BLACKBUCKS INTERNSHIP REPORT

LANGUAGE TRANSLATOR AND TEXT TO SPEECH

SUBMITTED BY

TAMMALI AMULYA (21B91A54H0)

PURILLA PALLAVI PRIYA (21B91A6147)

UNDER THE GUIDANCE OF MR. GURU SANTHOSH SEENIVASAN

Blackbuck Engineers Pvt. Ltd

Road no 36, jubilee hills, Hyderabad

Title: Language Translator and Text to Speech Convertor.

Abstract:

As some countries conduct the different international business and board meetings in their indigenous languages which is not familiar to other country people. Here the problem raised about language. So we came up with a automated language translator and text to speech convertor.

The main aim of this project is to convert text given in one language into text of another specified language and to convert the text into speech (audio) by using aws services such as Amazon polly, Amazon translate , EC2 and IAM.

Initially create IAM user. create role and policy by giving amazon polly, amazon translate access. add policy to user. Now create EC2 instance. Attach the role to the instance. Connect the instance to terminal and write the html code by connecting to instance region and providing user permissions to instance using access key and other requirements. Now test the project by using public IP of instance.

By using this project we can convert text into required form. This project helps in spreading new information, knowledge and ideas across the world. It also enhances the international marketing and expands the business.

Team members:

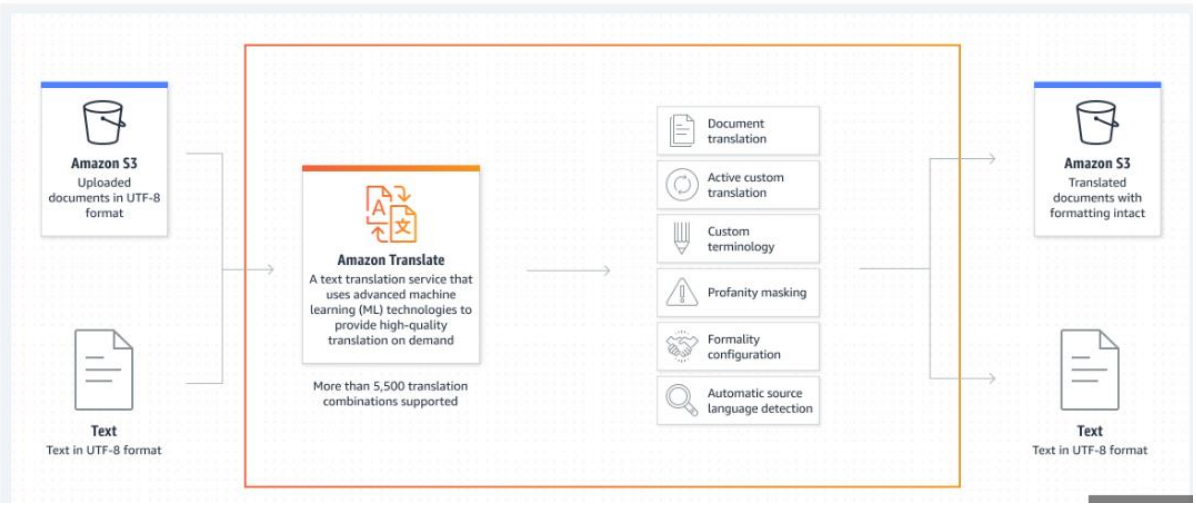
Tammali Amulya (21B91A54H0)

Purilla Pallavi Priya (21B91A6147)

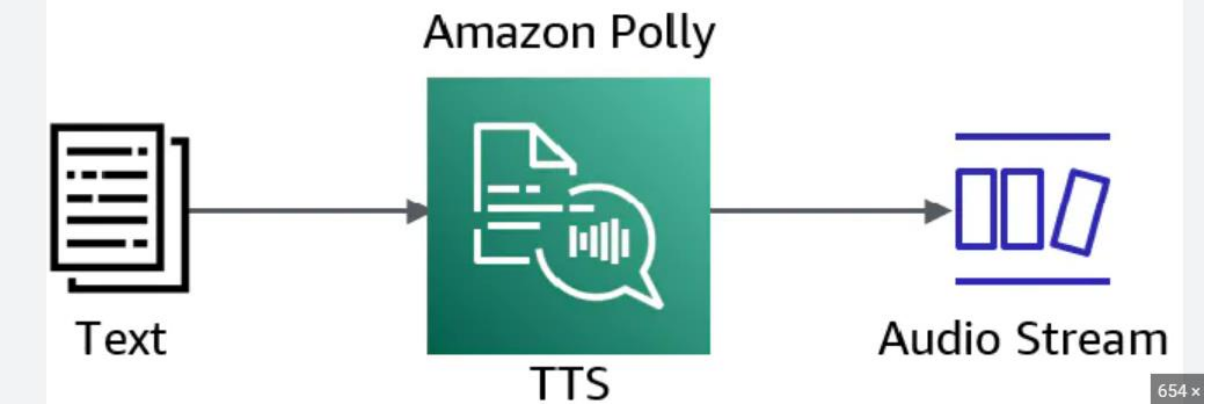
TABLE OF CONTENTS

Service used: **EC2, IAM, Amazon polly, Amazon Translate.**

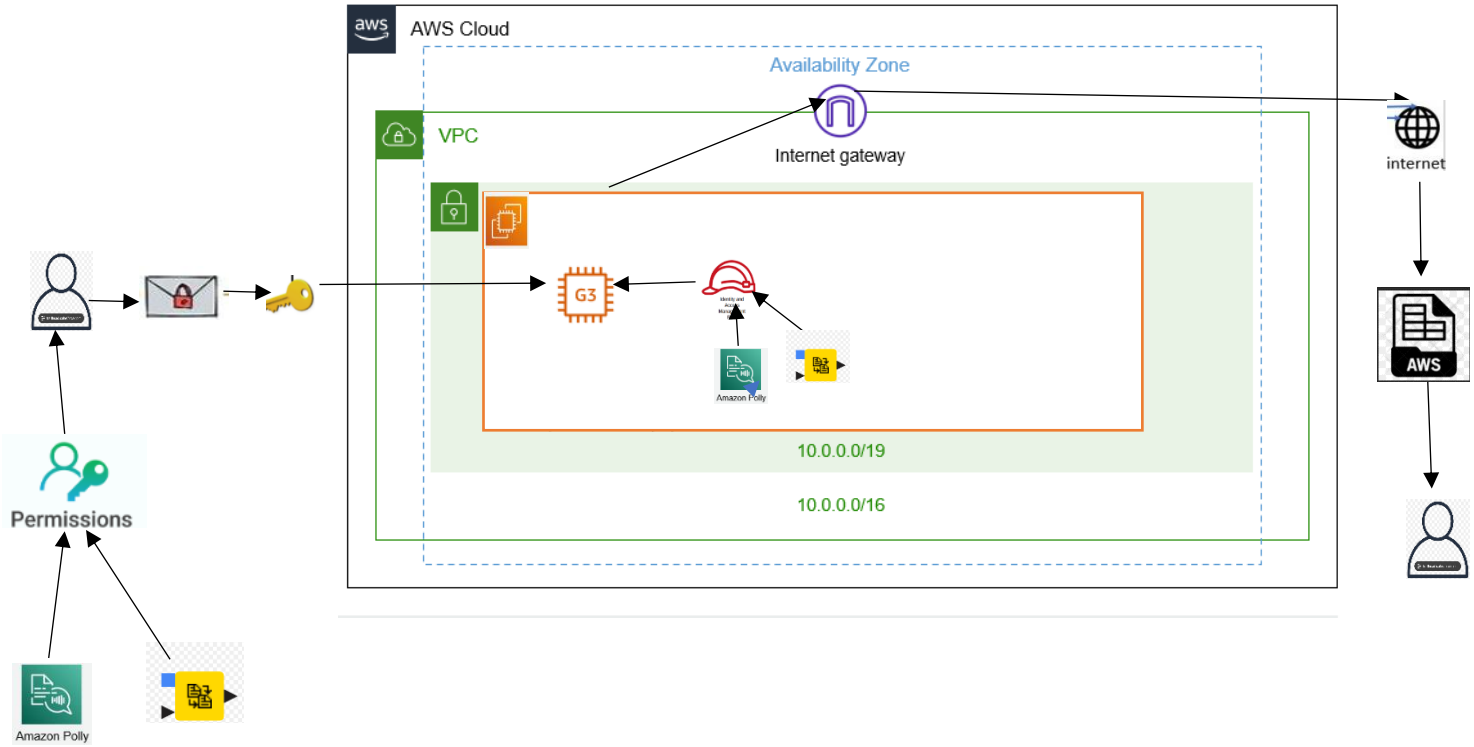
ARCHITECTURES:



Rough architecture for language translator



Rough architecture for text to speech



final architecture of language translator and text to speech

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 onpremises. 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, payas-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 SaaSbased 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

• Amazon Kinesis

• 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. 12

• 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 RDS:

Amazon Relational Database Service (or Amazon RDS) is a distributed relational database service by Amazon Web Services (AWS). It is a web service running "in the cloud" designed to simplify the setup, operation, and scaling of a relational database for use in applications. Administration processes like patching the database software, backing up databases and enabling point-in-time

recovery are managed automatically. Scaling storage and compute resources can be performed by a single API call to the AWS control plane on-demand. AWS does not offer an SSH connection to the underlying virtual machine as part of the managed service.

Multiple Availability Zone (AZ) Deployment

In May 2010 Amazon announced Multi-Availability Zone deployment support. Amazon RDS Multi-Availability Zone (AZ) allows users to automatically provision and maintain a synchronous physical or logical "standby" replica, depending on database engine, in a different Availability Zone (independent infrastructure in a physically separate location). Multi-AZ database instance can be developed at creation time or modified to run as a multi-AZ deployment later. Multi-AZ deployments aim to provide enhanced availability and data durability for MySQL, MariaDB, Oracle, PostgreSQL and SQL Server instances and are targeted for production environments. In the event of planned database maintenance or unplanned service disruption, Amazon RDS automatically fails over to the up-to-date standby, allowing database operations to resume without administrative intervention. Multi-AZ RDS instances are optional and have a cost associated with them. When creating a RDS instance, the user is asked if they would like to use a multi-AZ RDS instance. In Multi AZ RDS deployments backups are done in the standby instance so I/O activity is not suspended any time, but users may experience elevated latencies for a few minutes during backups.

Read replicas.

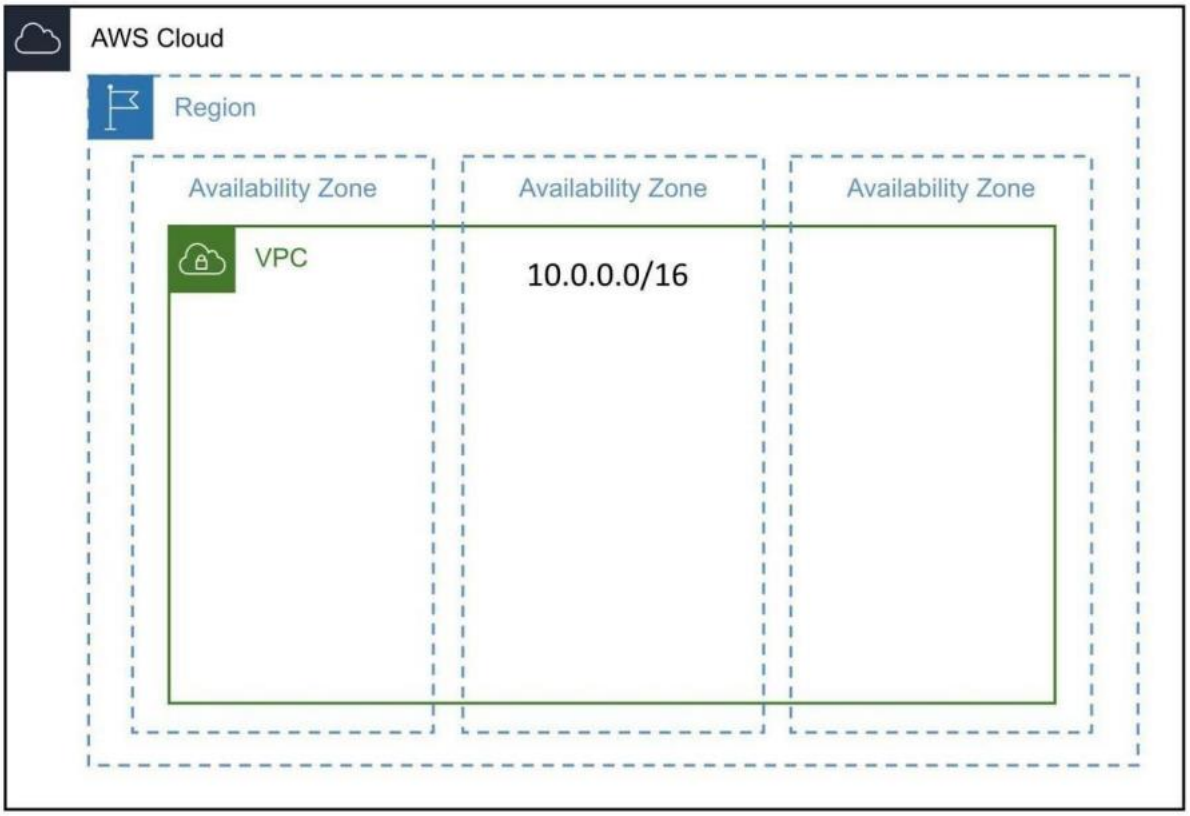
Read replicas allow different use cases such as scale in for read-heavy database workloads. There are up to five replicas available for MySQL, MariaDB, and PostgreSQL. Instances use the native, asynchronous replication functionality of their respective database engines. They have no backups configured by default and are accessible and can be used for read scaling. MySQL and MariaDB read replicas and can be made writeable again since October 2012; PostgreSQL read replicas do not support it. Replicas are done at database instance level and do not support replication at database or table level.

Performance metrics and monitoring

Performance metrics for Amazon RDS are available from the AWS Management Console or the Amazon CloudWatch API. In December 2015, Amazon announced an optional enhanced monitoring feature that provides an expanded set of metrics for the MySQL, MariaDB, and Aurora database engines.

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.

Amazon S3:

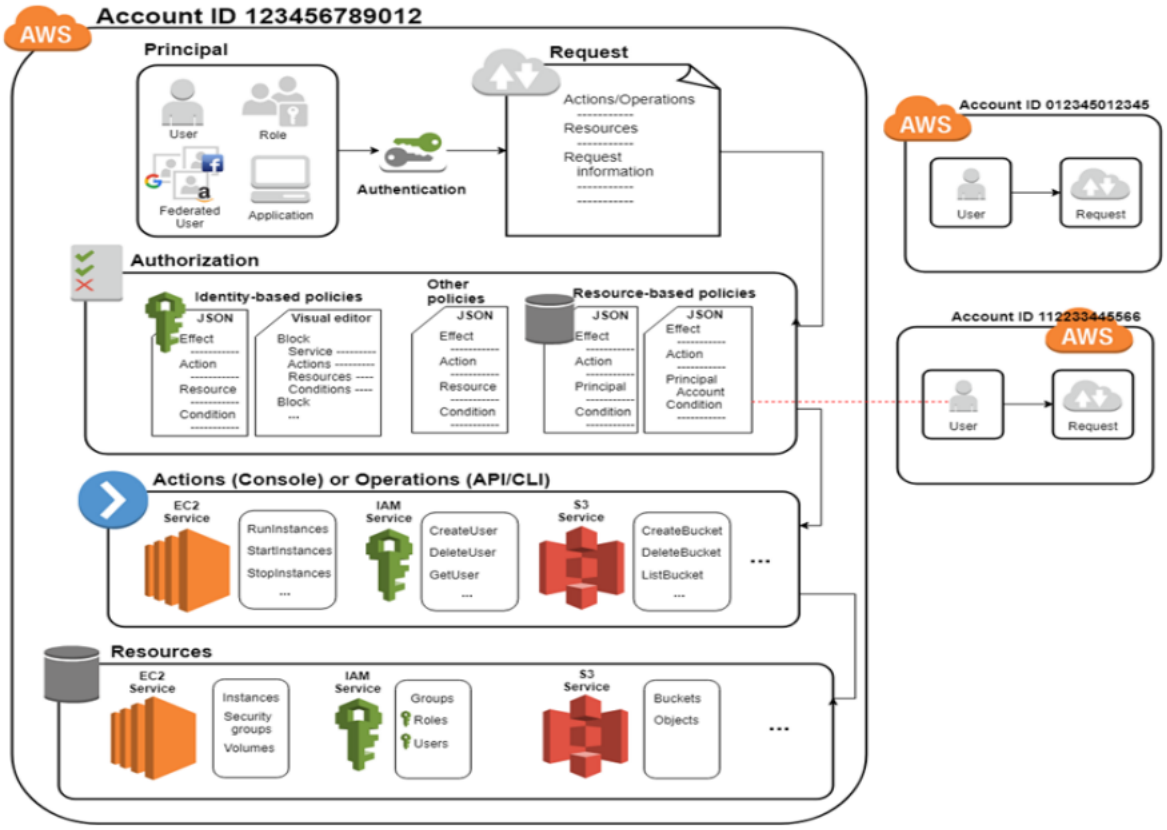
Amazon S3 manages data with an object storage architecture which aims to provide scalability, high availability, and low latency with high durability. The basic storage units of Amazon S3 are objects which are organized into buckets. Each object is identified by a unique, userassigned key. Buckets can be

managed using the console provided by Amazon S3, programmatically with the AWS SDK, or the REST application programming interface.

Objects can be up to five terabytes in size. Requests are authorized using an access control list associated with each object bucket and support versioning which is disabled by default. Since buckets are typically the size of an entire file system mount in other systems, this access control scheme is very coarse-grained. In other words, unique access controls cannot be associated with individual files. [citation needed] Amazon S3 can be used to replace static web-hosting infrastructure with HTTP client-accessible objects, index document support and error document support.

Amazon IAM:

IAM provides the infrastructure necessary to control authentication and authorization for your AWS account. The IAM infrastructure is illustrated by the following diagram.



First, a human user or an application uses their sign-in credentials to authenticate with AWS. Authentication is provided by matching the sign-in credentials to a principal (an IAM user, federated user, IAM role, or application) trusted by the AWS account.

Next, a request is made to grant the principal access to resources. Access is granted in response to an authorization request. For example, when you first sign into the console and are on the console home page, you are not accessing a specific service. When you select a service, the request for authorization is sent to that service and it looks to see if your identity is on the list of authorized users, what policies are being enforced to control the level of access granted, and any other policies that might be in effect. Authorization requests can be made by principals within your AWS account or from another AWS account that you trust.

Once authorized, the principal can take action or perform operations on resources in your AWS account. For example, the principal could launch a new Amazon Elastic Compute Cloud instance, modify IAM group membership, or delete Amazon Simple Storage Service buckets.

The previous illustration we used specific terminology to describe how to obtain access to resources. These IAM terms are commonly used when working with AWS:

IAM Resources

The user, group, role, policy, and identity provider objects that are stored in IAM. As with other AWS services, you can add, edit, and remove resources from IAM.

IAM Identities

The IAM resource objects that are used to identify and group. You can attach a policy to an IAM identity. These include users, groups, and roles.

IAM Entities

The IAM resource objects that AWS uses for authentication. These include IAM users and roles.

Principals

A person or application that uses the AWS account root user, an IAM user, or an IAM role to sign in and make requests to AWS. Principals include federated users and assumed roles.

Human users

Also known as human identities; the people, administrators, developers, operators, and consumers of your applications.

