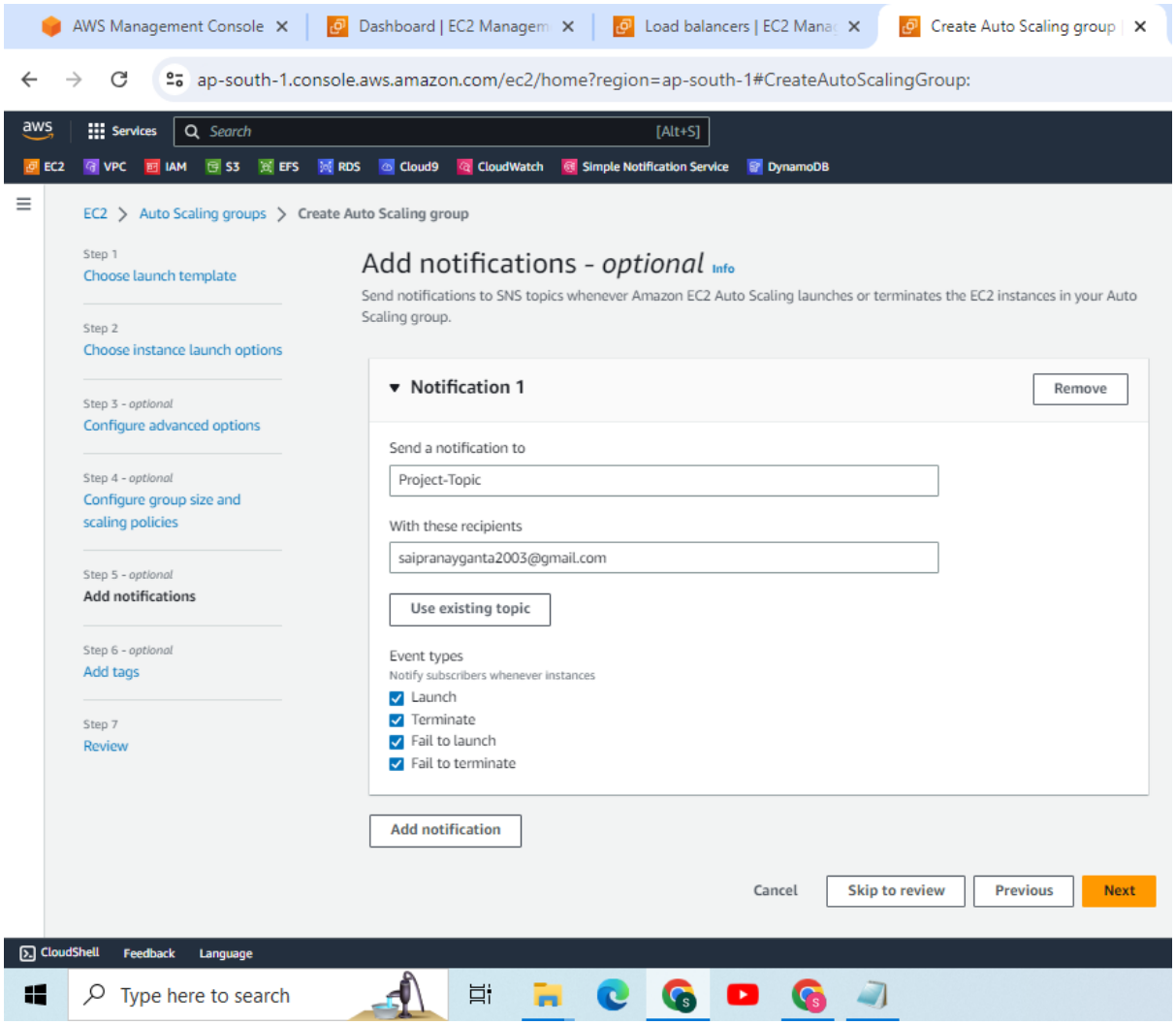
Giving the scaling policies
-Different capacity of instances requirements



