Aim -

To study and implement Identity and Access Management (IAM) practices on AWS/Azure cloud.

Theory -

Identity as a Service (IDaaS) is a cloud-based service model that provides organizations with identity and access management (IAM) capabilities over the internet. In IDaaS, organizations can manage user identities, authentication, authorization, and access control policies centrally through a cloud-based platform, rather than deploying and managing IAM infrastructure on-premises.

Advantages of Identity as a Service (IDaaS):

- Centralized Identity Management: IDaaS enables organizations to centrally manage user identities, access controls, and authentication mechanisms across multiple applications and services. This centralization simplifies identity management tasks and improves administrative efficiency.
- Scalability: IDaaS platforms offer scalability features that allow organizations to easily scale their identity management infrastructure up or down based on changing requirements. This elasticity enables organizations to accommodate growing user populations and expanding application portfolios without the need for manual intervention.
- Cost Efficiency: IDaaS follows a pay-as-you-go pricing model, where organizations only pay for the identity management services and resources they consume. This eliminates the need for upfront investment in IAM infrastructure, reducing capital expenditure and enabling cost-effective scaling based on demand.
- Improved Security: Many IDaaS providers offer advanced security features, such as multi-factor authentication (MFA), single sign-on (SSO), identity federation, and risk-based authentication, to enhance security posture and protect against unauthorized access and data breaches.
- Enhanced User Experience: IDaaS platforms provide seamless and consistent authentication and access experiences for users across multiple applications and devices. Features such as single sign-on (SSO) and self-service password reset improve user convenience and productivity while reducing helpdesk support overhead.

Disadvantages of Identity as a Service (IDaaS):

- Data Privacy and Compliance Concerns: Storing and managing user identities and access control policies in a third-party cloud environment may raise

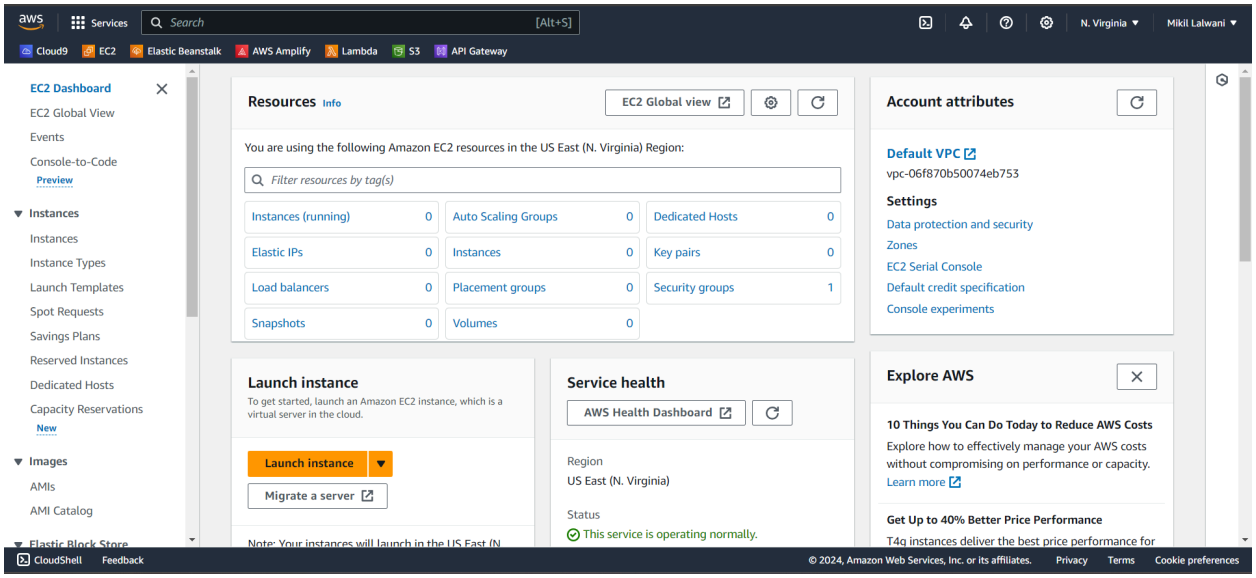
concerns about data privacy and compliance with regulatory requirements, such as GDPR or HIPAA. Organizations must carefully evaluate the data privacy and compliance measures implemented by IDaaS providers to ensure compliance with relevant regulations.