Workload

A collection of resources and code that delivers business value, such as an application or backend process. Can include applications, operational tools, and components.

AWS Lambda:

AWS Lambda is a compute service that lets you run code without provisioning.

Lambda runs your code on a high-availability compute infrastructure and performs all of the administration of the compute resources, including server and operating system maintenance, capacity provisioning and automatic scaling, and logging. With Lambda, all you need to do is supply your code in one of the language runtimes that Lambda supports.

You organize your code into Lambda functions. The Lambda service runs your function only when needed and scales automatically. You only pay for the compute time that you consume— there is no charge when your code is not running.

When using Lambda, you are responsible only for your code. Lambda manages the compute fleet that offers a balance of memory, CPU, network, and other resources to run your code. Because Lambda manages these resources, you cannot log in to compute instances or customize the operating system on provided runtimes.

Lambda performs operational and administrative activities on your behalf, including managing capacity, monitoring, and logging your Lambda functions.

If you do need to manage your compute resources, AWS has other compute services to consider, such as:

• AWS App Runner builds and deploys containerized web applications automatically, load balances traffic with encryption, scales to meet your traffic needs, and allows for the configuration of how services are accessed and communicate with other AWS applications in a private Amazon VPC

. • AWS Fargate with Amazon ECS runs containers without having to provision, configure, or scale clusters of virtual machines.

• Amazon EC2 lets you customize operating system, network and security settings, and the entire software stack.

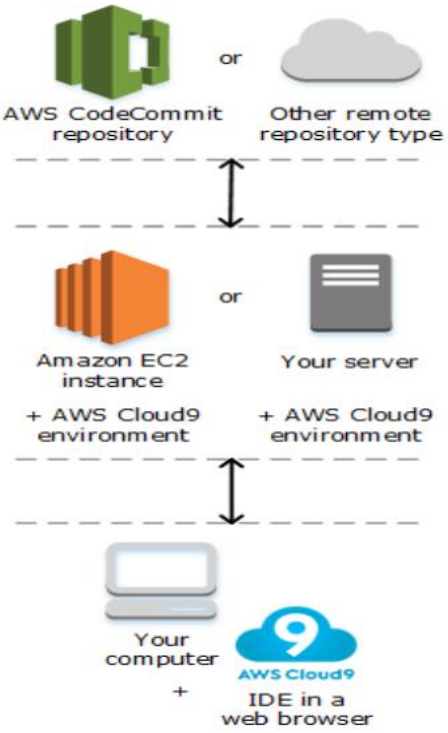
You are responsible for provisioning capacity, monitoring fleet health and performance, and using Availability Zones for fault tolerance. You can use environment variables to adjust your function's behavior without updating code. An environment variable is a pair of strings that is stored in a function's version-specific configuration. The Lambda runtime makes environment variables available to your code and sets additional environment variables that contain information about the function and invocation request.

AWS Cloud9:

AWS Cloud9 is an integrated development environment, or IDE.

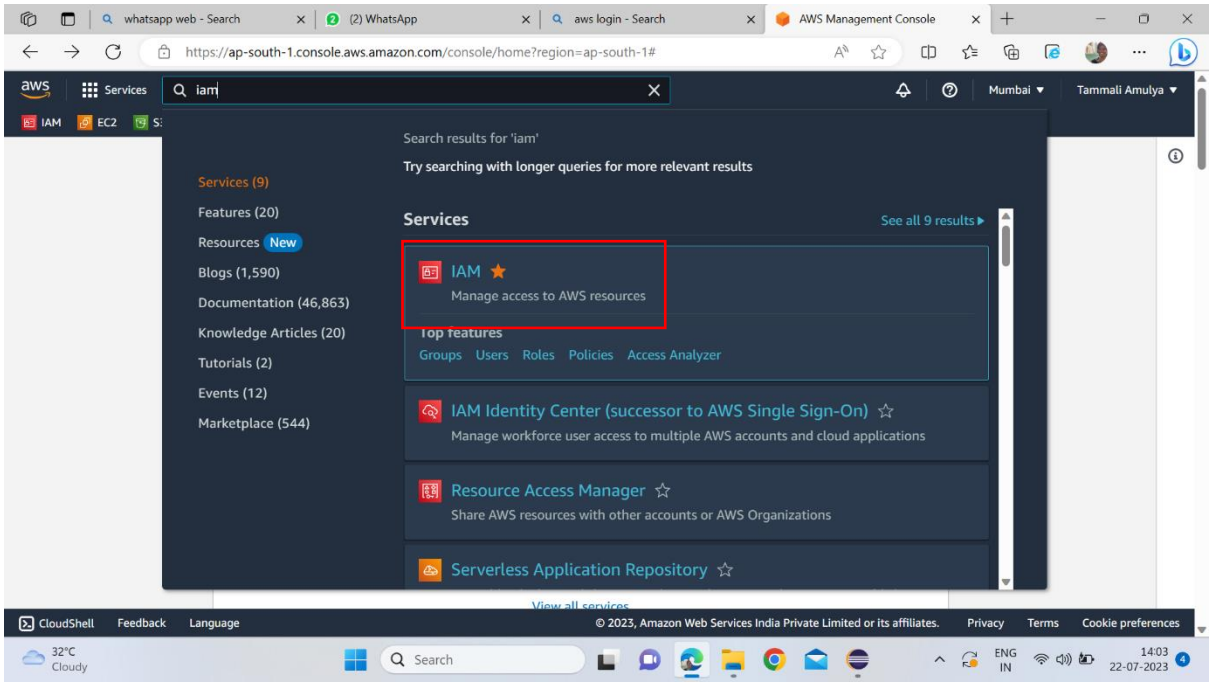
The AWS Cloud9 IDE offers a rich code-editing experience with support for several programming languages and runtime debuggers, and a built-in terminal. It contains a collection of toolsthat you use to code, build, run, test, and debug software, and helps you release software to the cloud.

You access the AWS Cloud9 IDE through a web browser. You can configure the IDE to your preferences. You can switch color themes, bind shortcut keys, enable programming language specific syntax coloring and code formatting, and more.

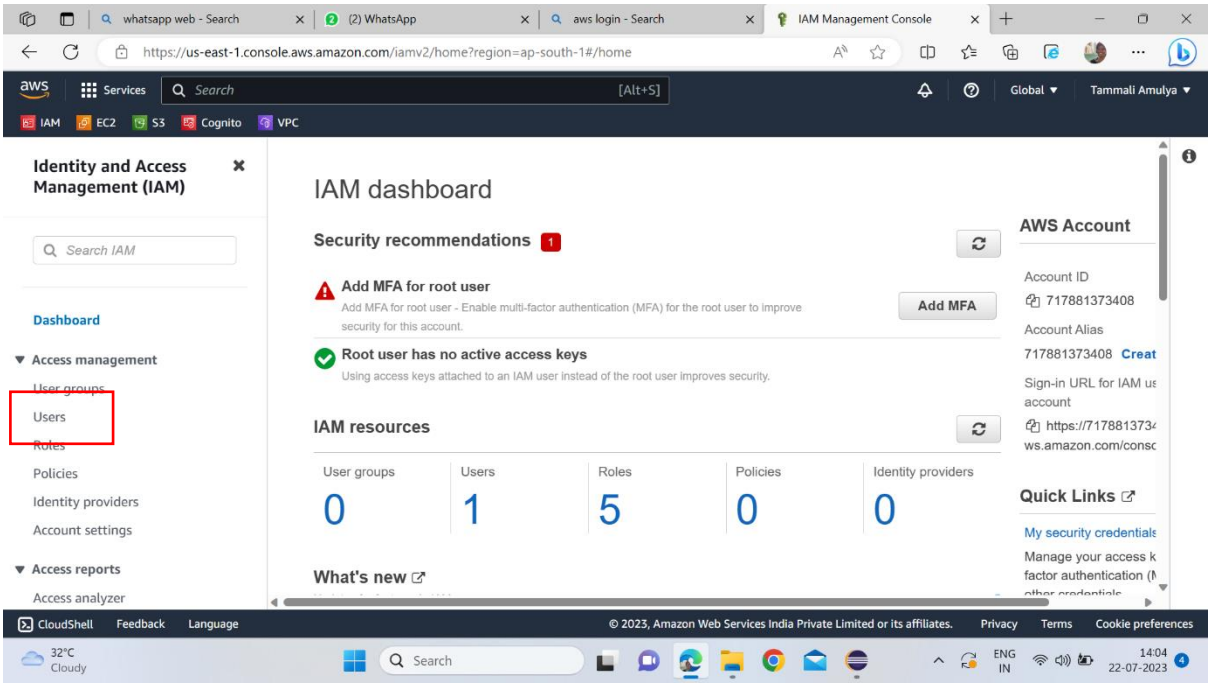


STEPS:

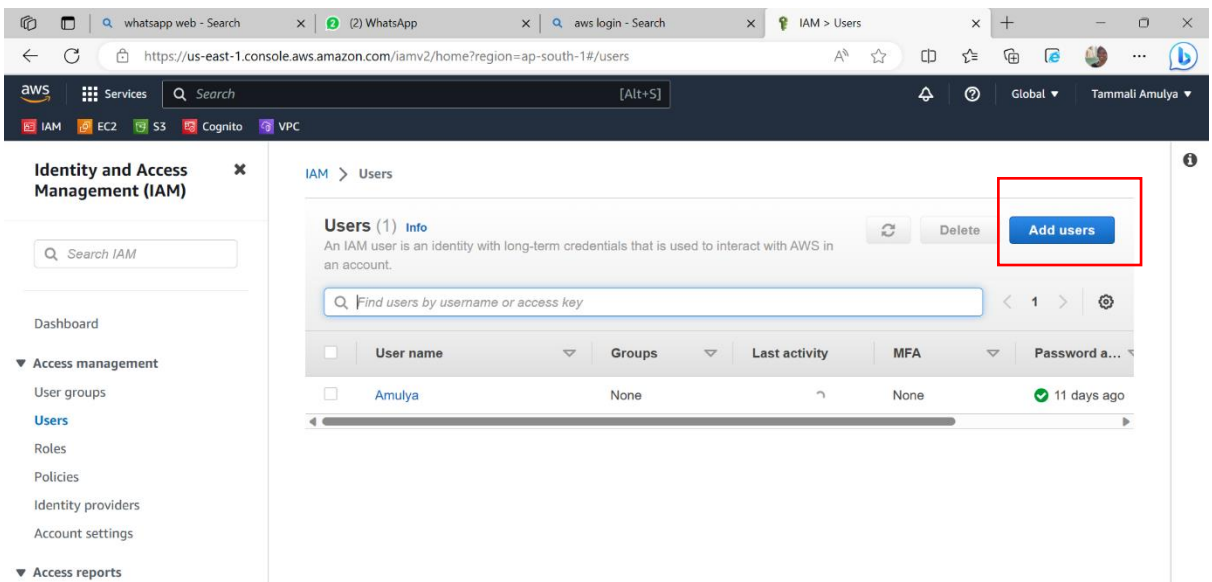
Search for IAM in searchbar and go to IAM service.



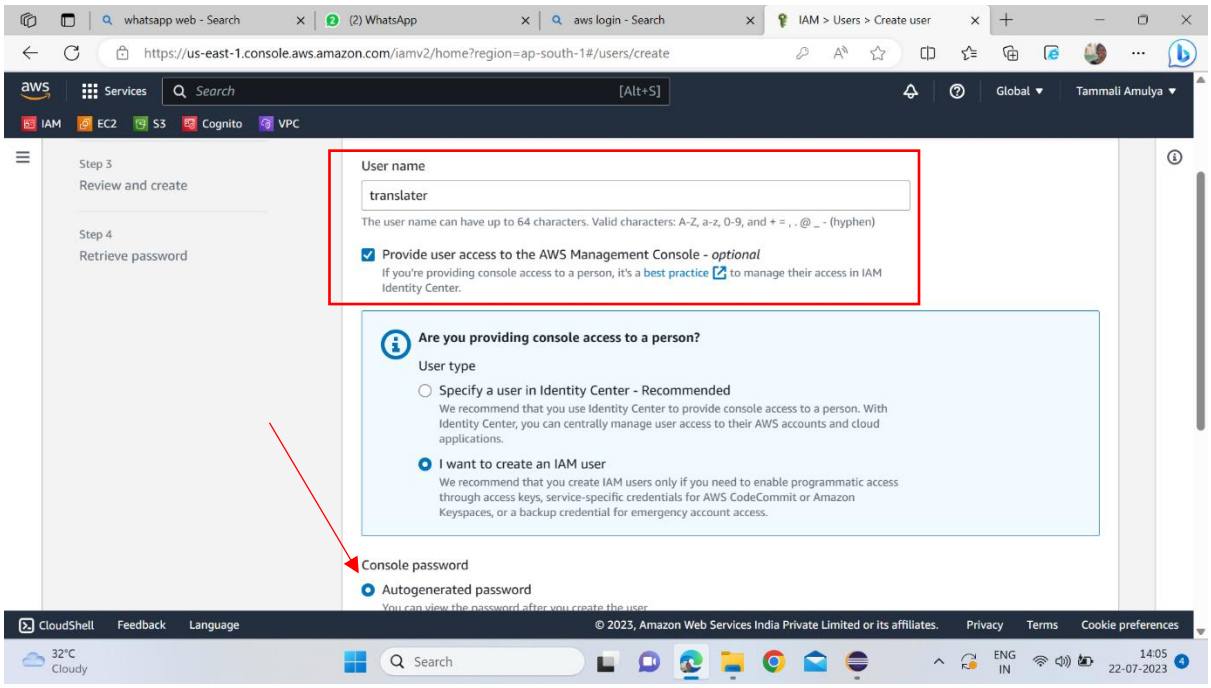
Go to users.



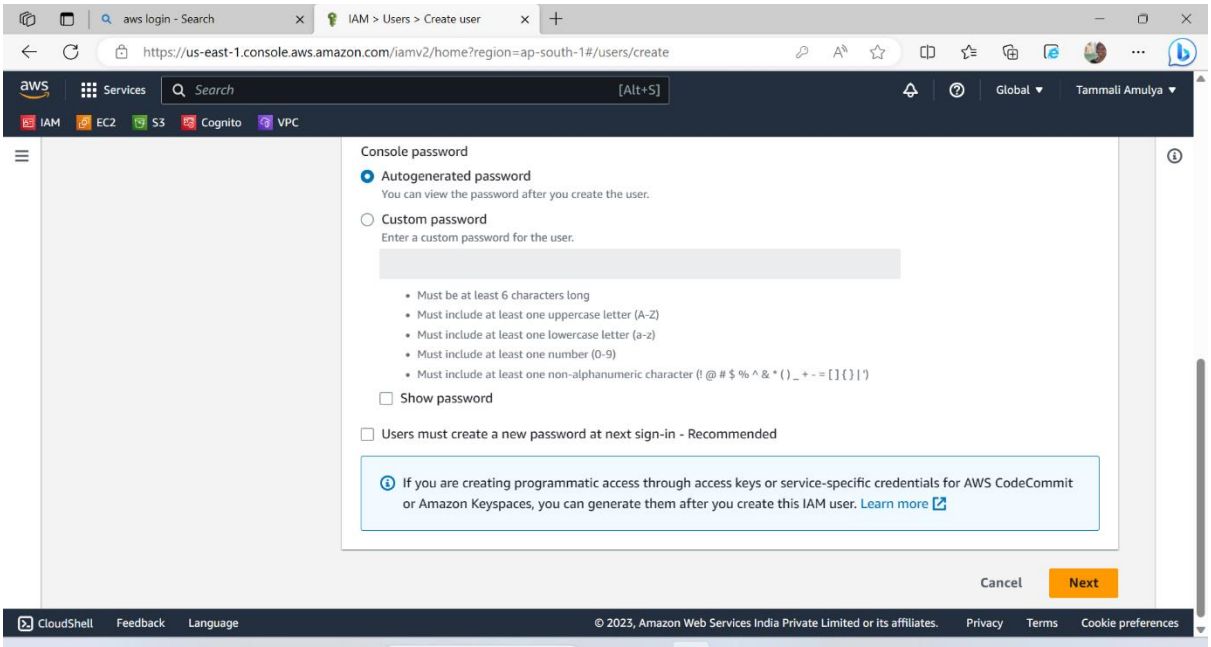
Click on add user.



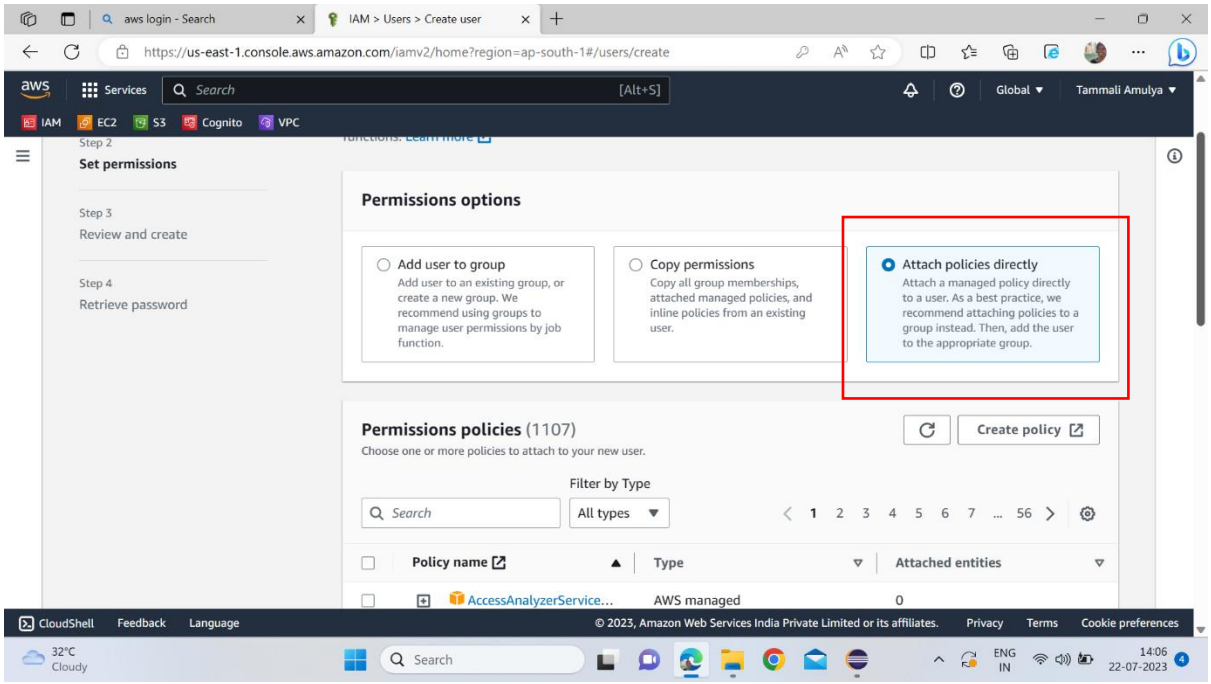
Give user name and enable access to AWS management console and enable autogenerated password.