Selecting a target Scaling policy
And click on next

#check carefully a photo removed here by me

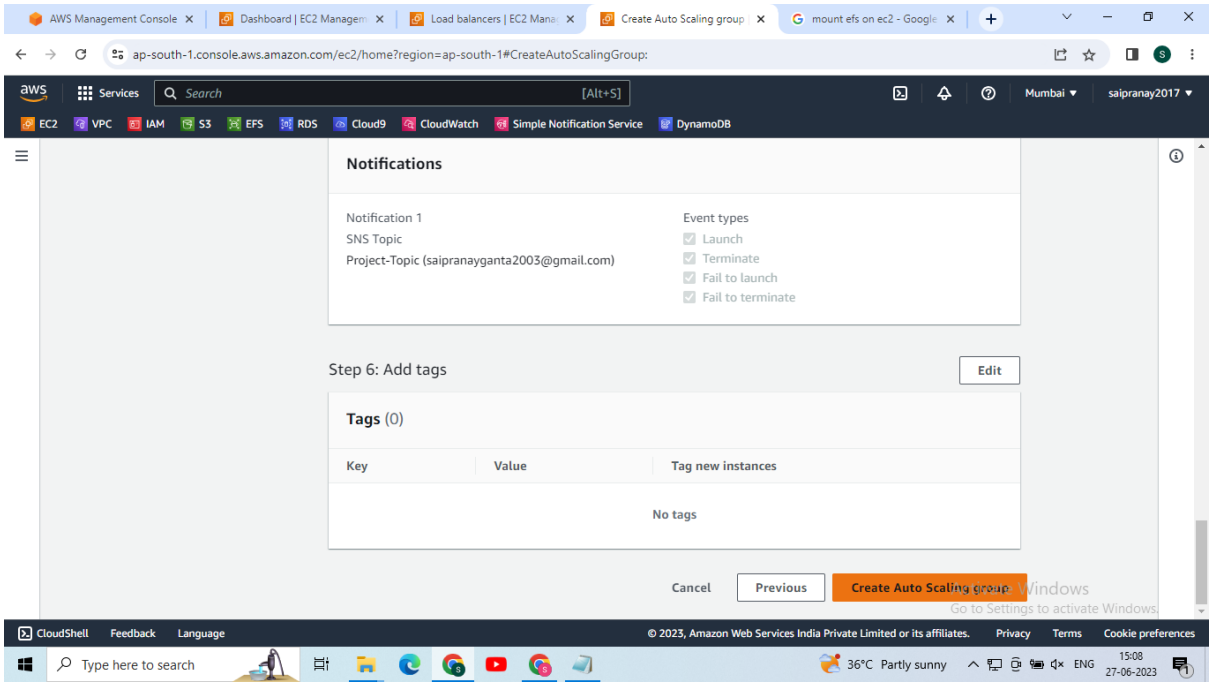
SNS :