- Dependency on Internet Connectivity: IDaaS relies on internet connectivity for accessing and managing identity management services, which may pose challenges in environments with limited or unreliable internet connectivity. Organizations should consider the potential impact of internet outages or disruptions on their identity management operations.
- Vendor Lock-In: Migrating user identities and access controls between different IDaaS providers or transitioning from IDaaS to an on-premises IAM deployment may be challenging and costly, leading to vendor lock-in. Organizations should consider the long-term implications of vendor lock-in and evaluate strategies to mitigate this risk.
- Customization and Integration Challenges: IDaaS platforms may have limitations in terms of customization and integration with existing IAM systems, directory services, and legacy applications. Organizations with complex identity management requirements may encounter challenges in fully integrating IDaaS solutions into their existing infrastructure.
- Performance and Reliability: Performance and reliability issues may arise in IDaaS environments, particularly in multi-tenant deployments where resources are shared among multiple users. Organizations should assess the performance characteristics and service level agreements (SLAs) of IDaaS offerings to ensure they meet their performance and reliability requirements.

Procedure-

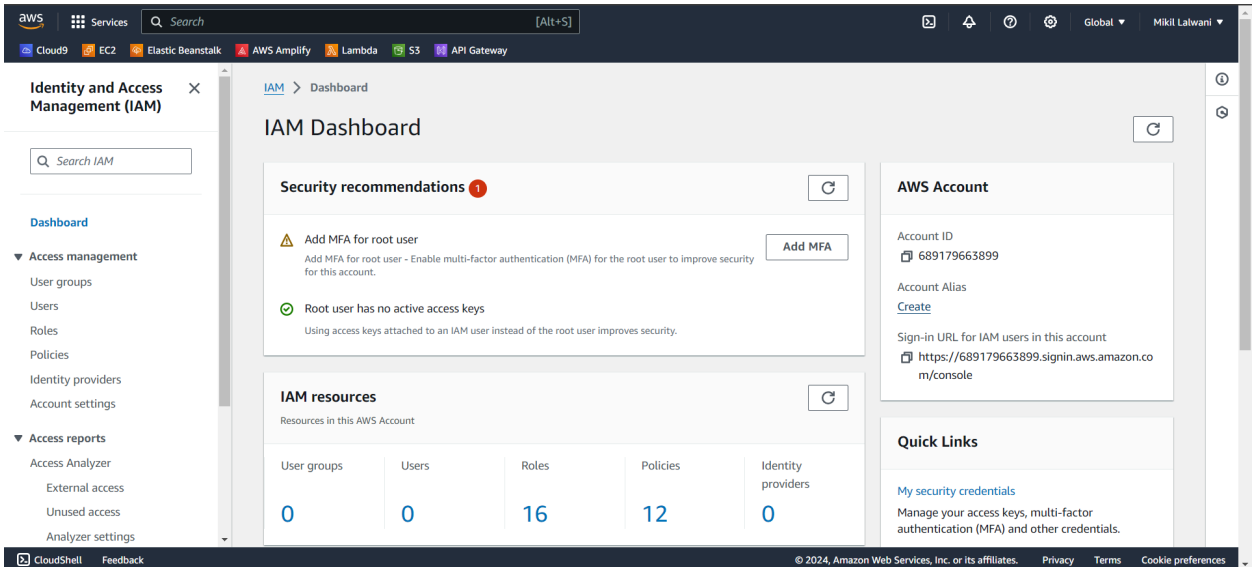
Login to AWS console

Make sure to check all Ec2 dashboard parameters

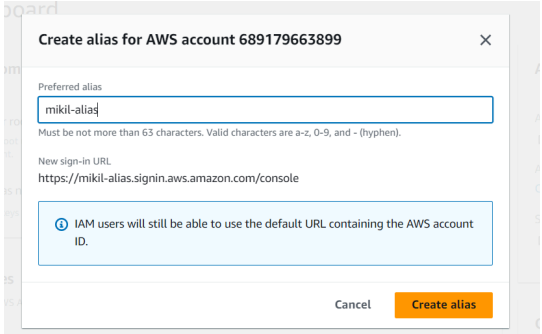


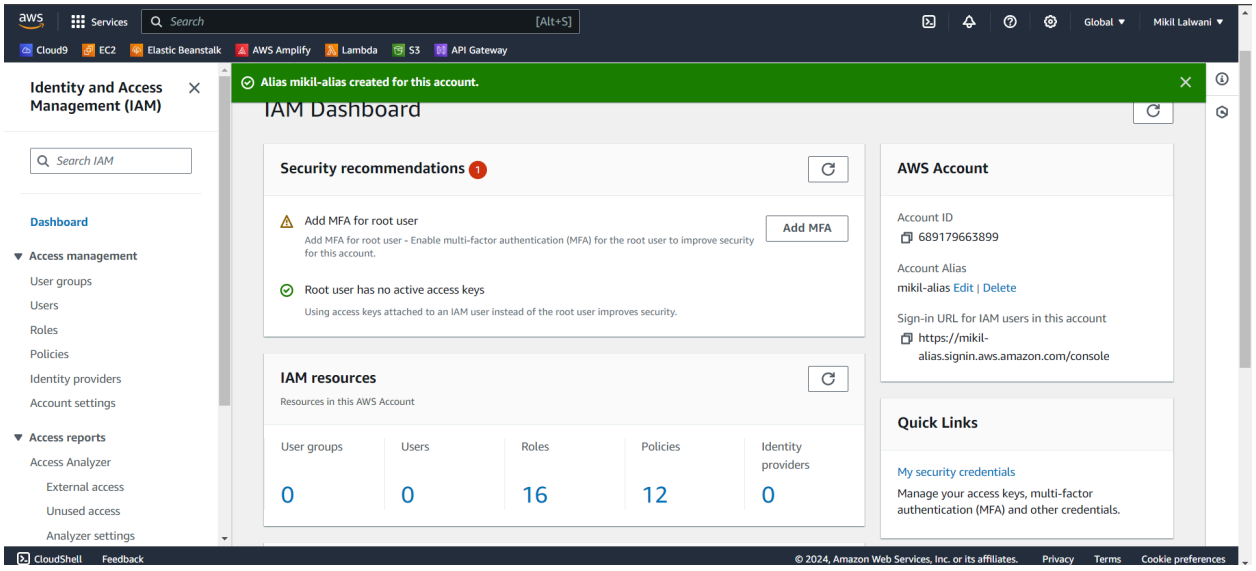
Configuring IAM Dashboard

Go to IAM dashboard



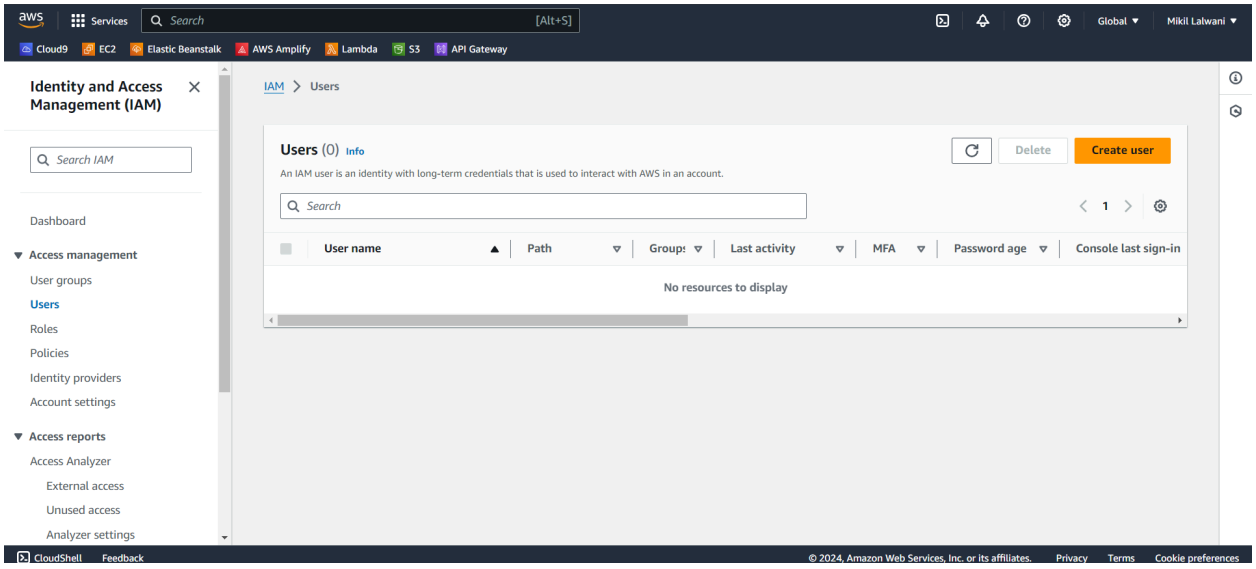
Click on create option under Account Alias and give a valid name; save changes