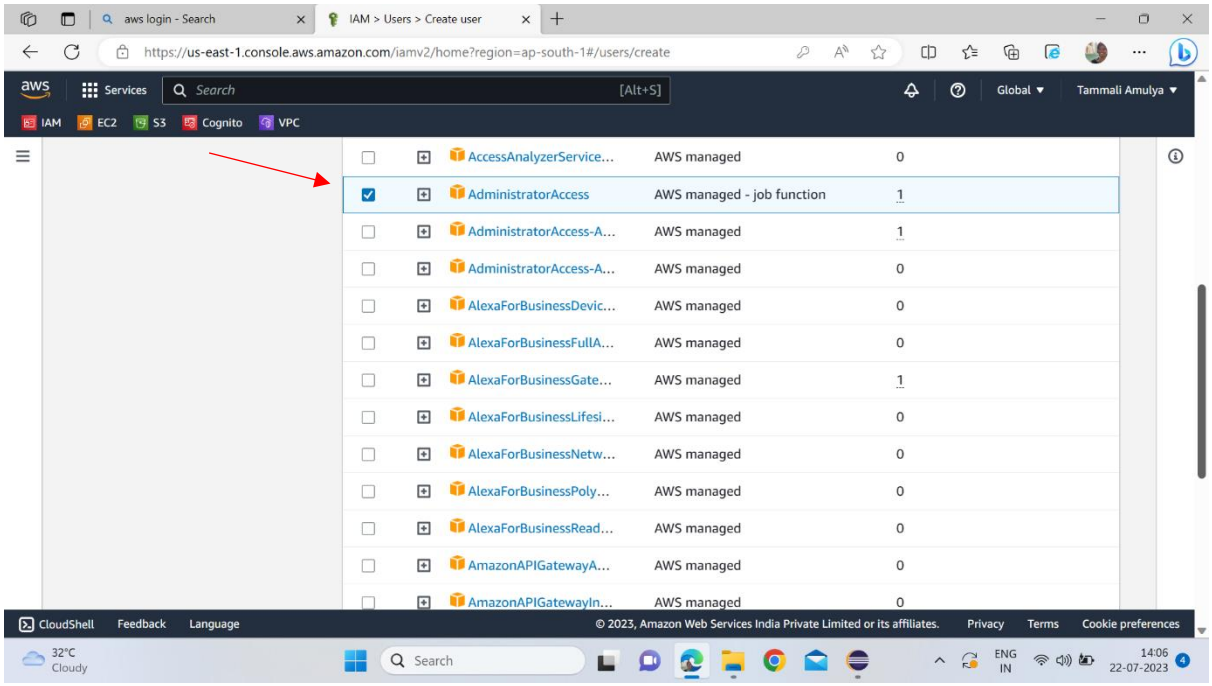
click on next.



Enable attach policies directly .

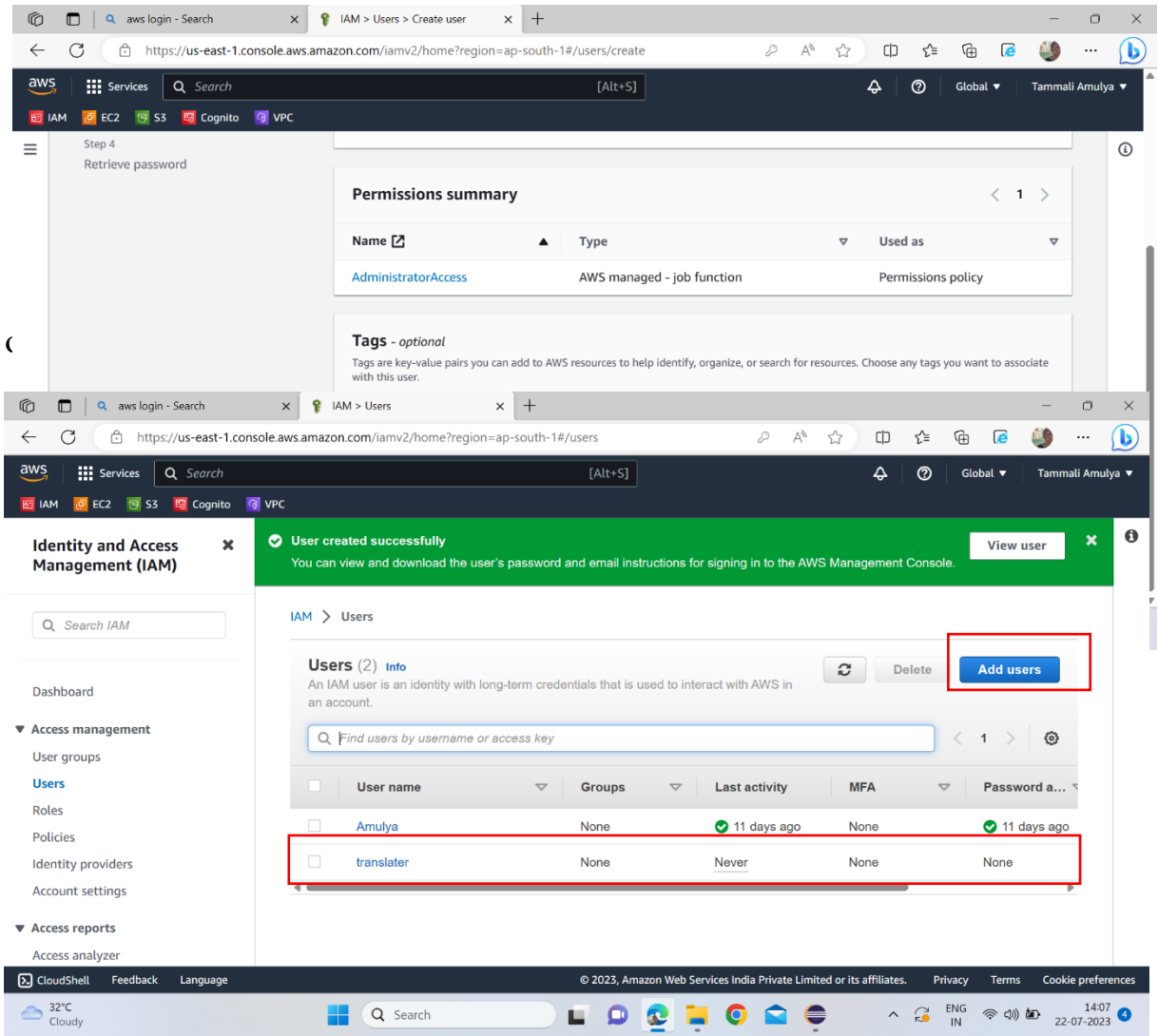


Give Administrator Access and click on next.

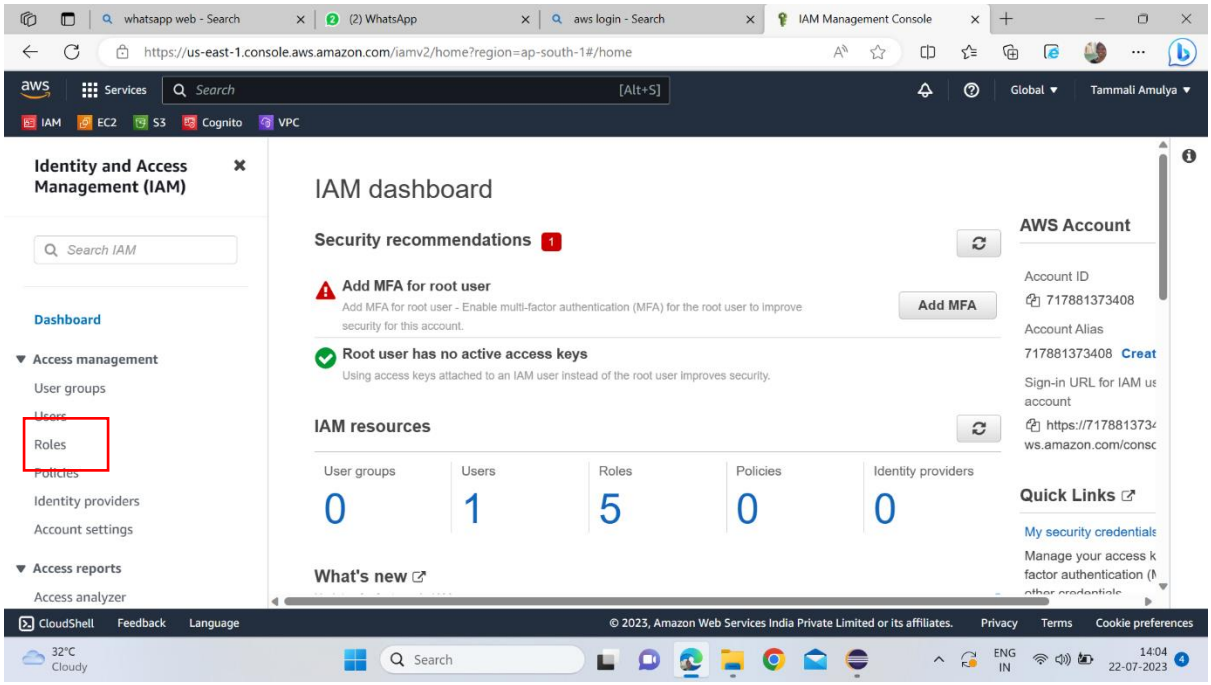


click on create user.

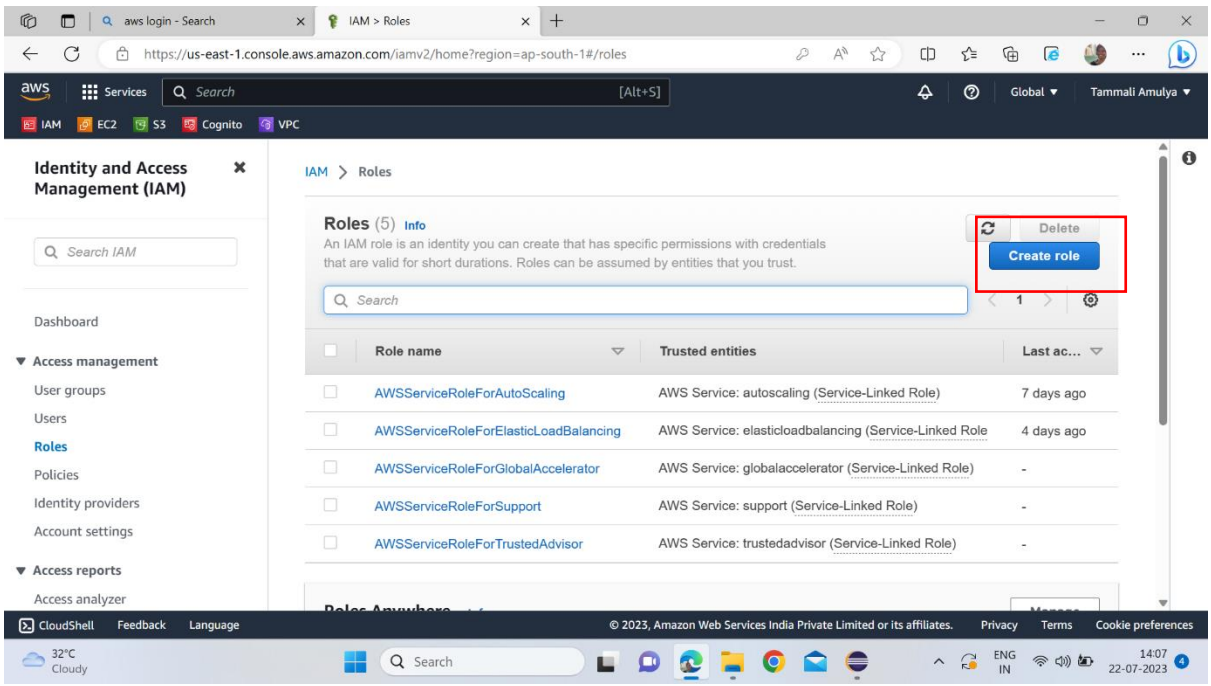
(



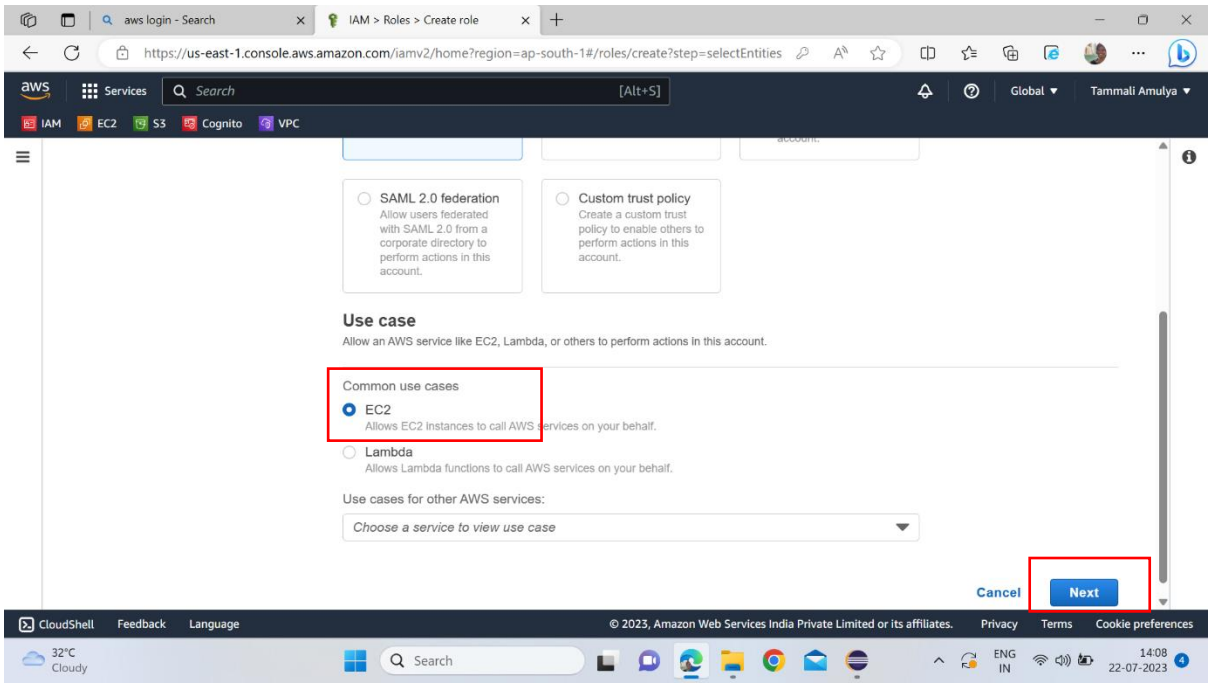
Now go to roles.



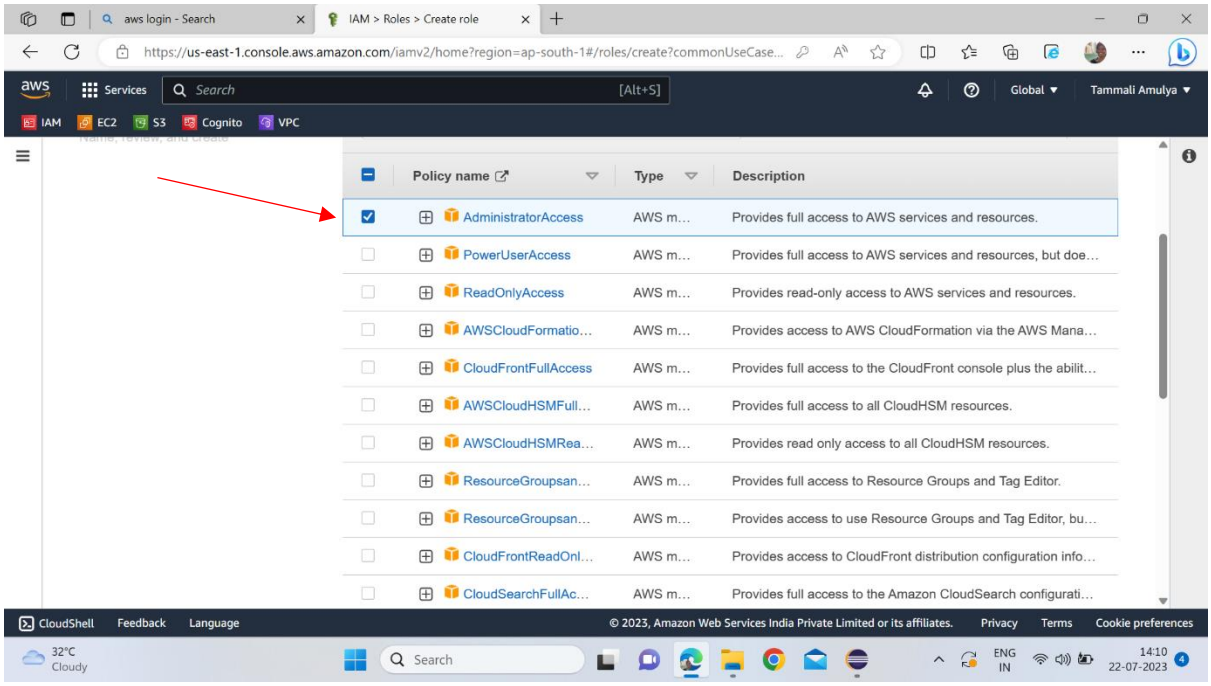
Click on create role.



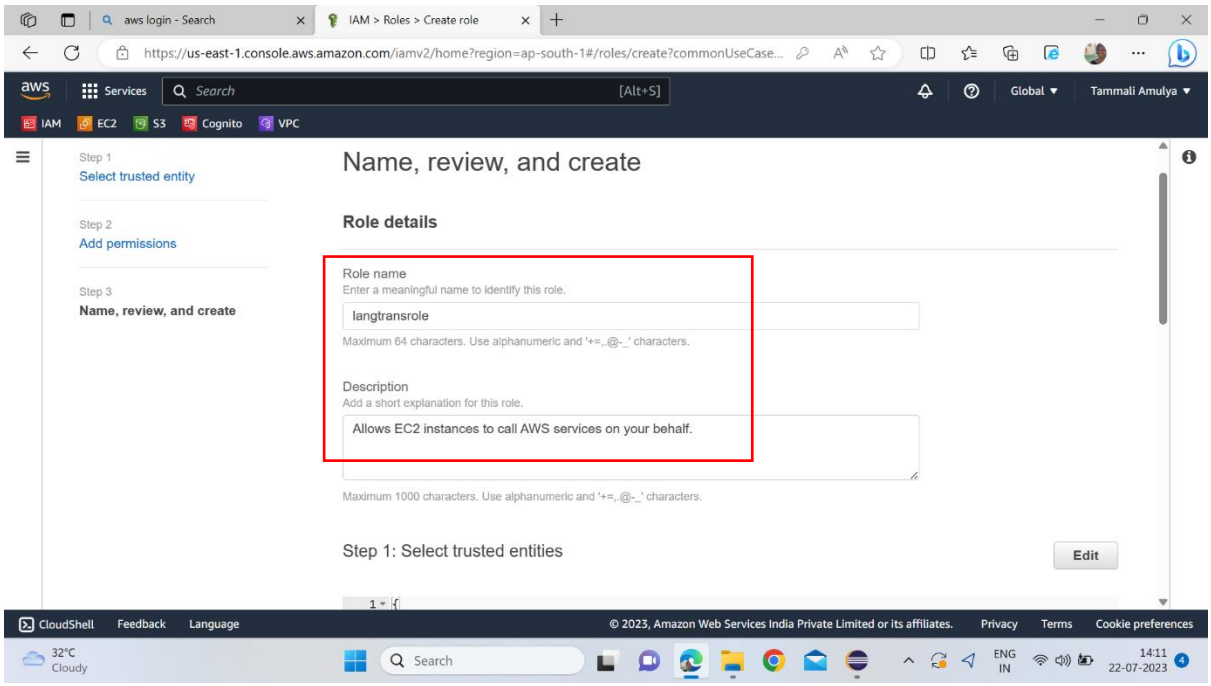
Enable EC2 service and click on next.