Creating a topic

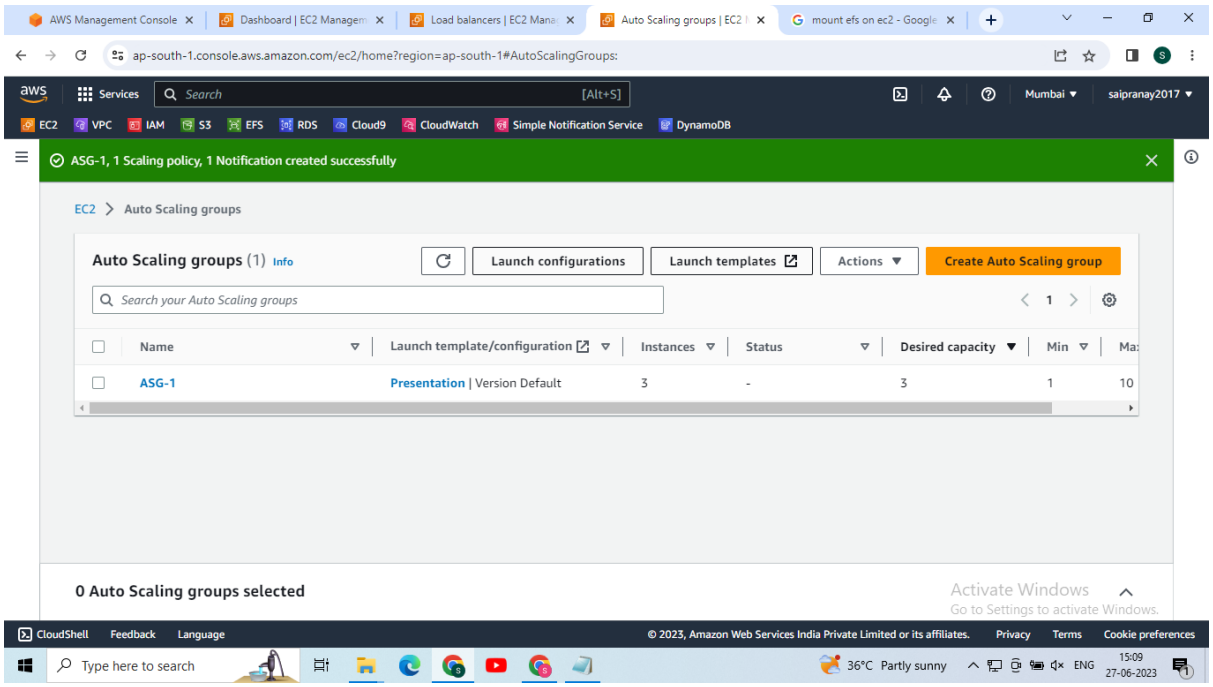
- Giving the topic name

- Giving an email add

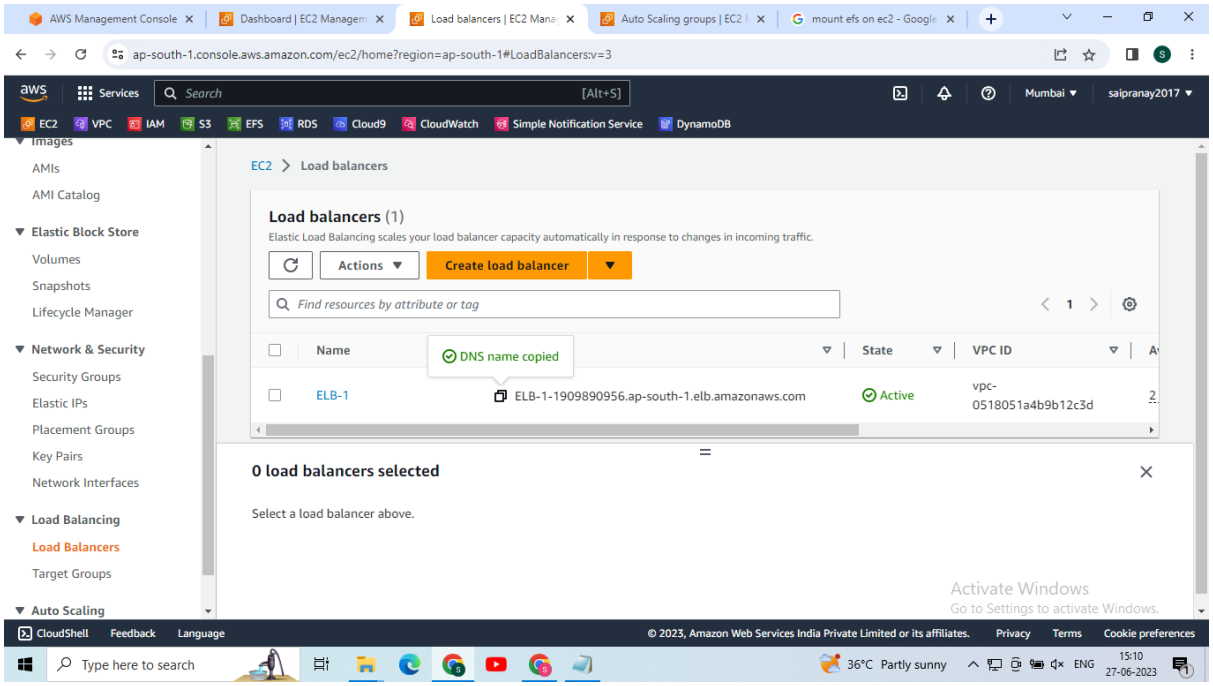


rest to whom the notifications should be received

Reviewing the selected conditions ans click on create

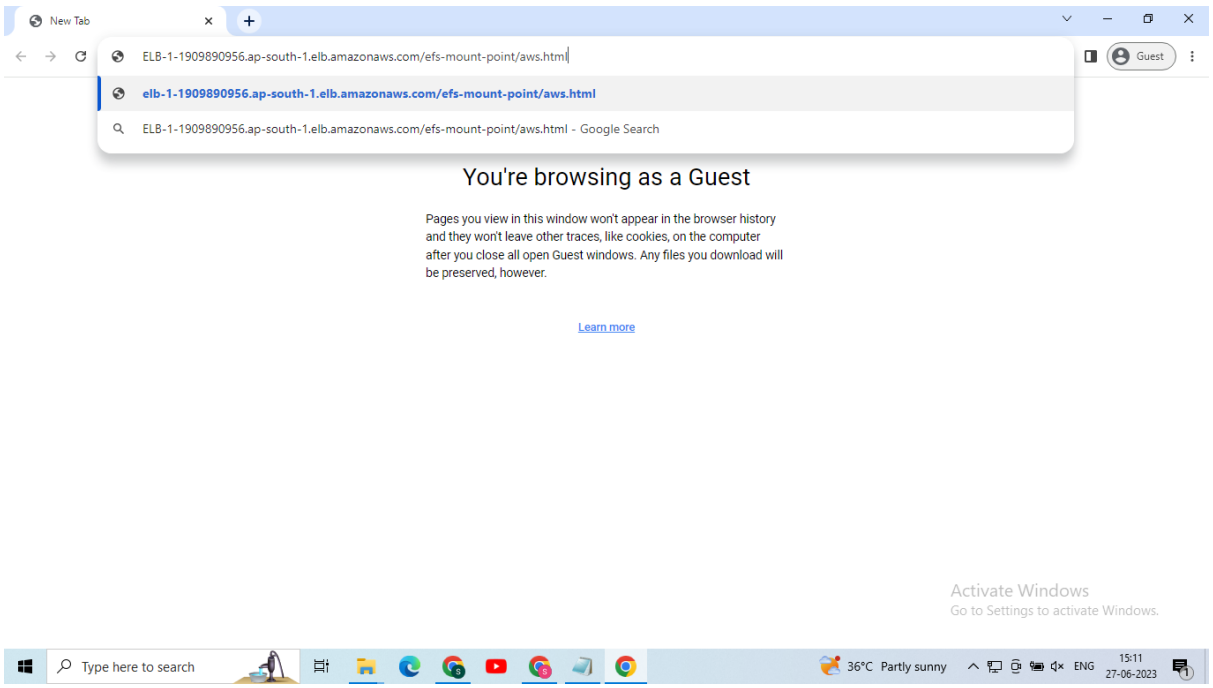