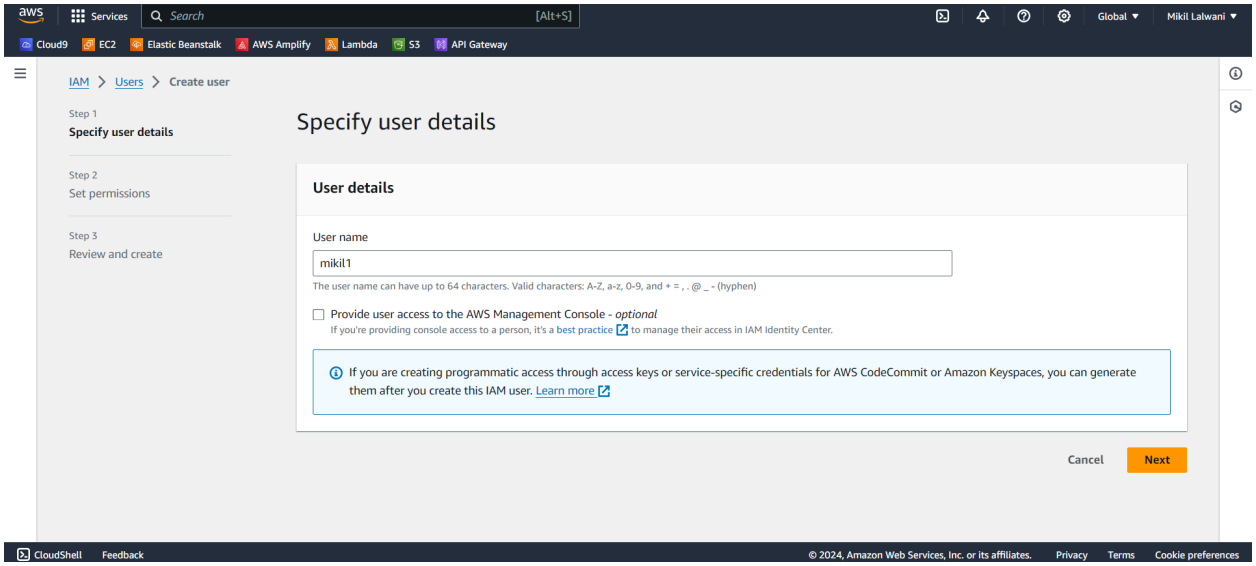


Creating a new User

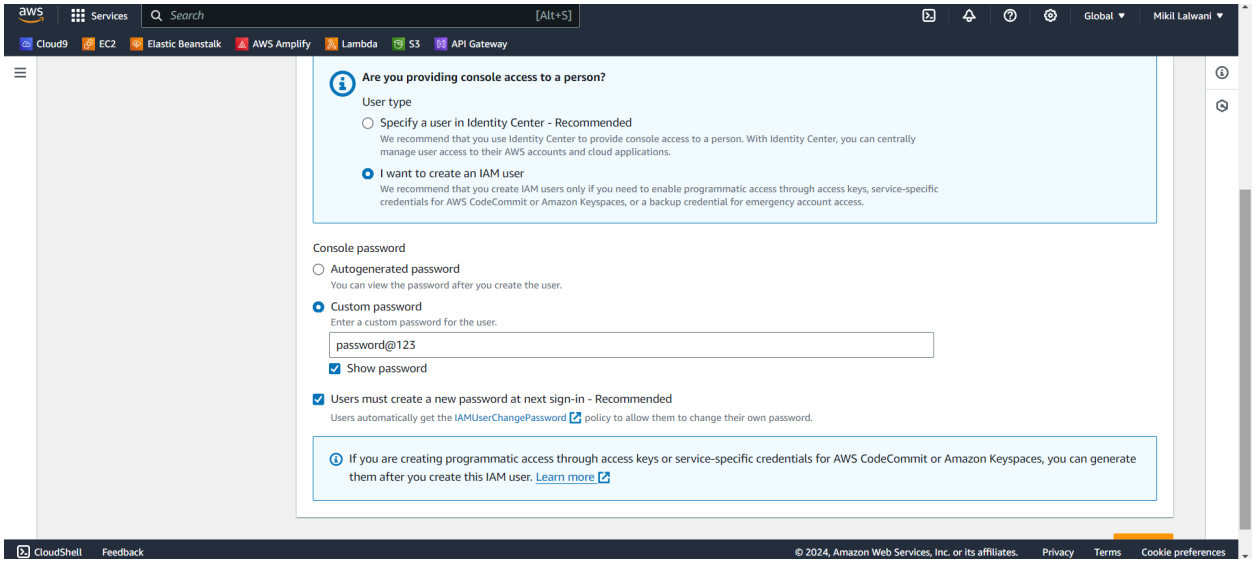
Click on “users” in the left column



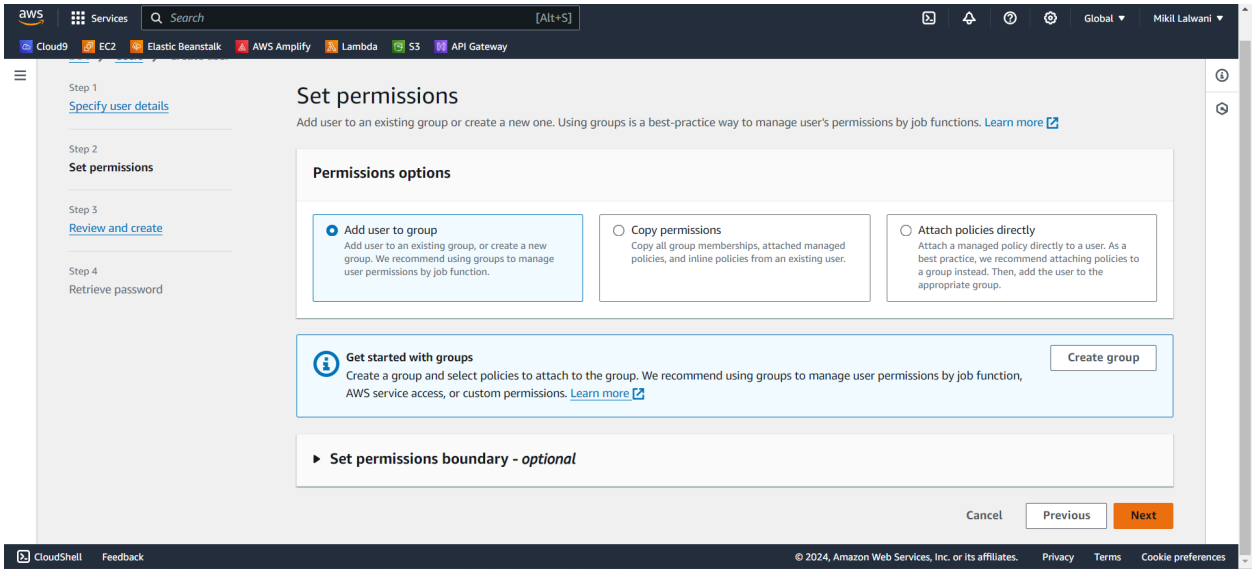
Click on Add users button