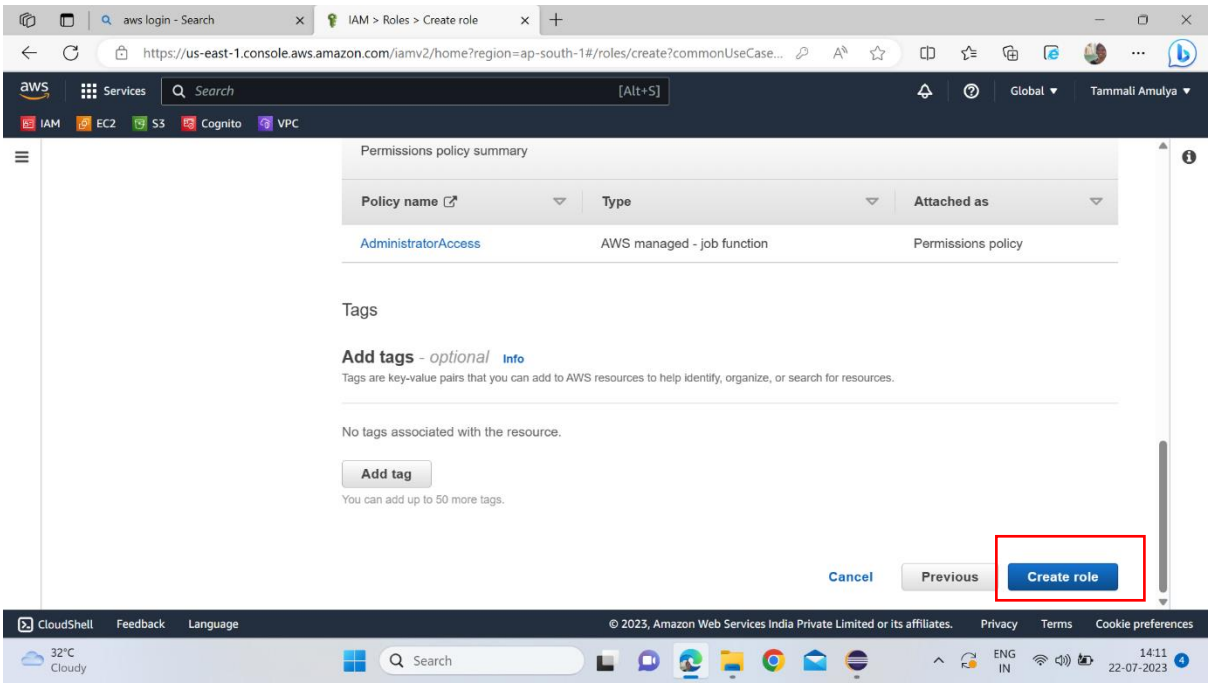
Give administrator access and click on next.



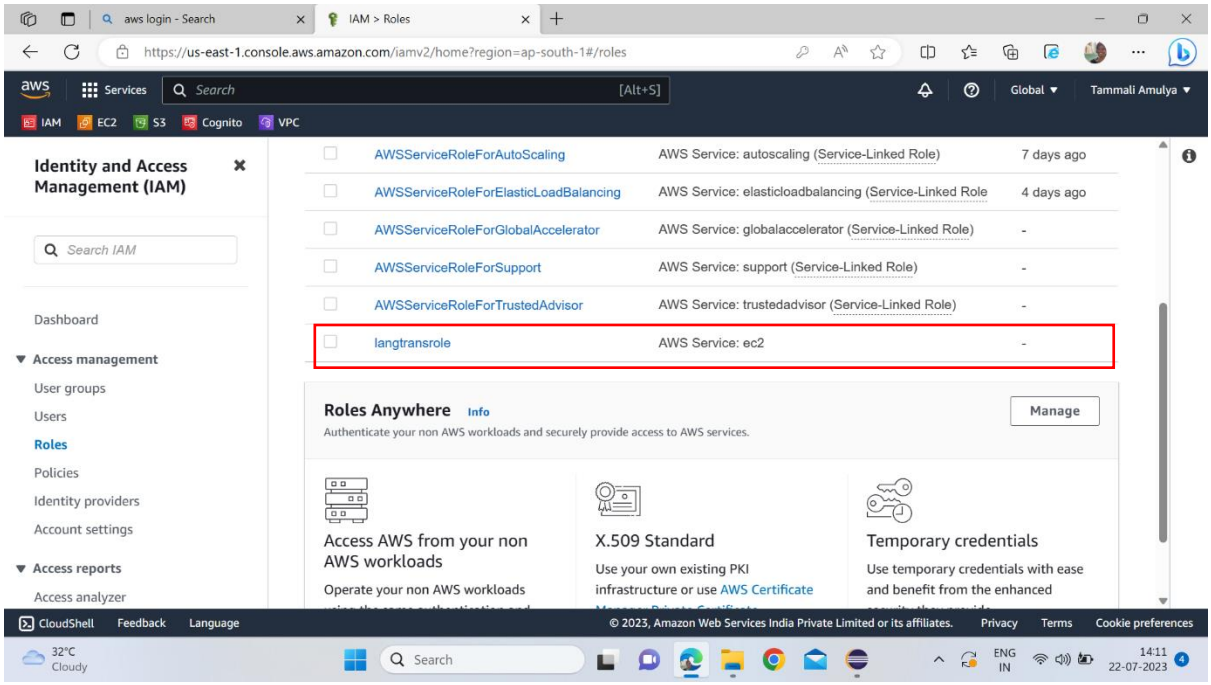
Give role name and description.



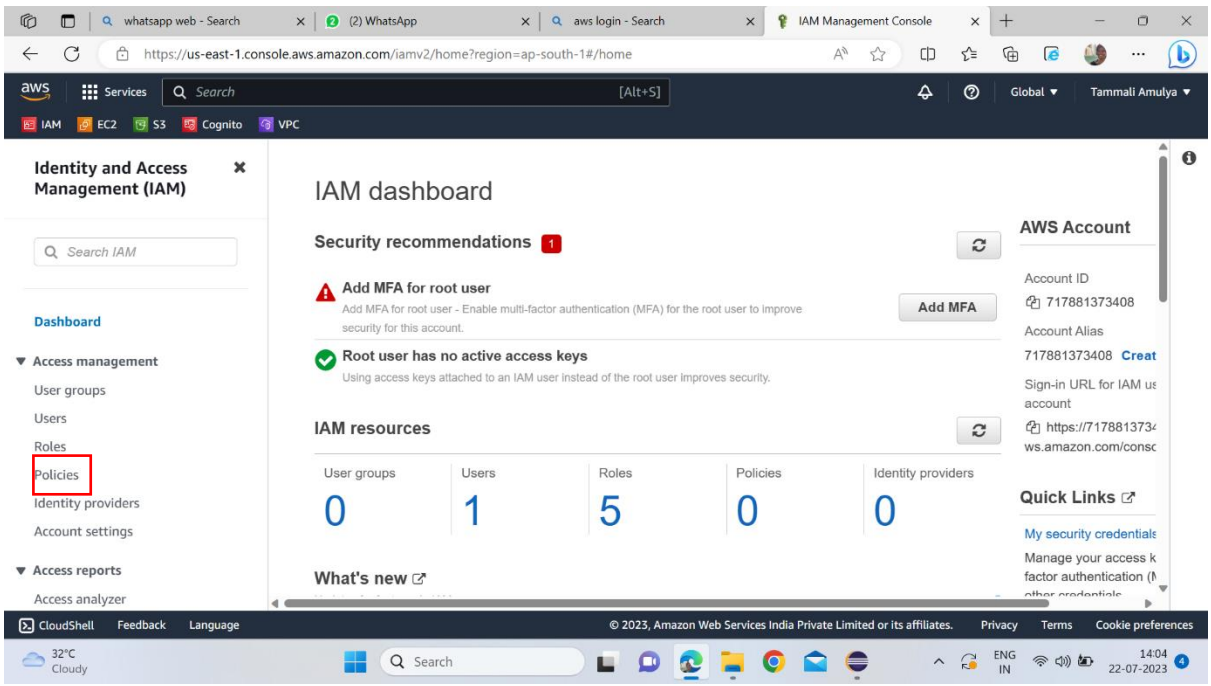
Click on create role.



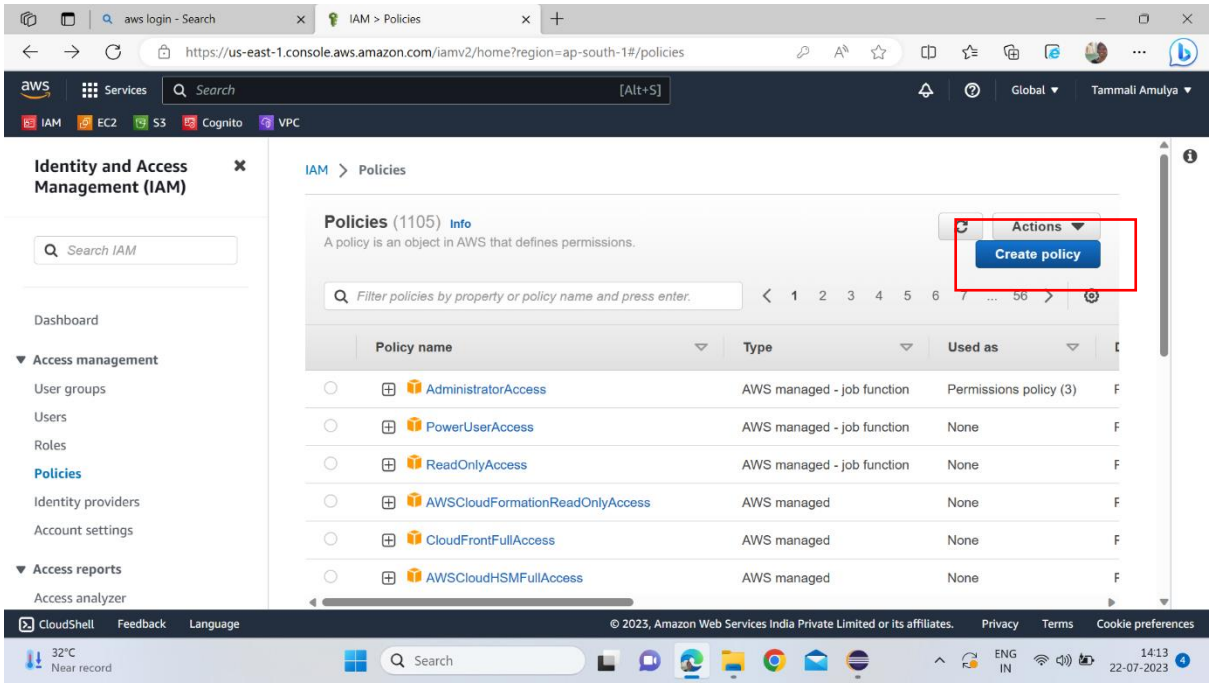
Our required role is created successfully.



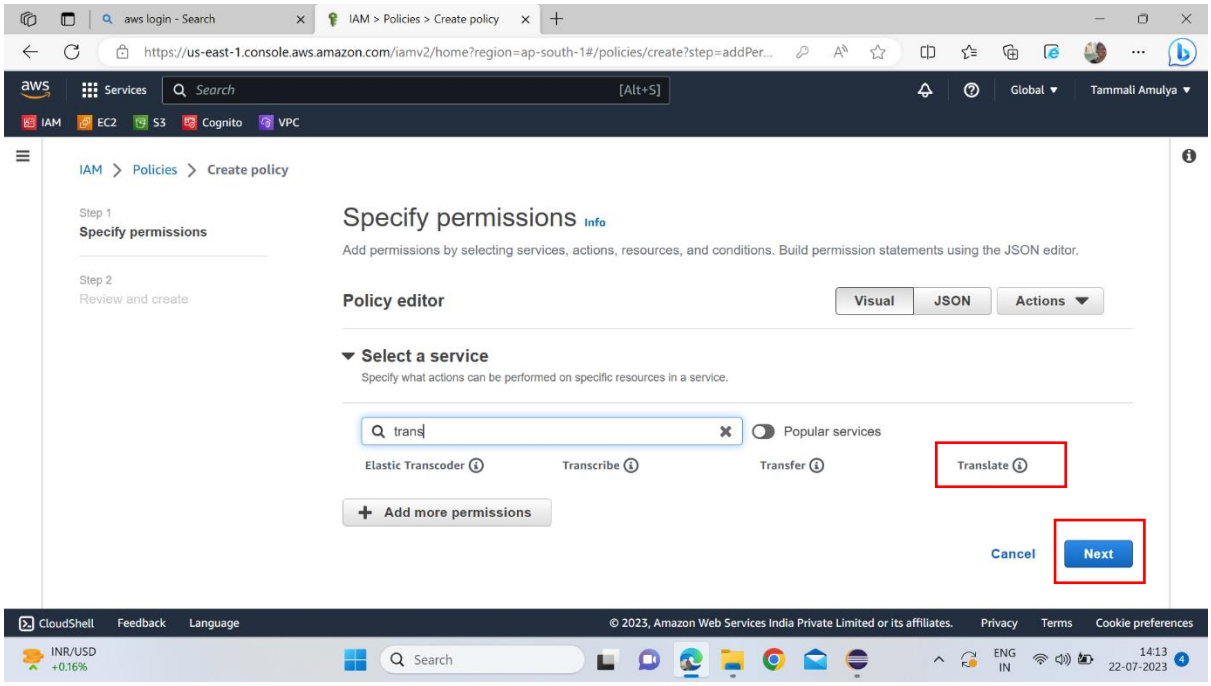
Now go to policies.



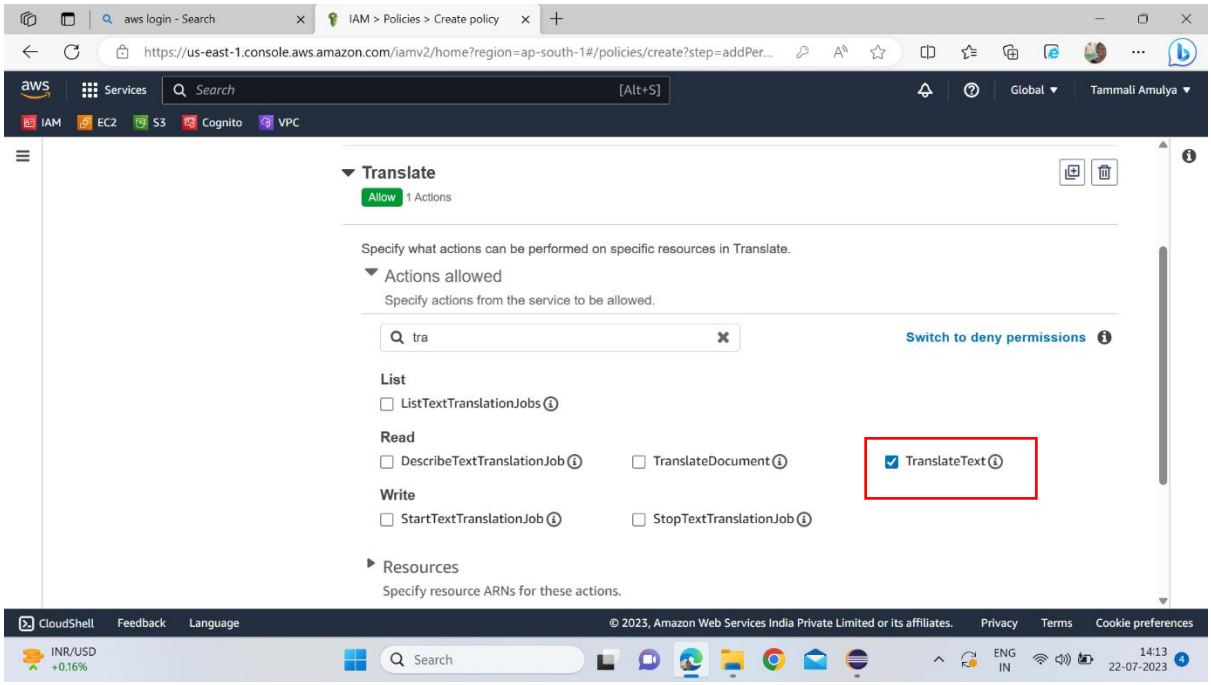
click on create policy.



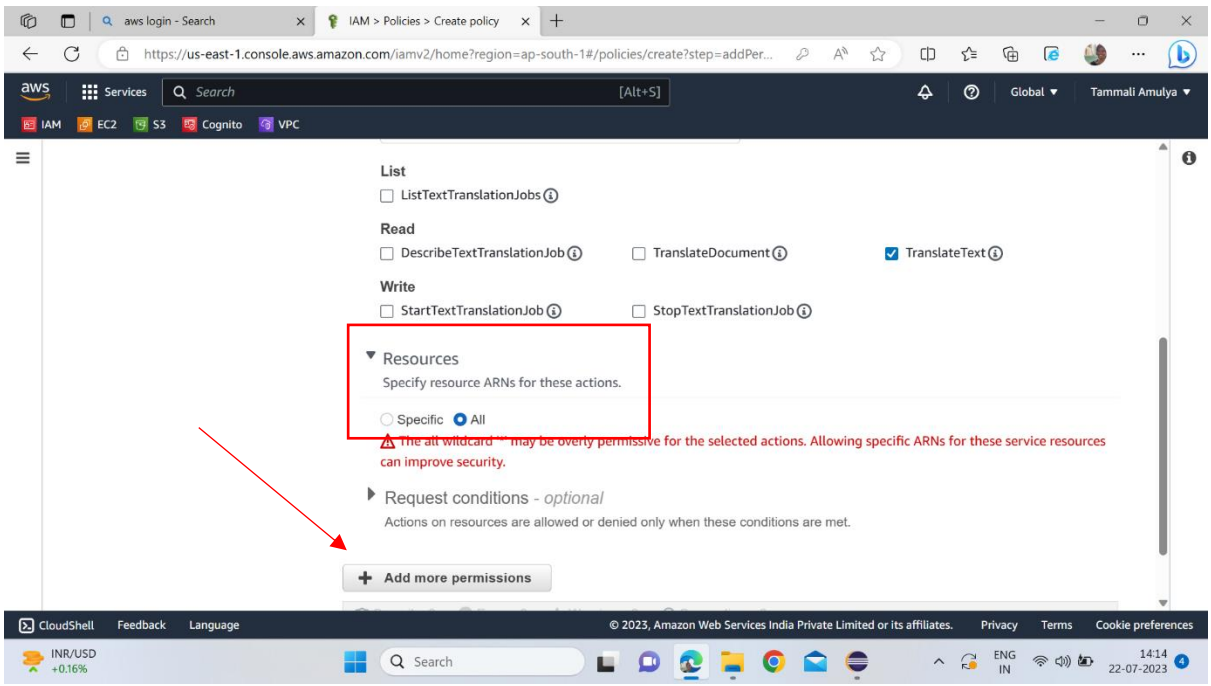
Select translate service and click on next.