Now the auto-scaling group is created.

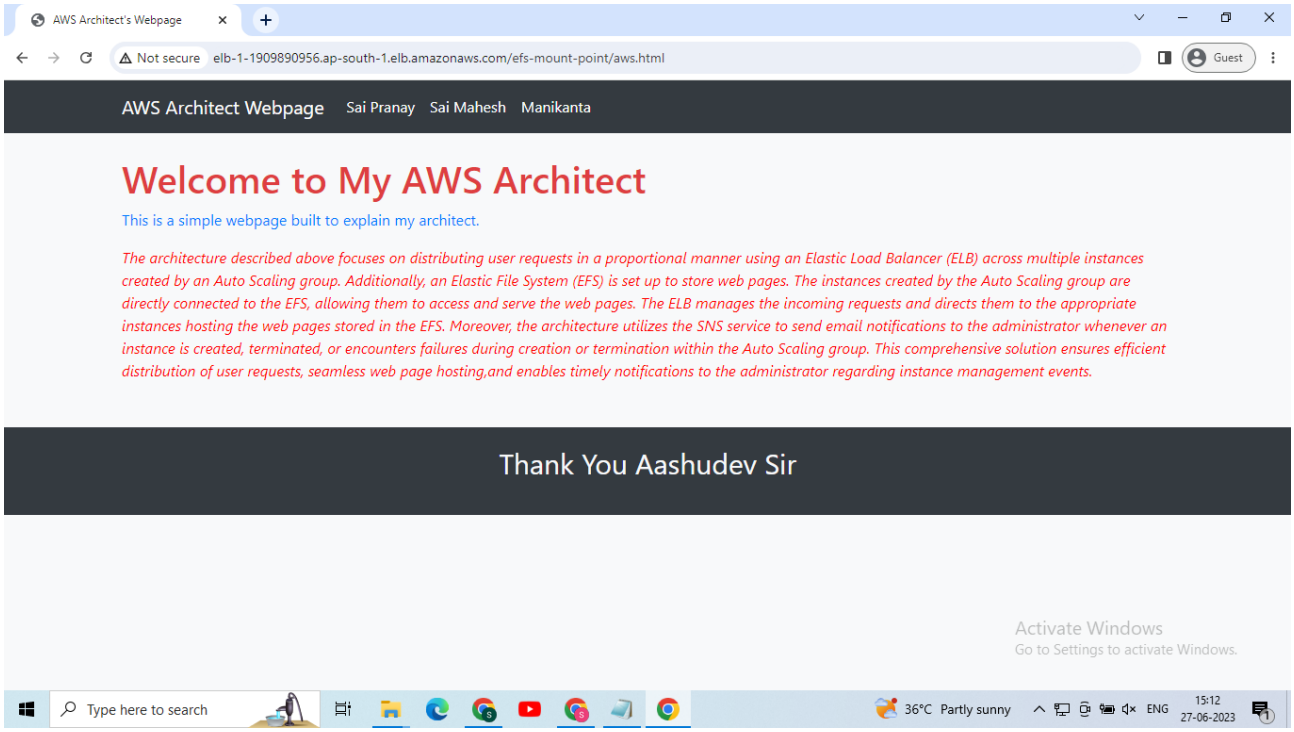


Now copy the DNS name of the Elastic load balancer.

Web Page Hosting :



Copy the link and add /efs-mount-point/aws.html
As the file is present in the EFS



Finally the web page is hosted :)