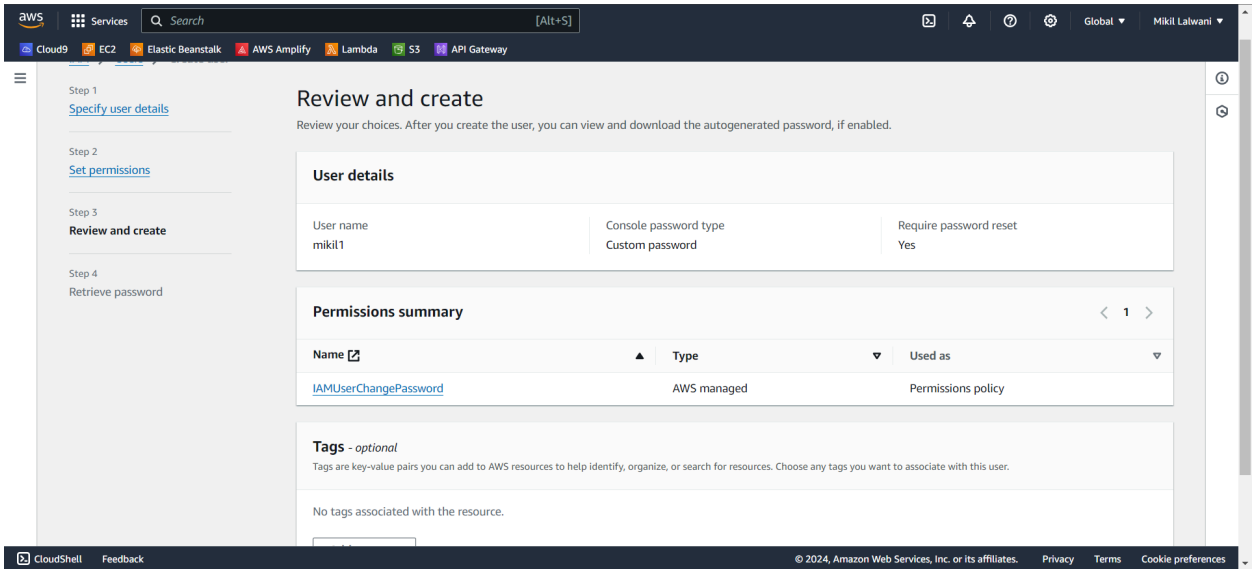
Set a custom valid password (Imc: Qwertyuiop123) and check the Require password reset box which will make you create a next password in the next sign in