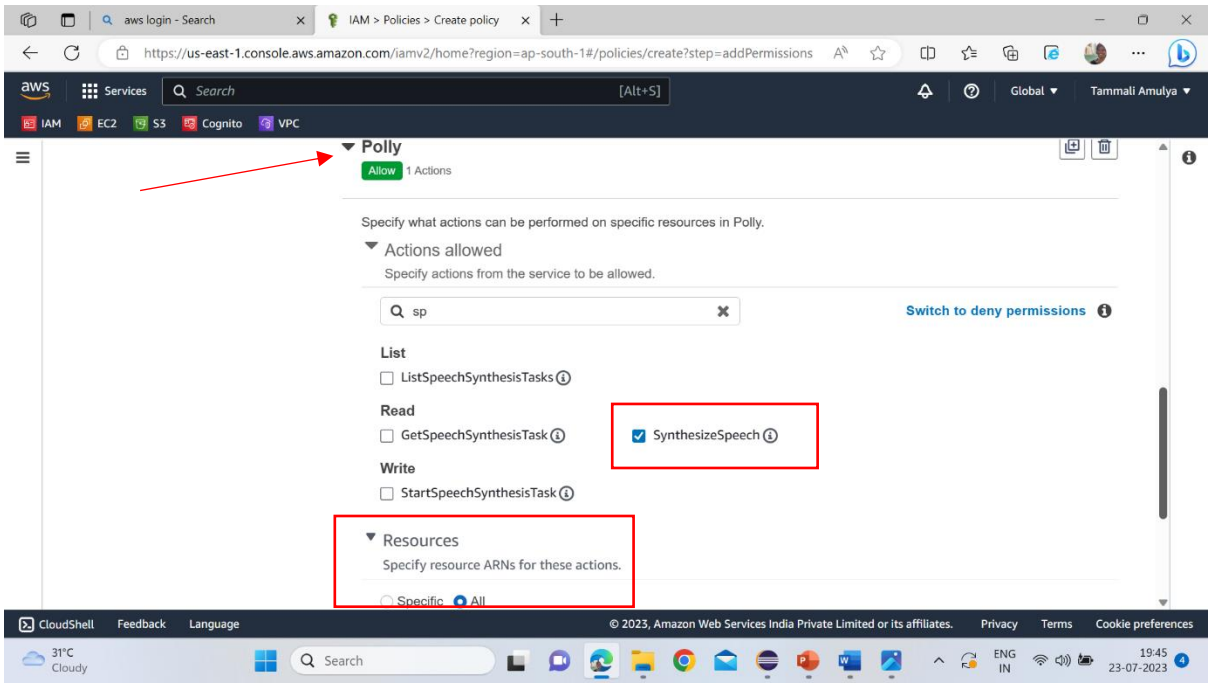
Enable Translate Text action.



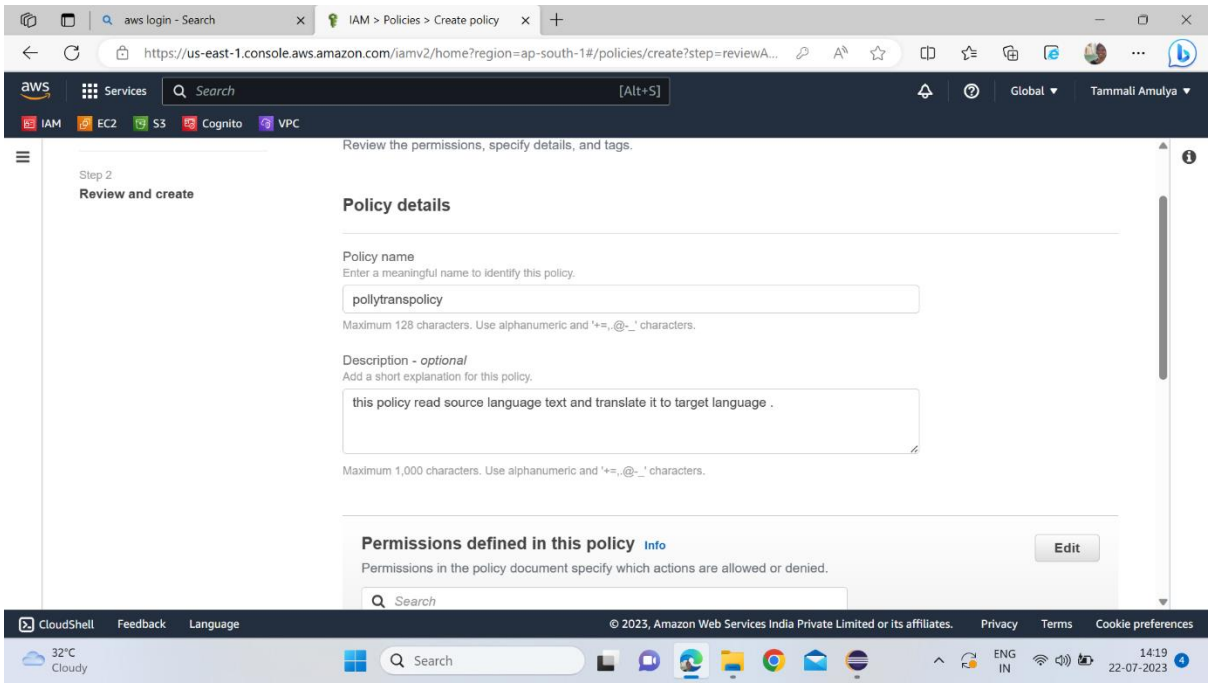
Enable all resources and click on add more permissions.



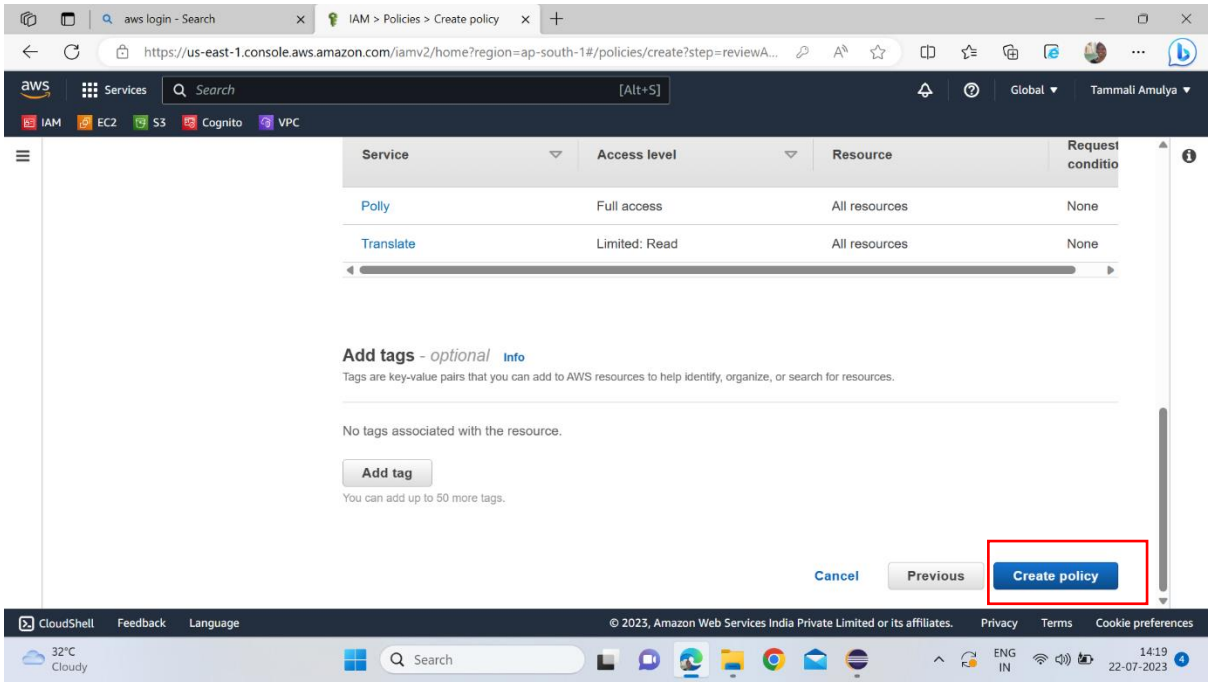
Select polly service , enable synthesizespeech action and enable all resources and click on next.



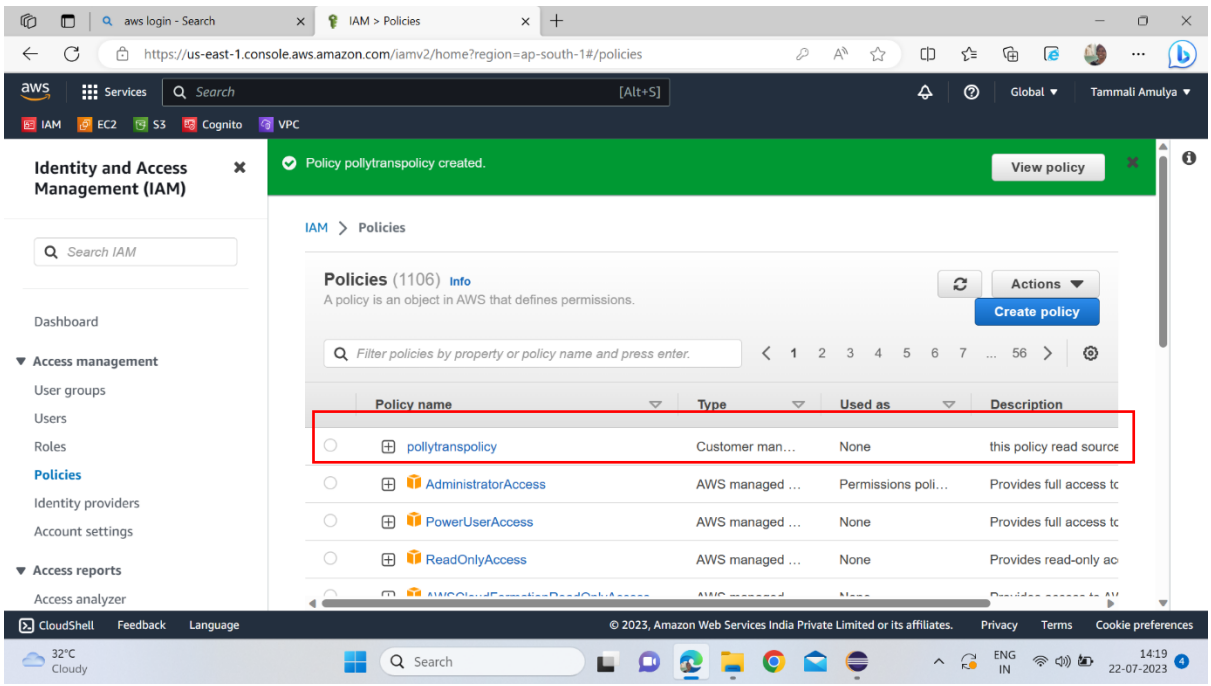
Give policy name and description.



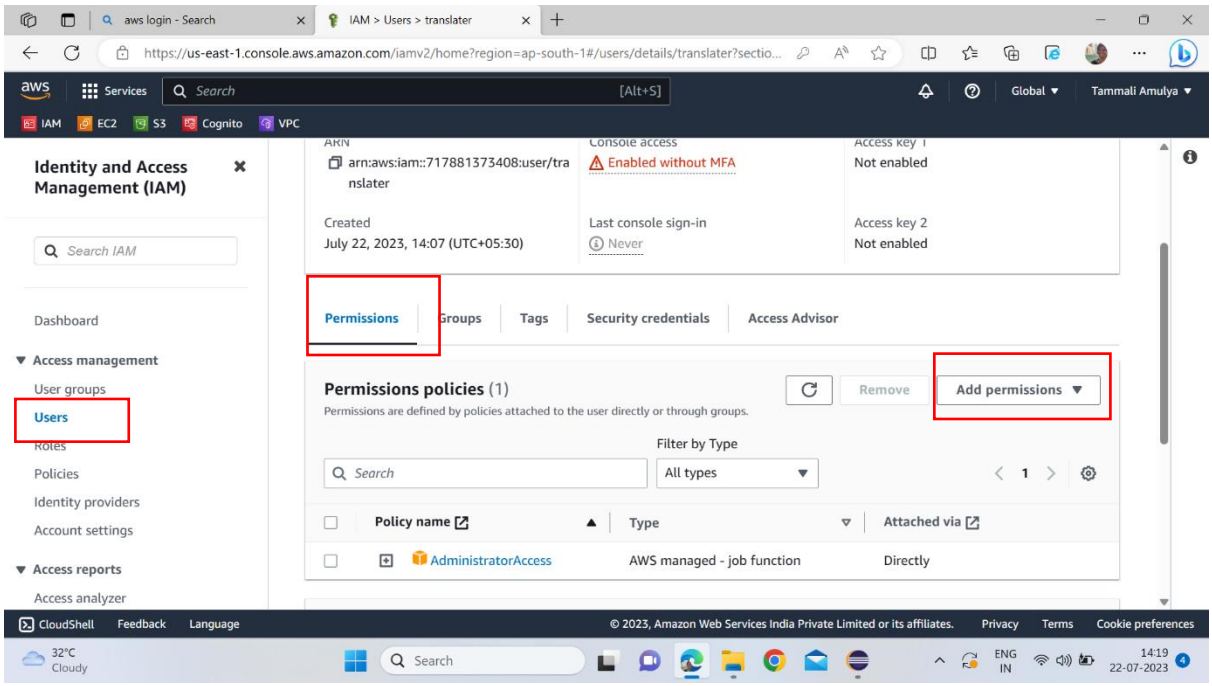
Click on create policy



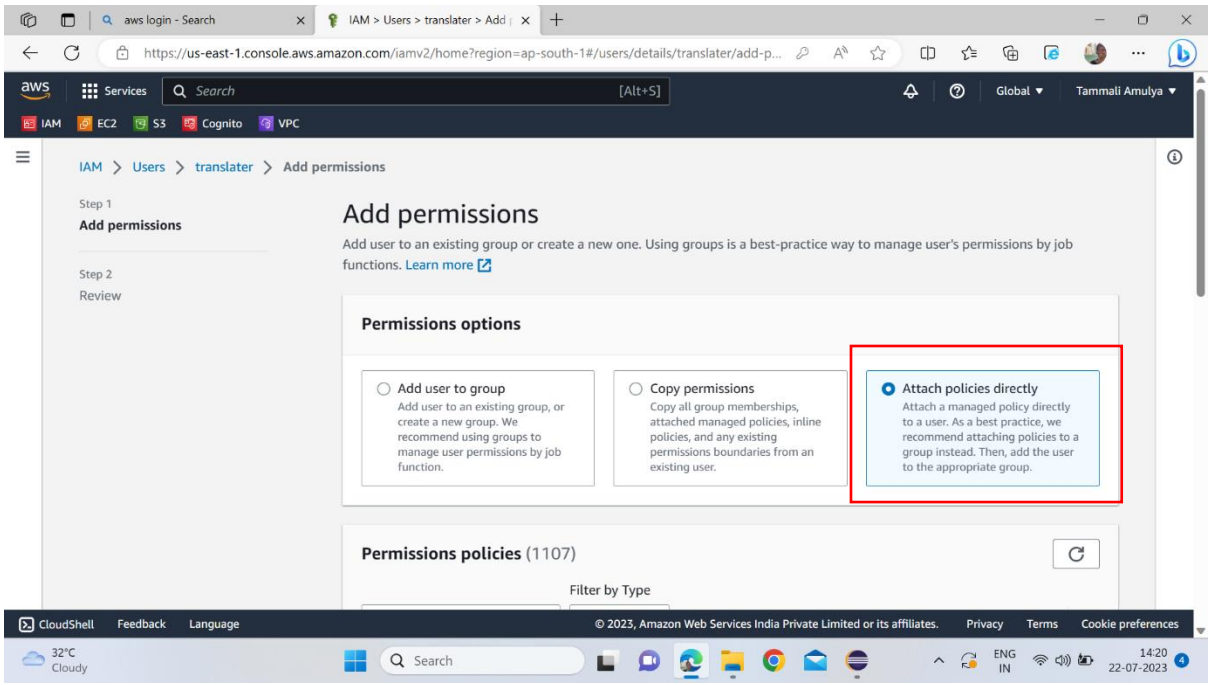
Required policy is created.



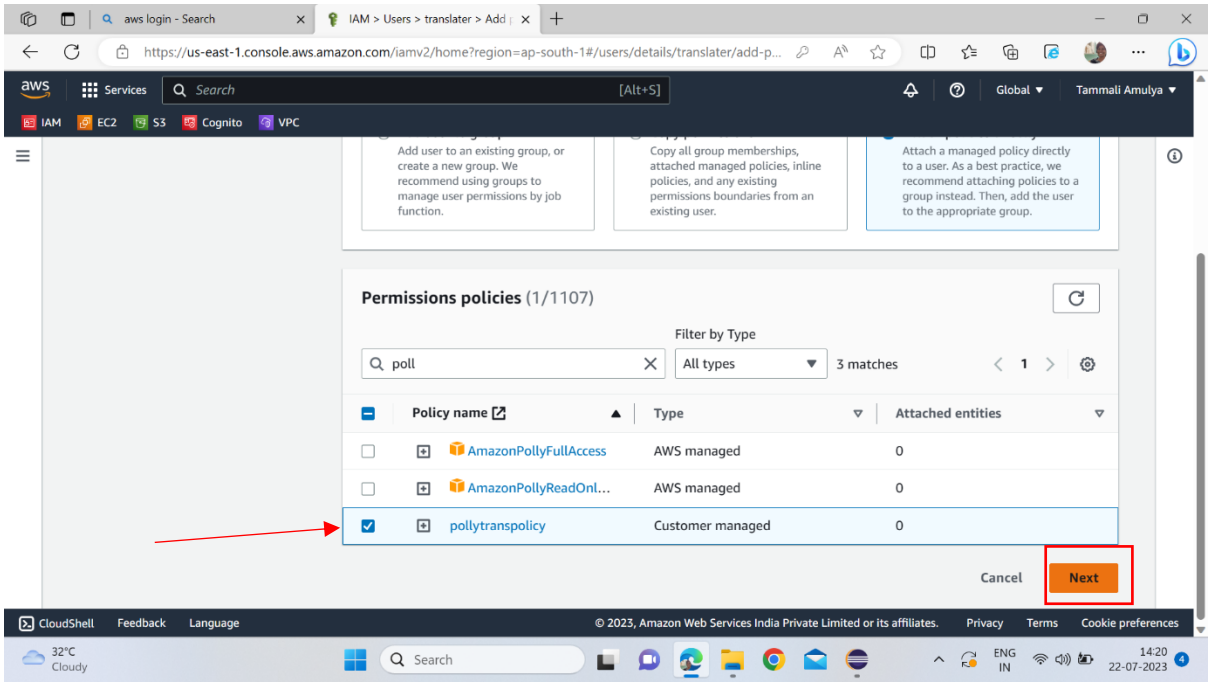
Now go to users, click on previously created user , go to permissions click on add permissions.



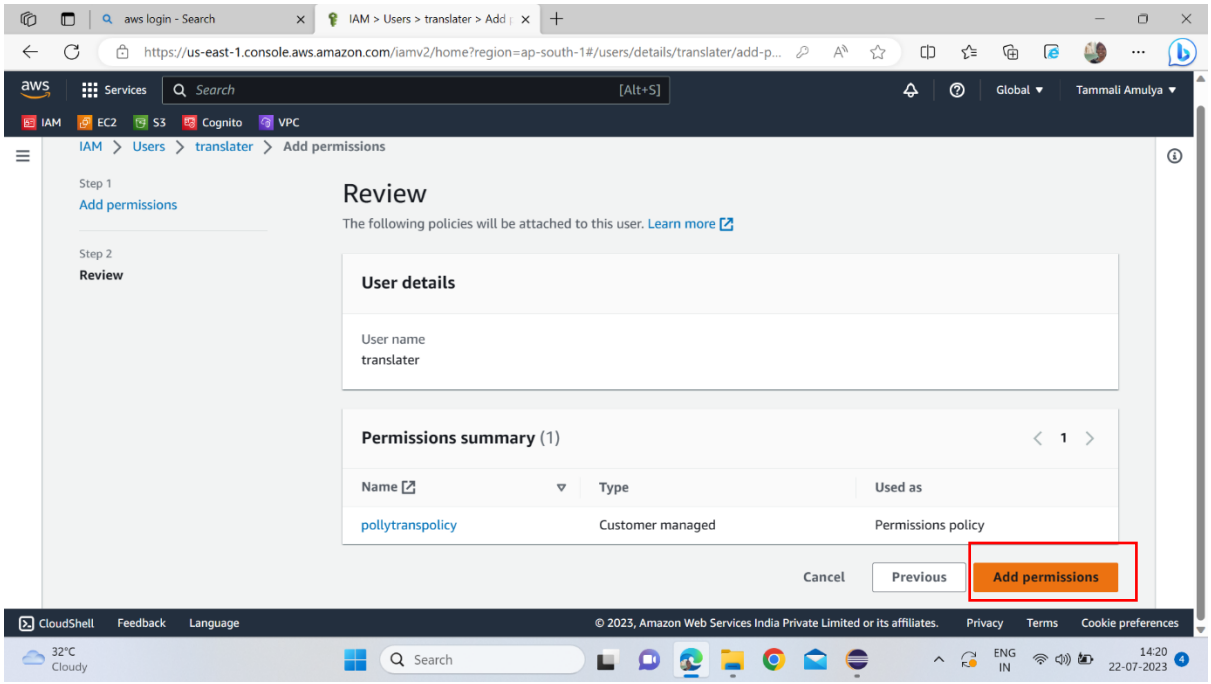
Enable attach policy directly.