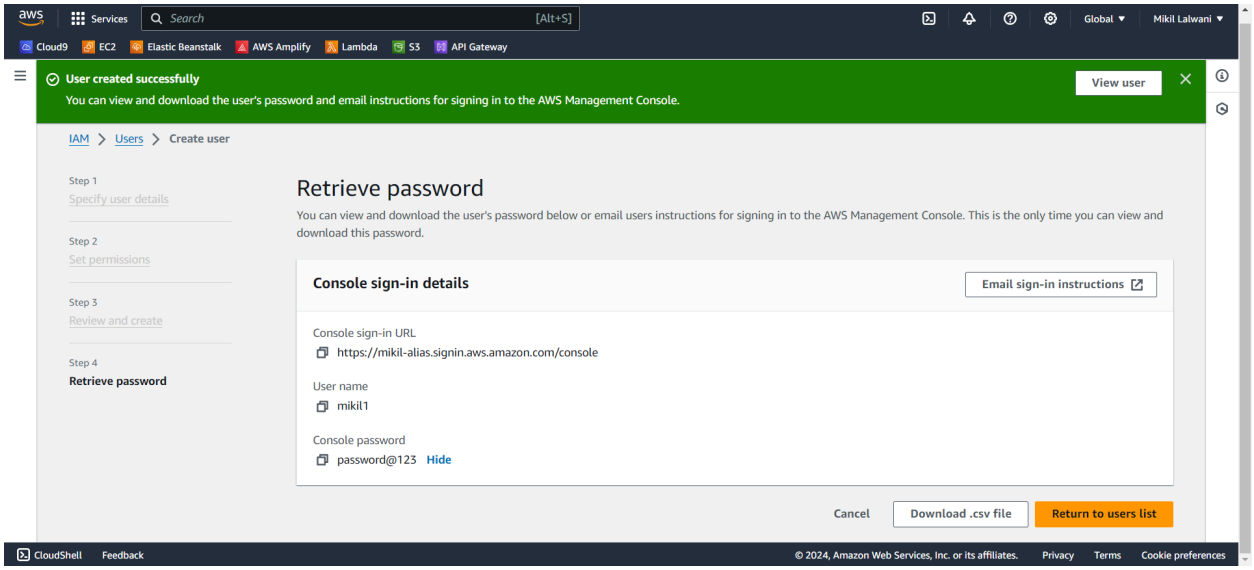
Click on Next



Add a tag if you want to just to keep track of your activities; then click on Next: Review