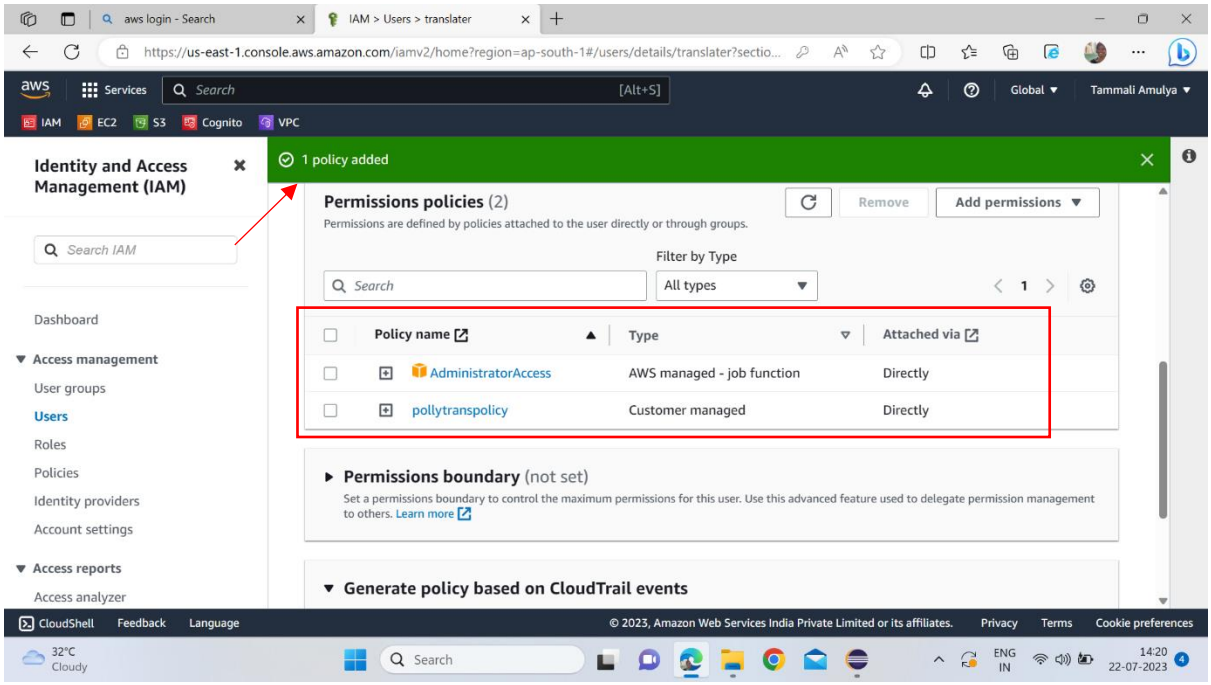
select previously created policy and click on next.



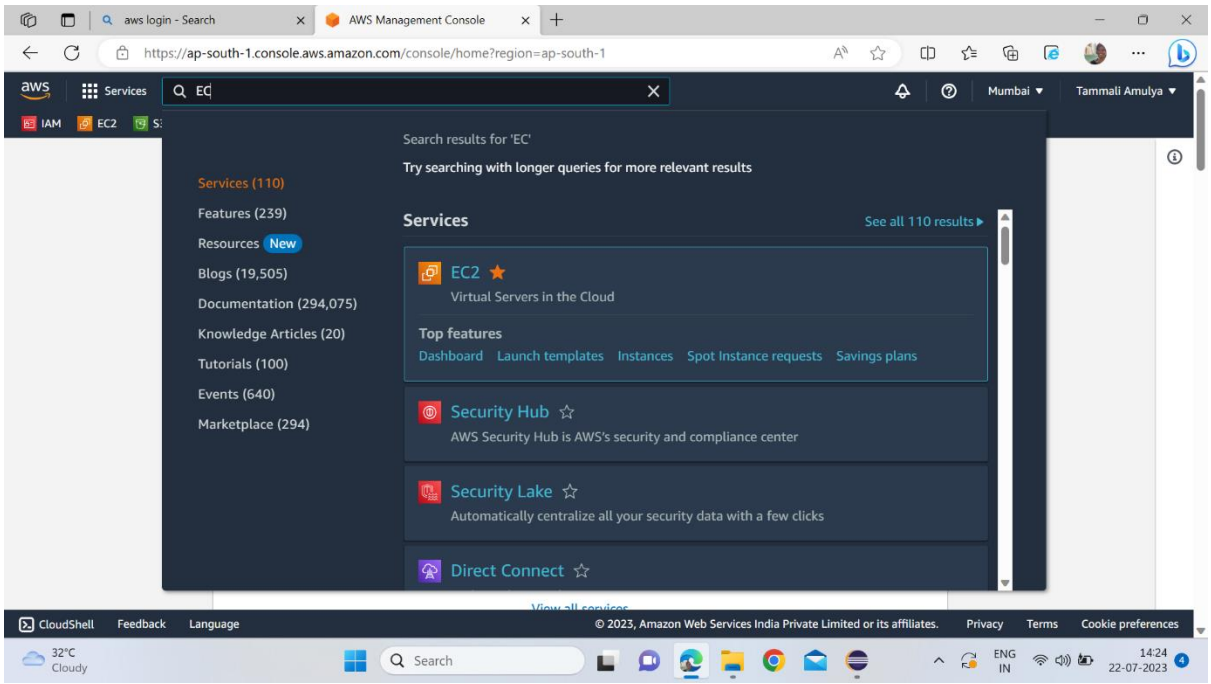
Click on add permissions.



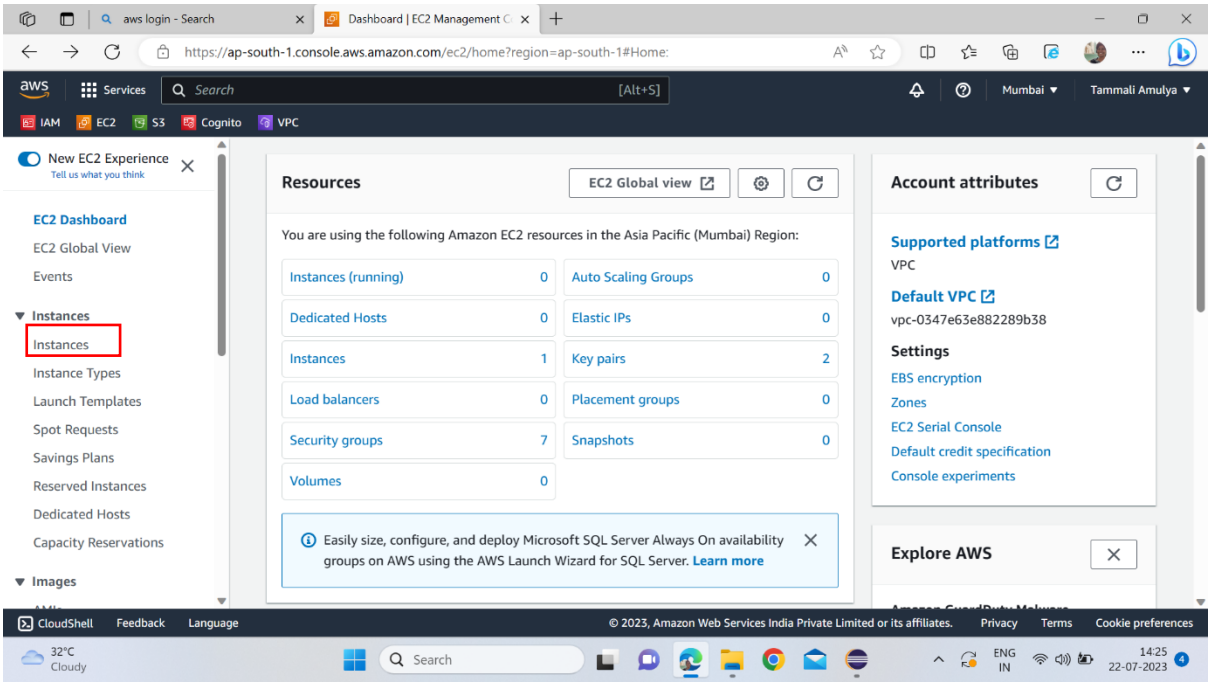
Our policy is attached to user.



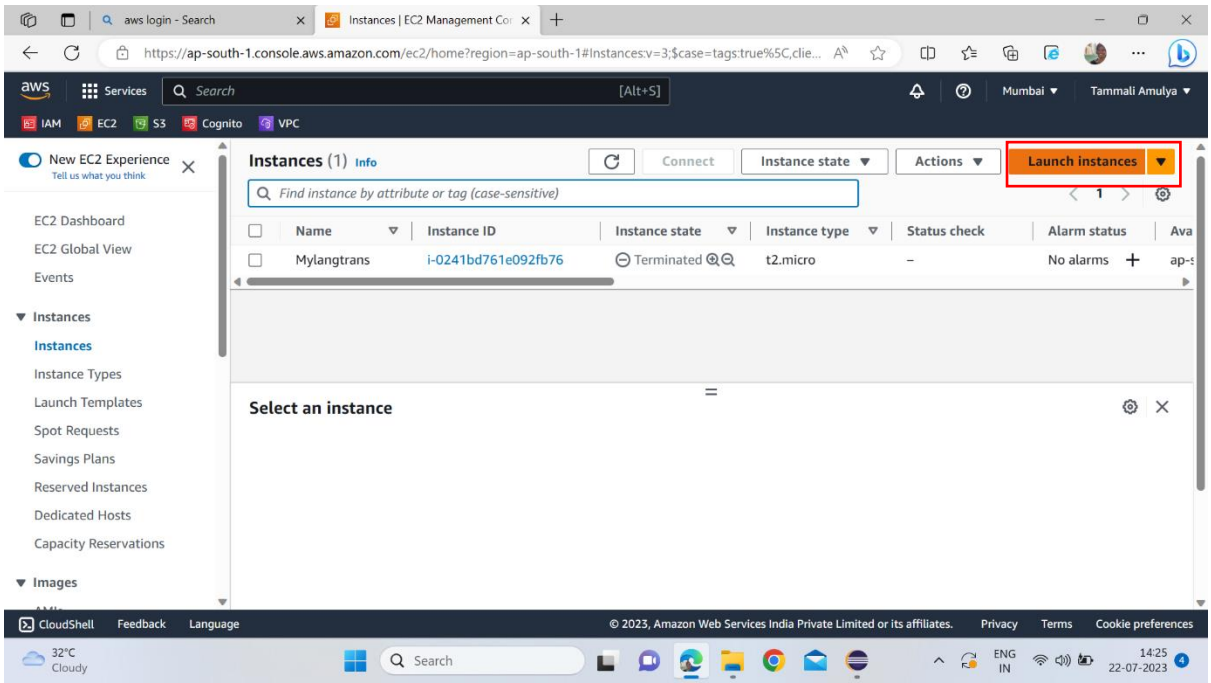
Now search for EC2 service and click on EC2.



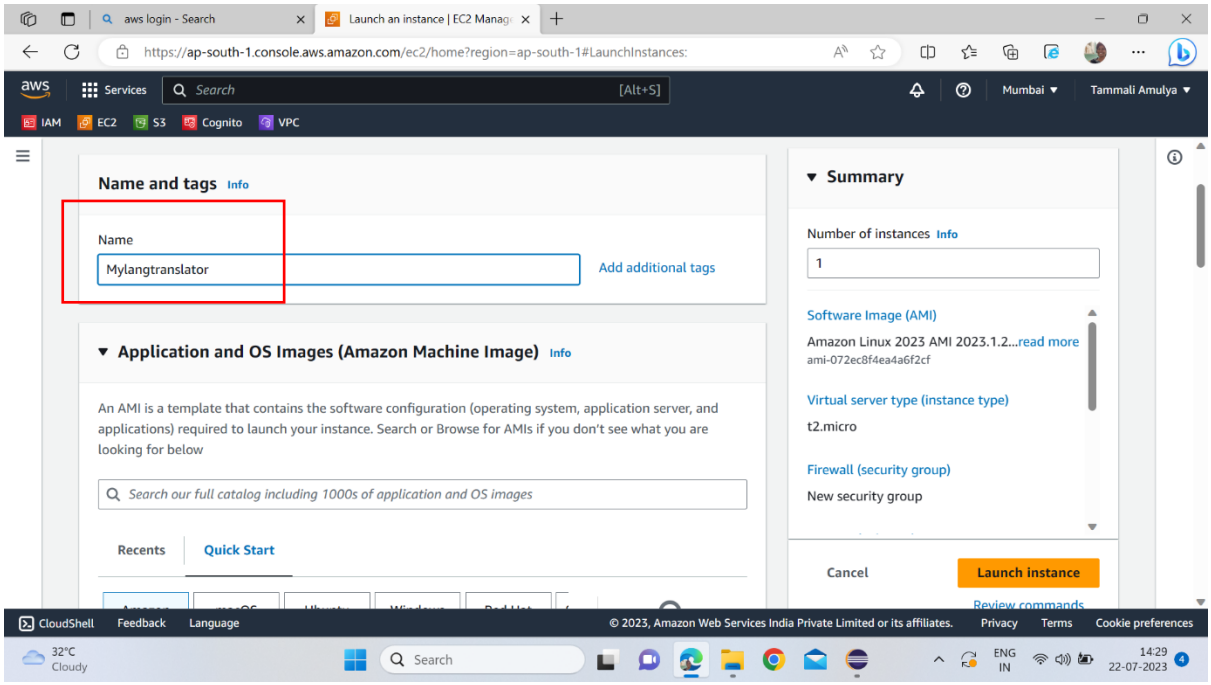
Go to instances.



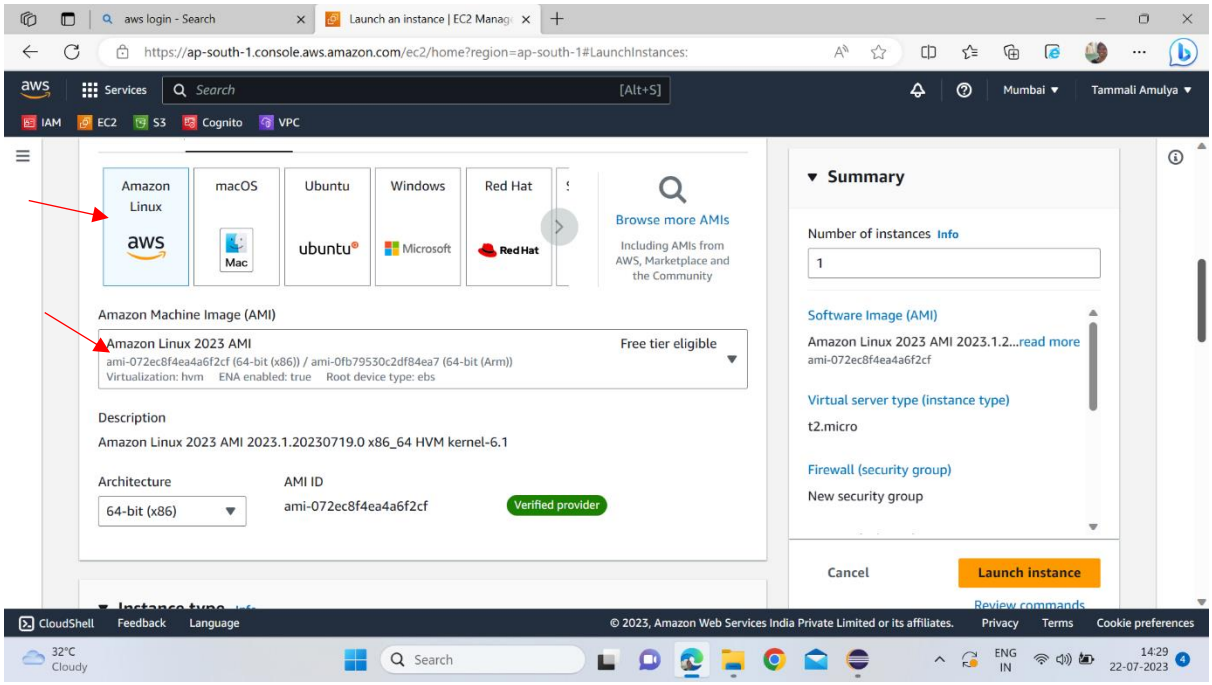
click on launch instances.



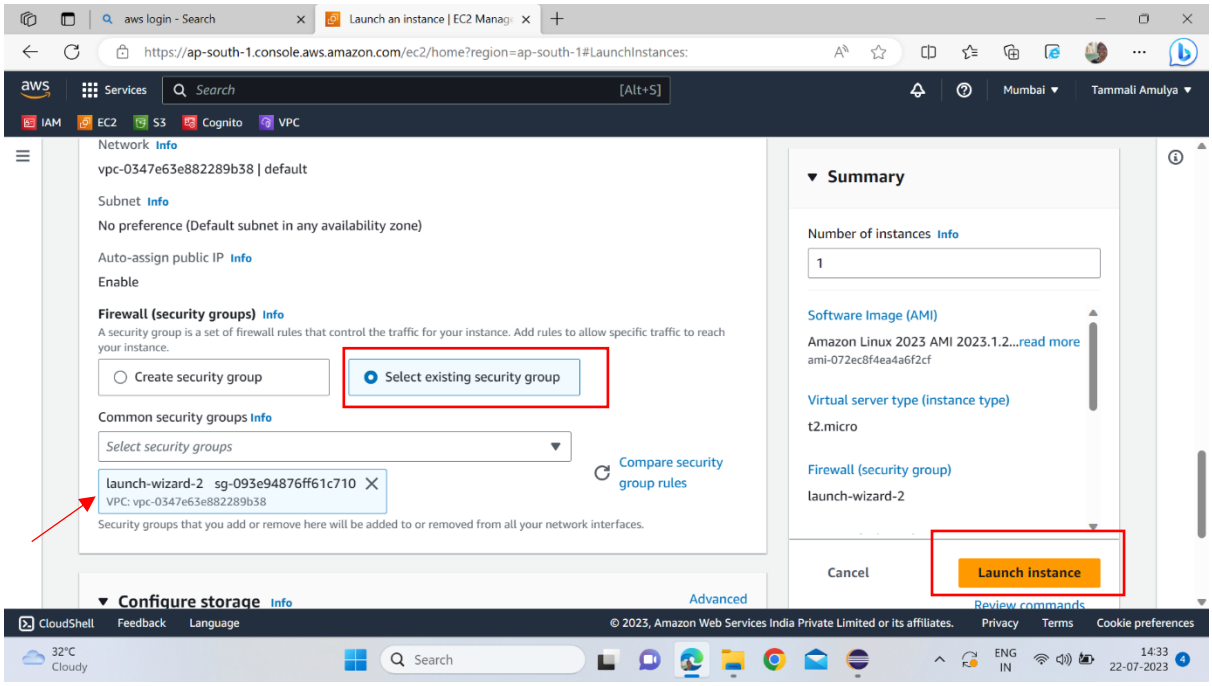
Give instance name.



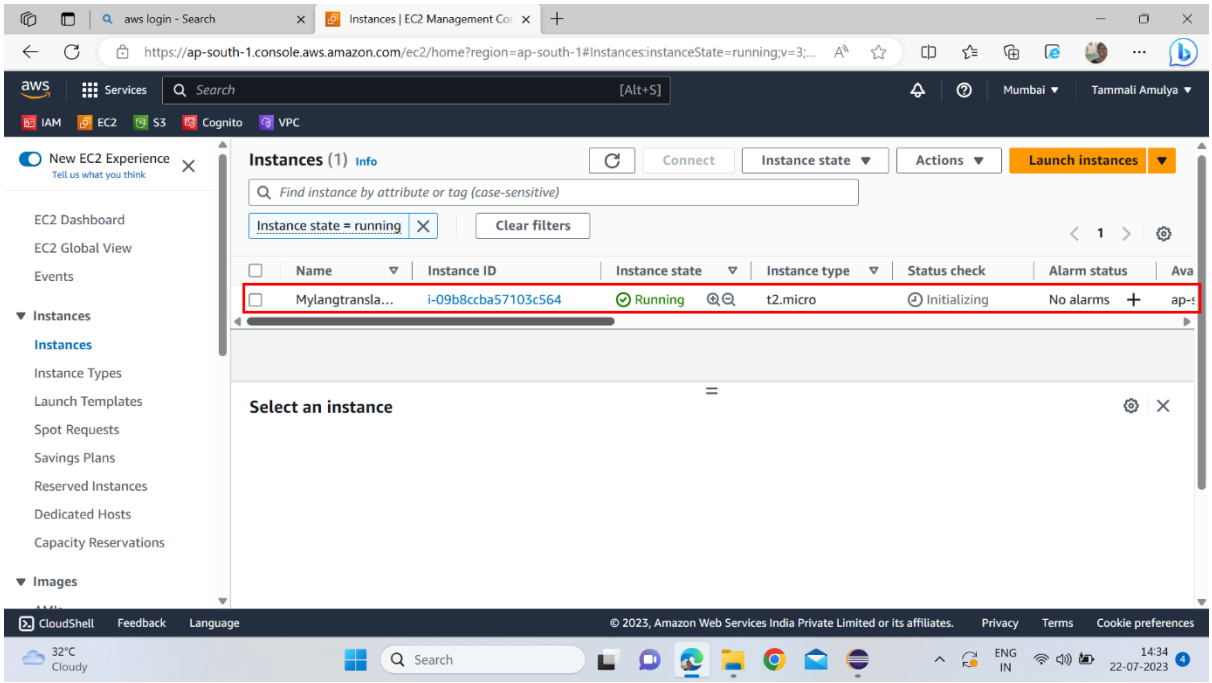
select operating system .



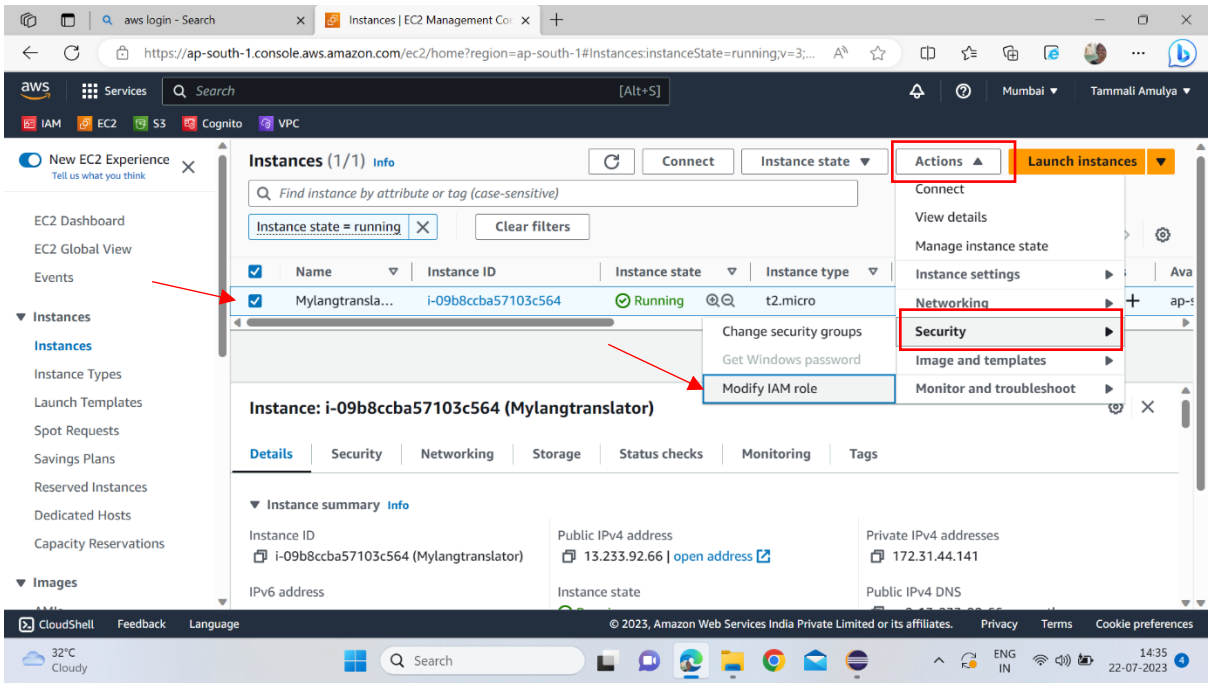
select existing security group and click on launch instances.



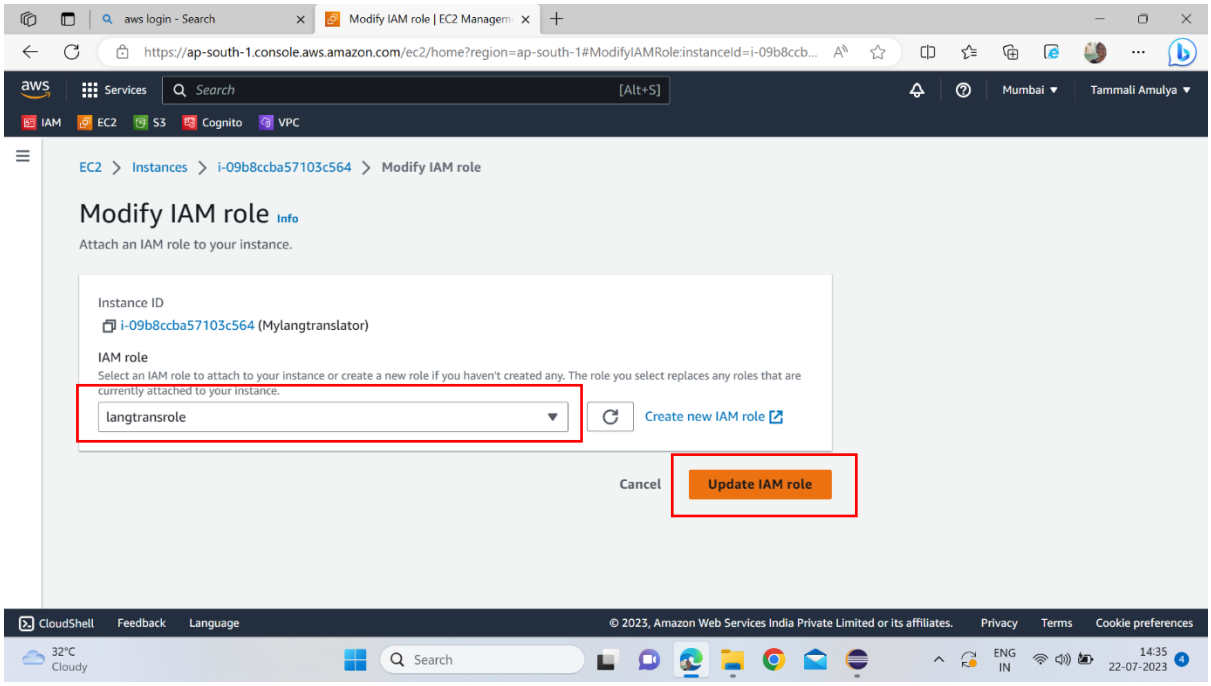
required instance created sucessfully



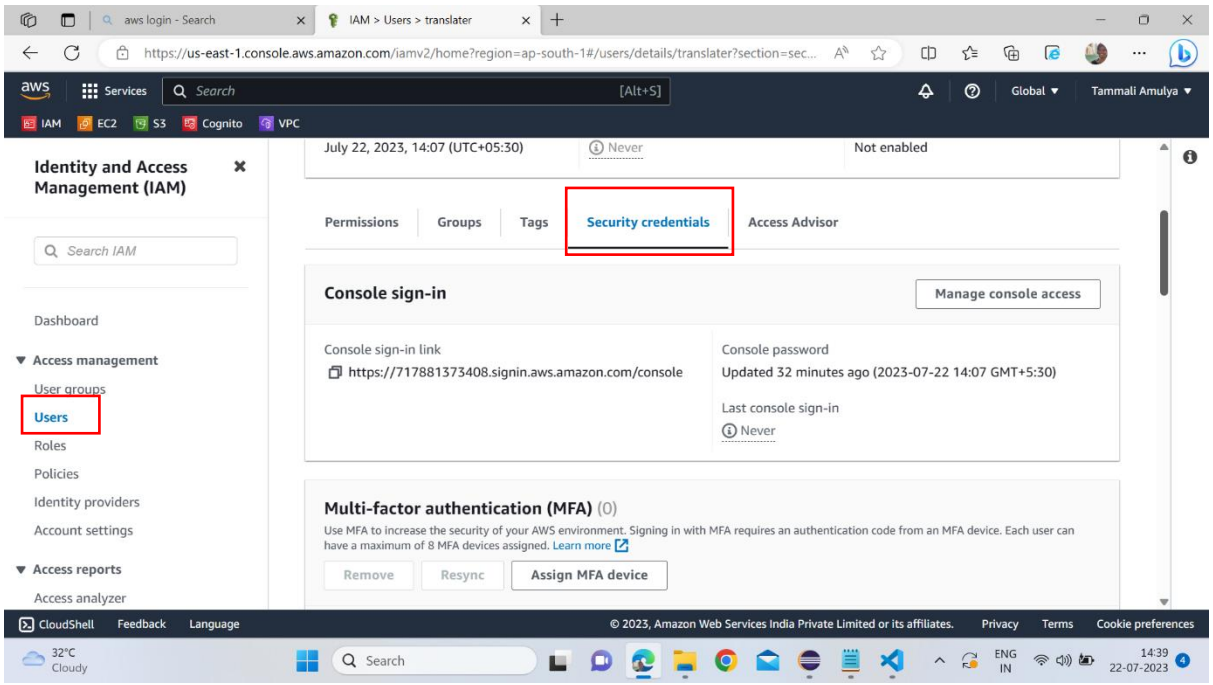
Select instance, click on action, go to security and click on modify IAM role.



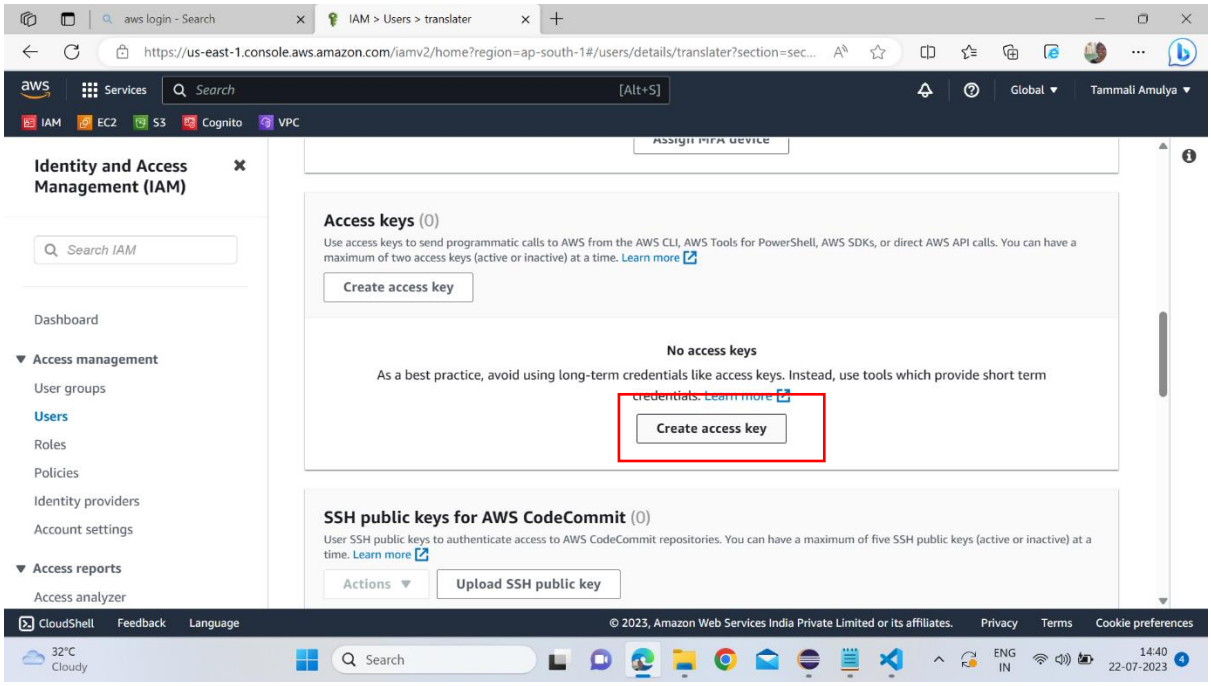
Select previously created role and click on update role.



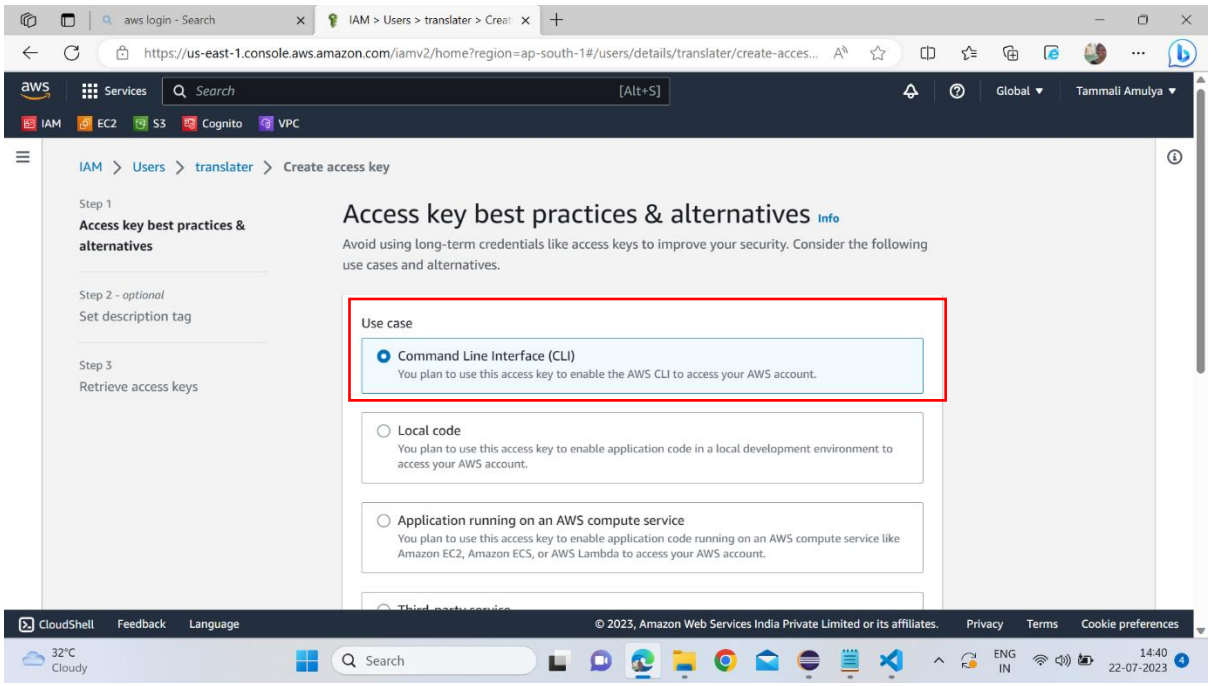
now go to IAM user, go to Security credentials.



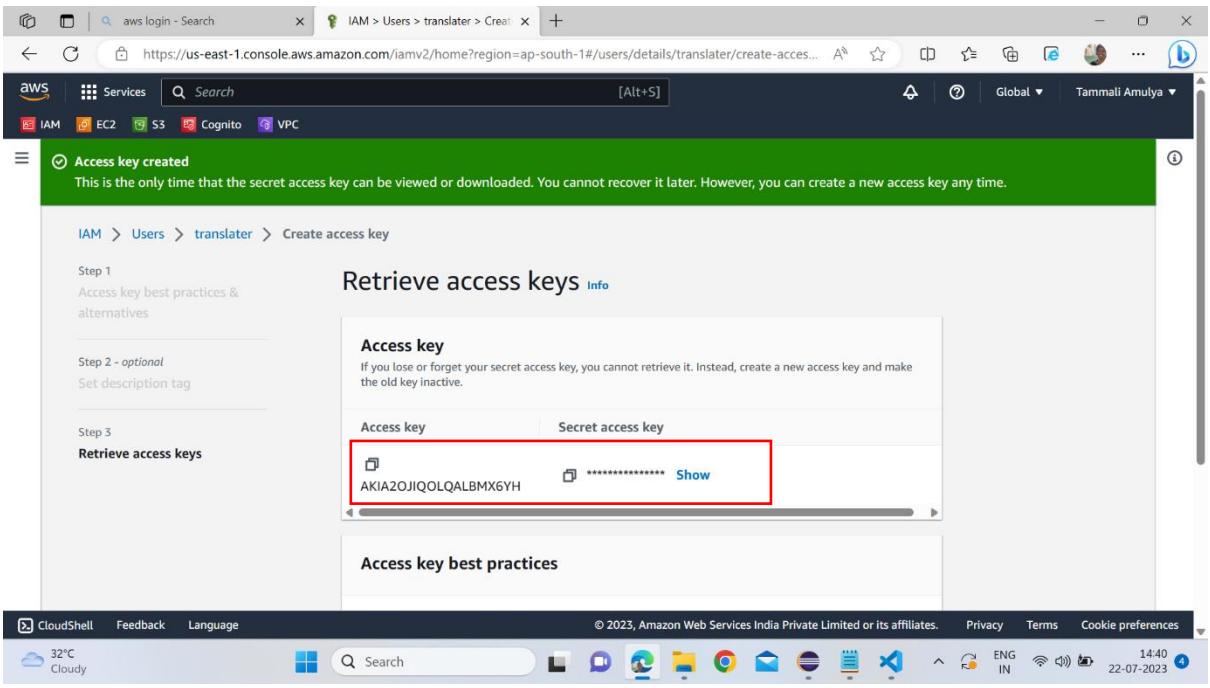
Click on create access key.