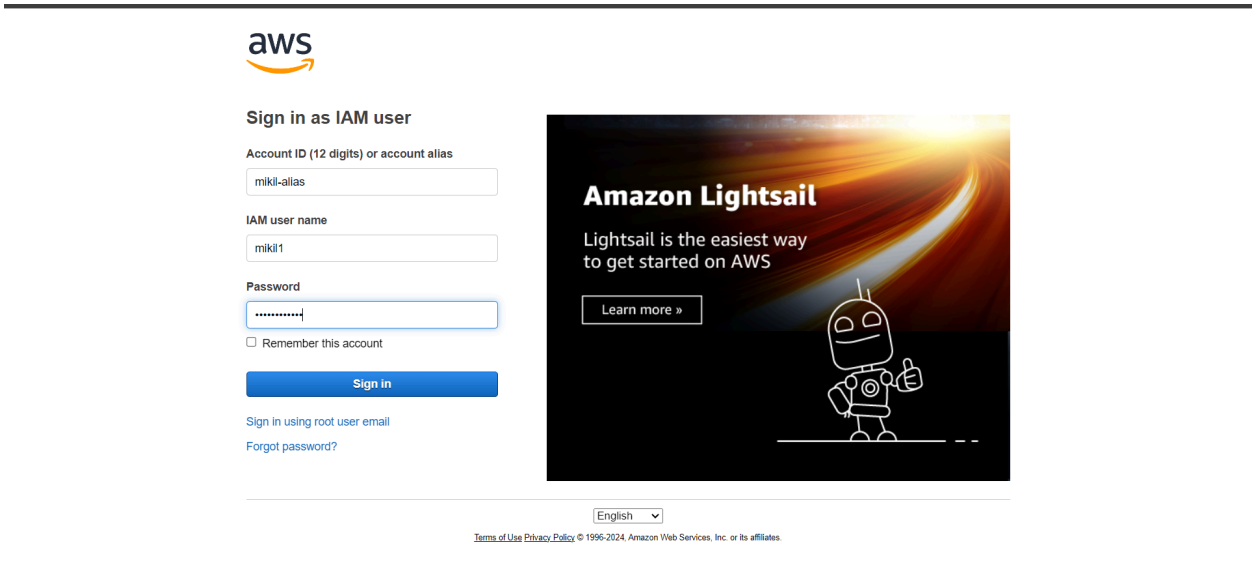
Click on Create User Button



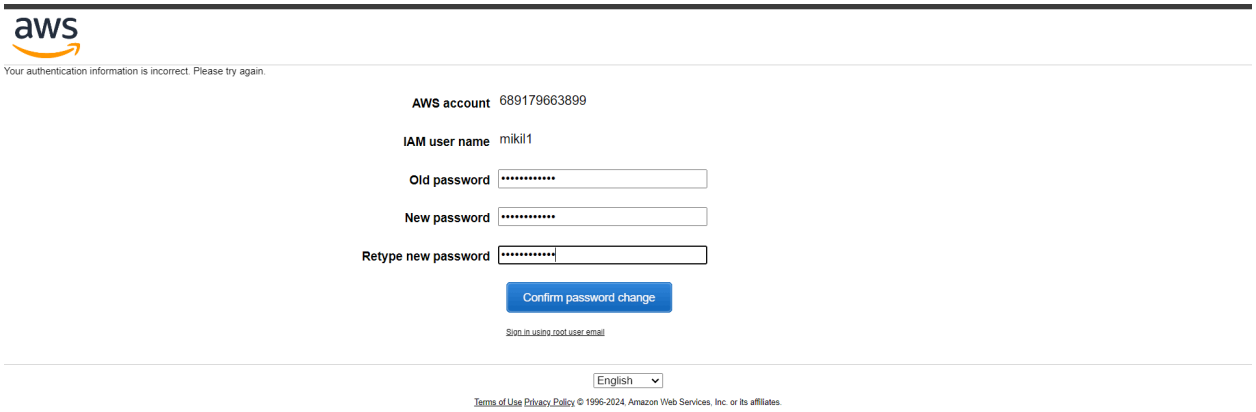
Open the URL in Incognito Mode

Logging in as the new User & Checking their permissions

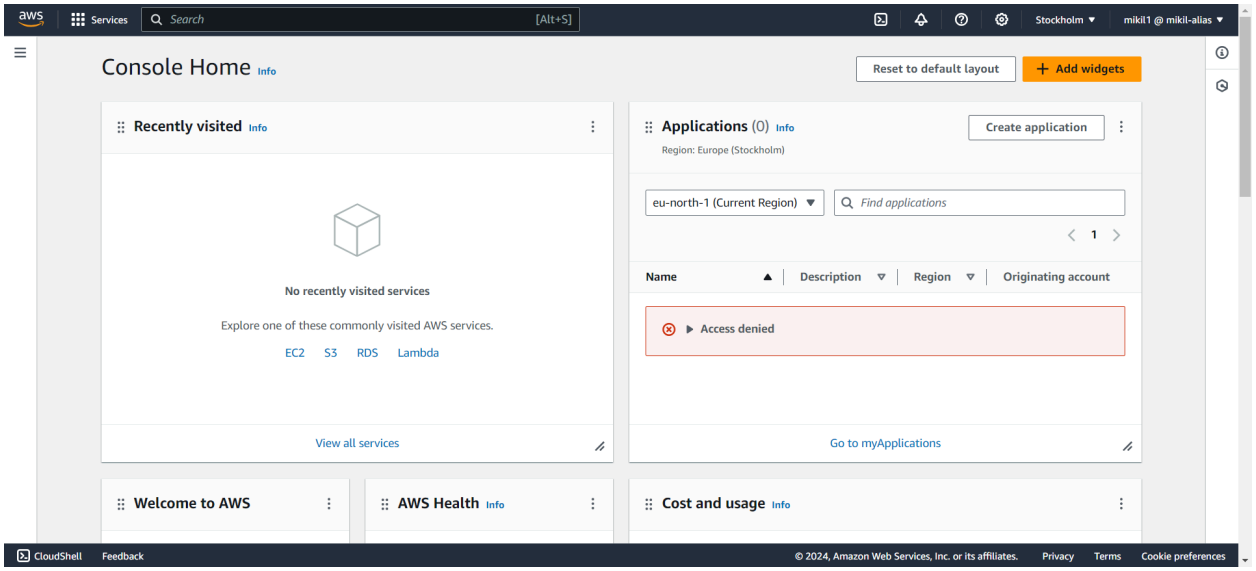
Enter the new user's name and password saved earlier