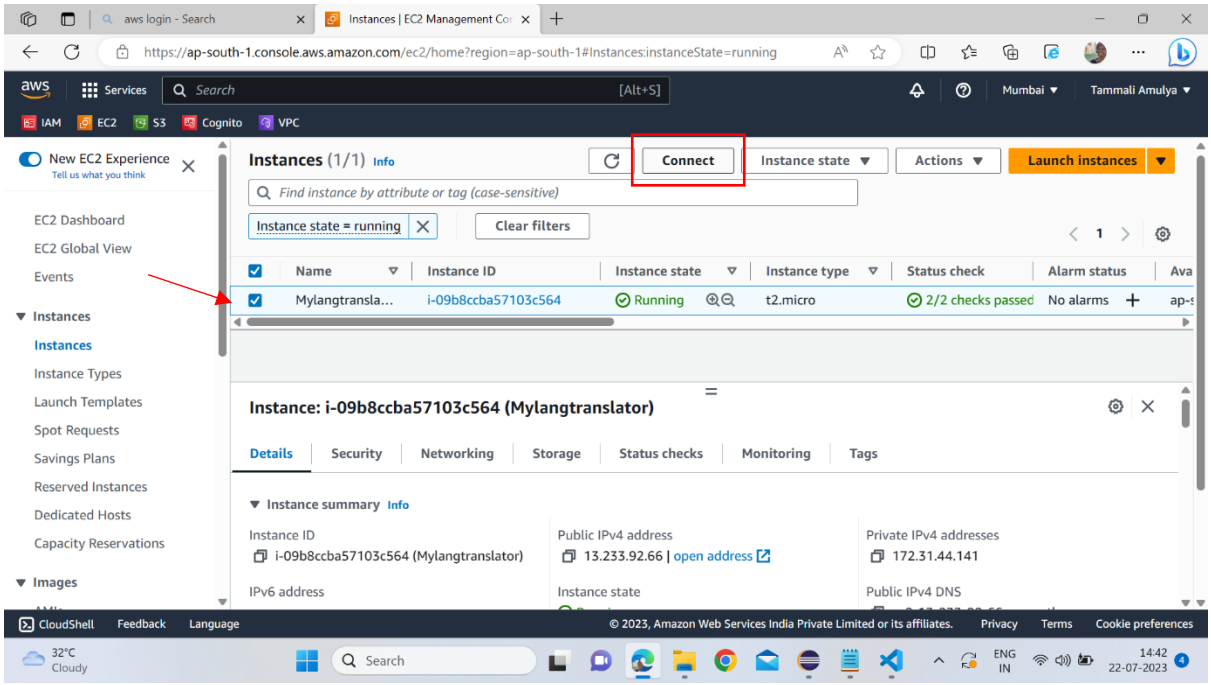
Enable CLI and click on next.



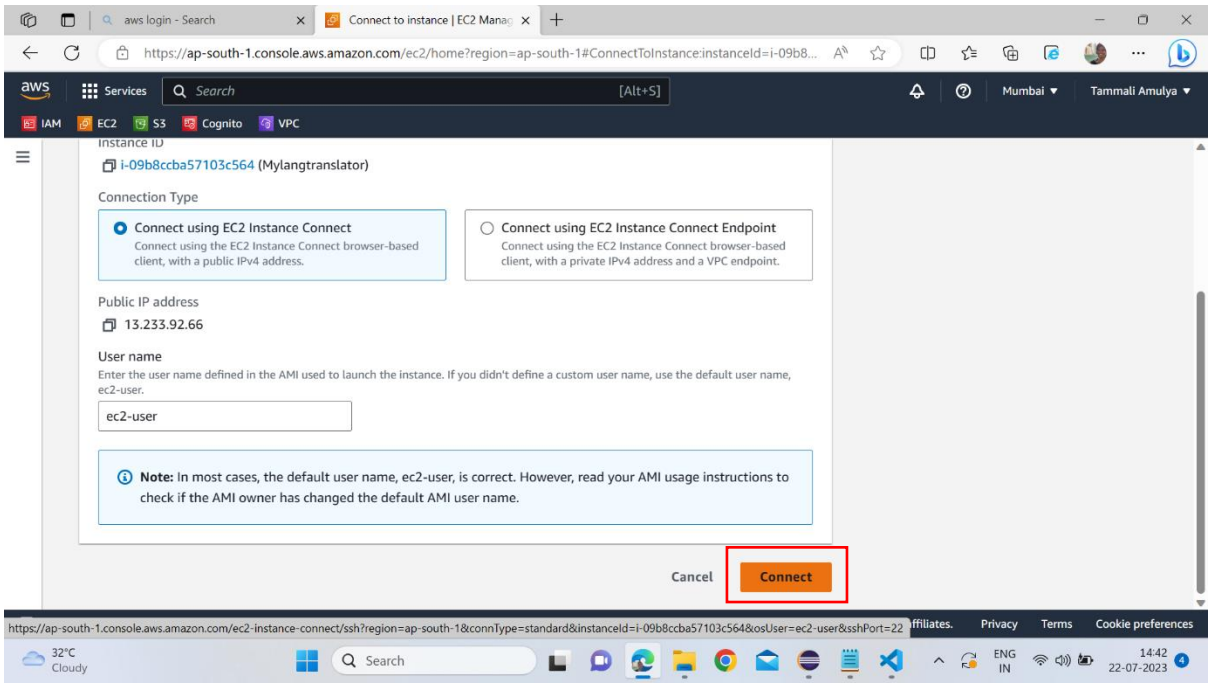
Access key is created.



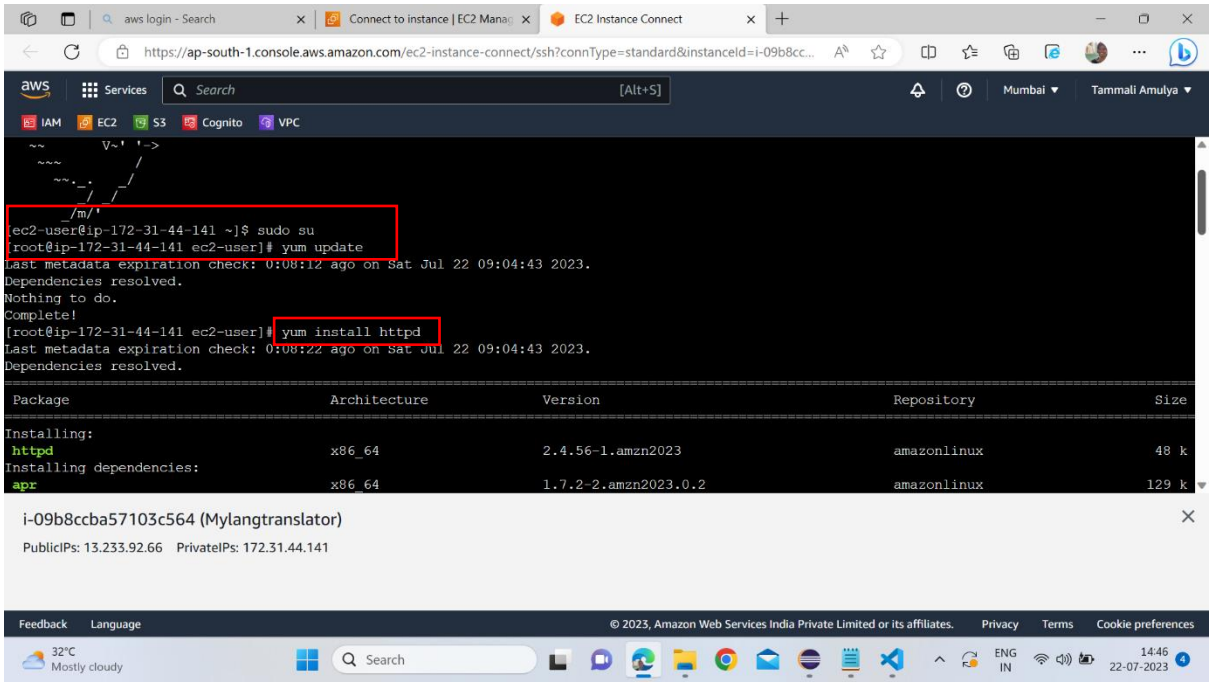
Now go to EC2, select the instance and click on connect.



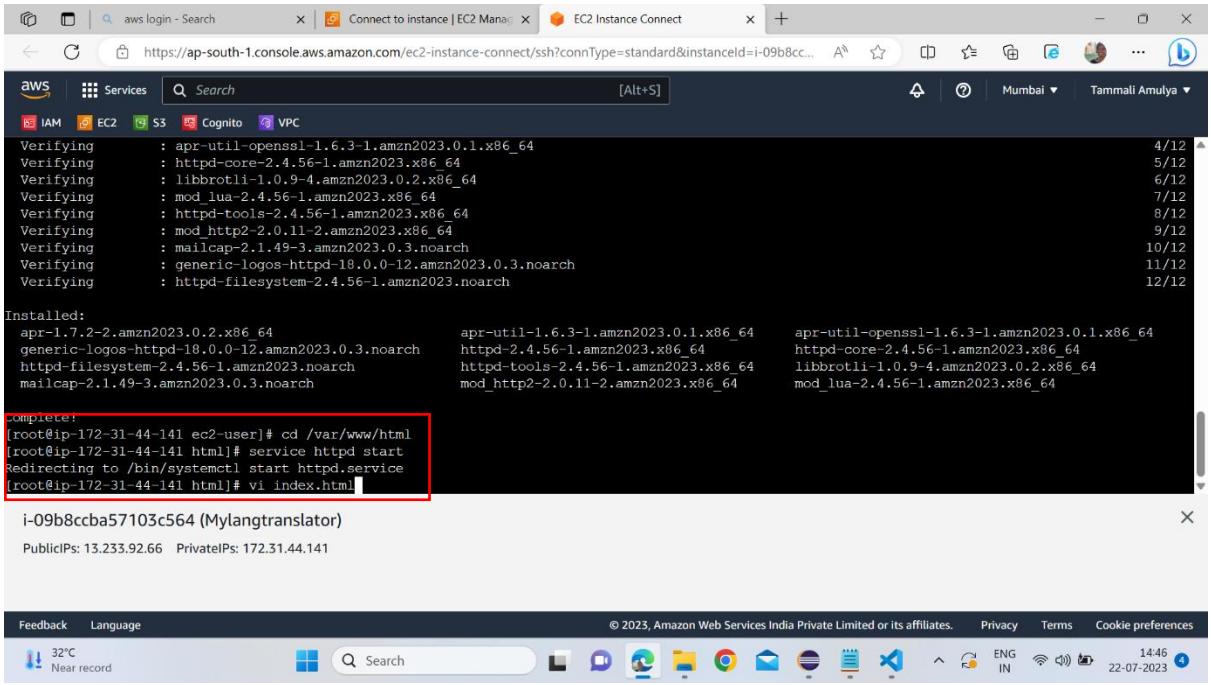
Connect the instance.



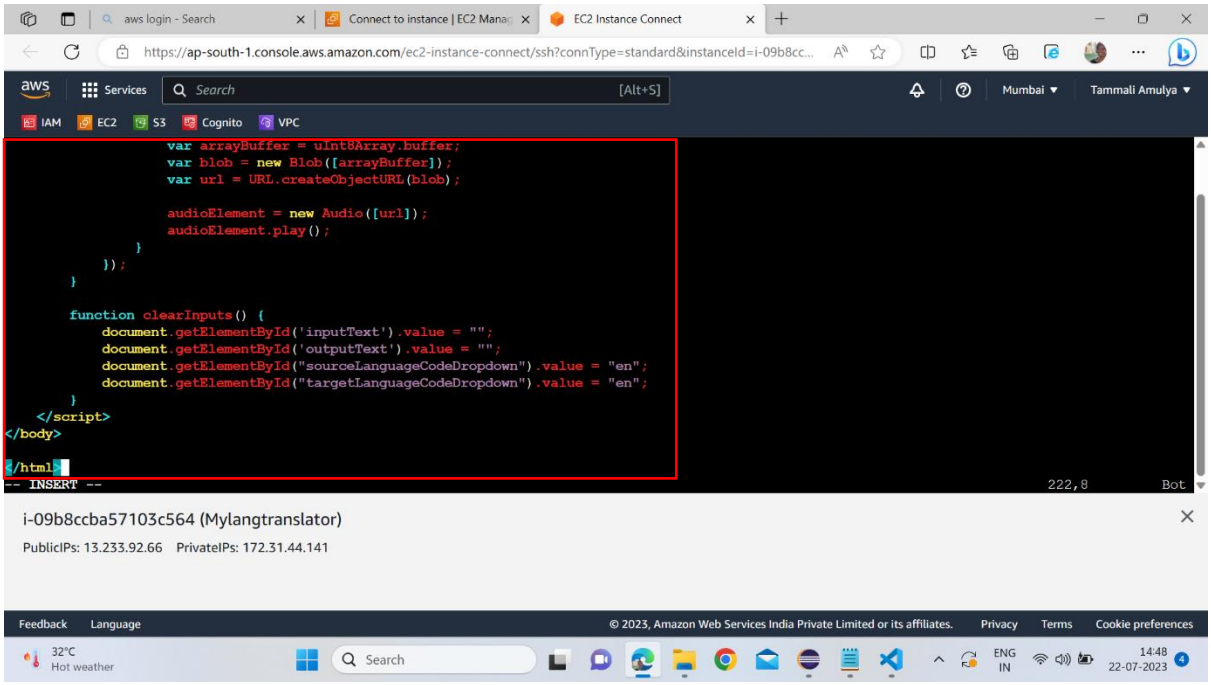
Execute the commands such as: sudo su, yum update, yum install httpd, Service httpd start.



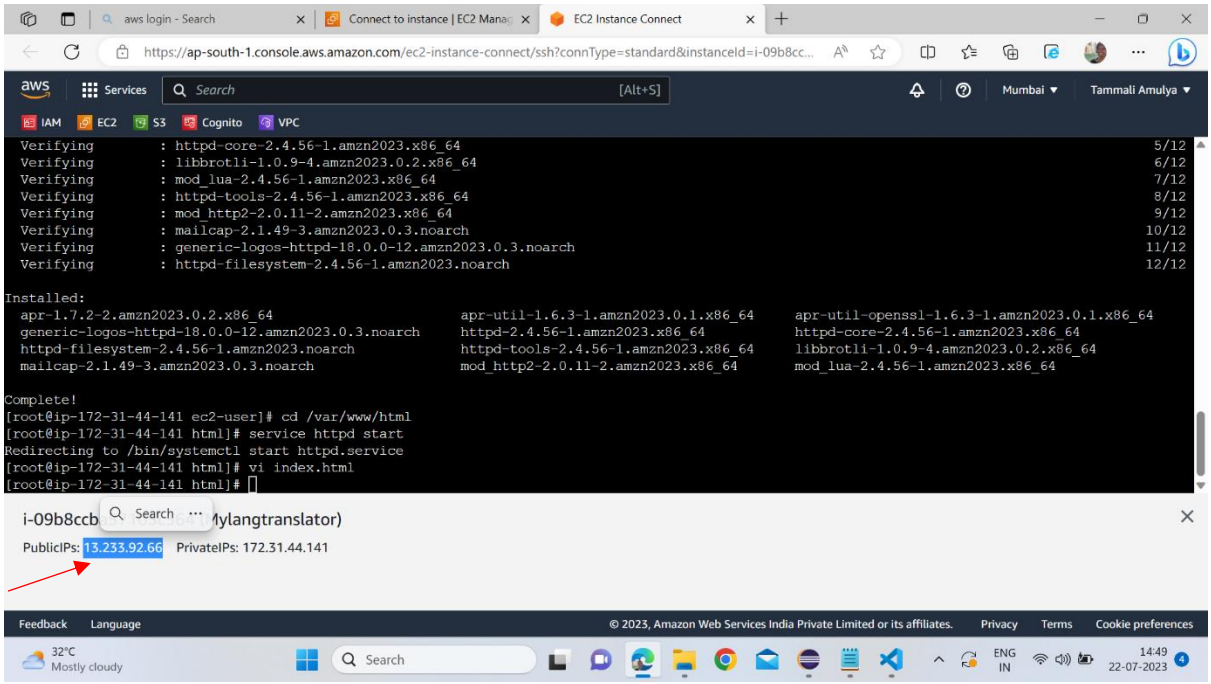
change directory to html and open index.html file by using commands



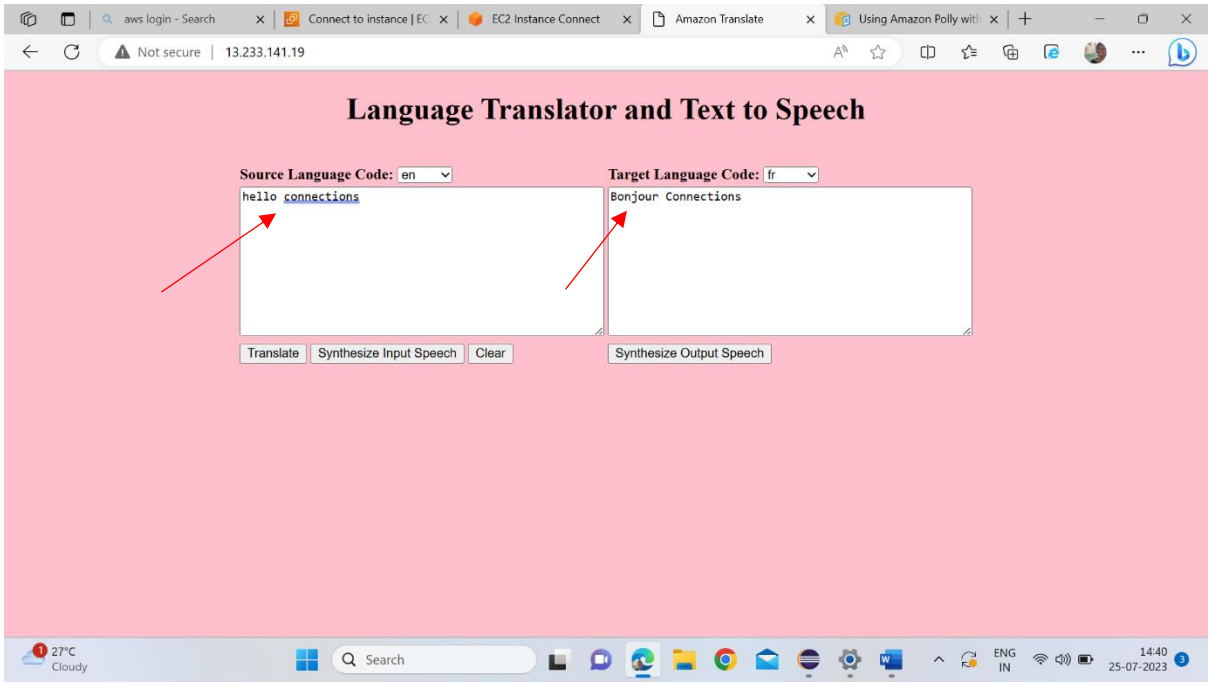
Write html with JS code by connecting to our instance region and user with previously created access key.



Save the file and copy the public IP of instance.



Browse it on a browser and open the index file. Give source text and translate the text into target language and check the speech.



DECLARATION

I hereby declare that "Language translator and text to speech" is the result of the project work carried out by me under the guidance of Mr. Guru Santhosh seenivasan.

DATE: SIGNATURE:

29/07/2023 T.Amulya, P.Pallavi.