Enter a new valid password

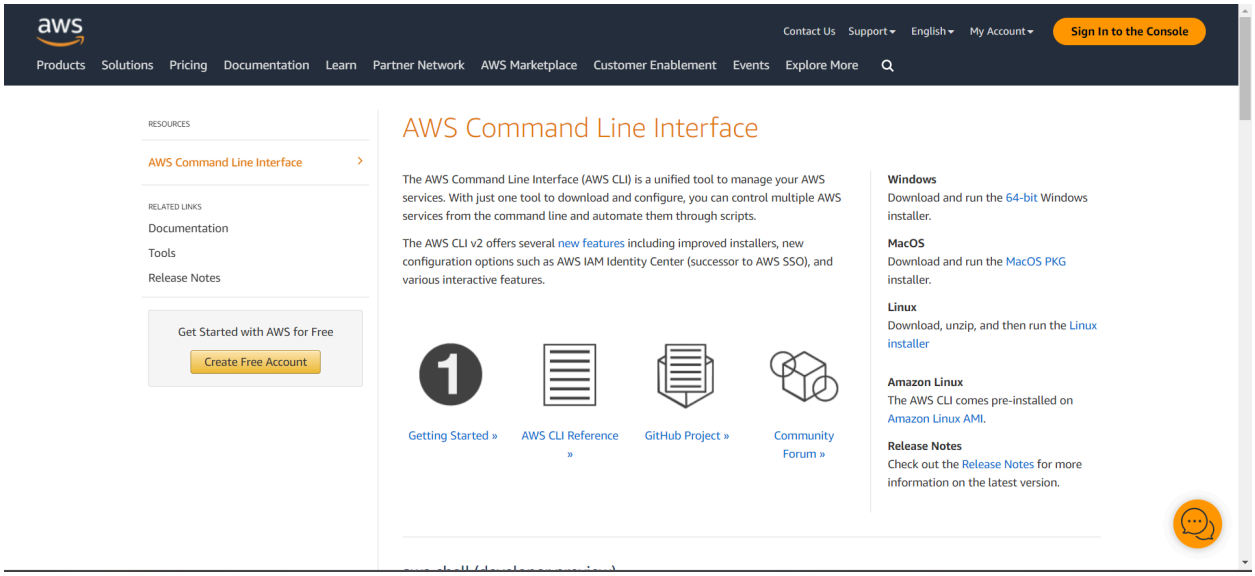


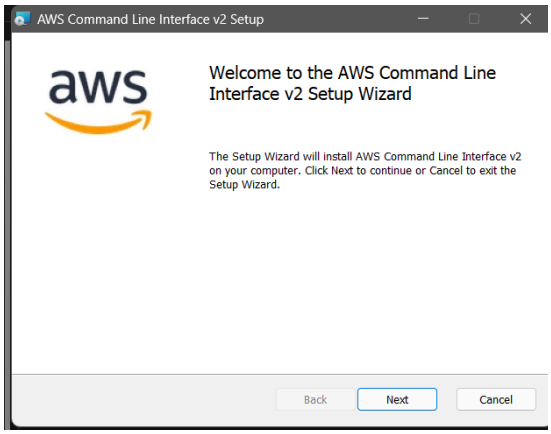
After logging in, you will notice that you don't have permission to do anything yet

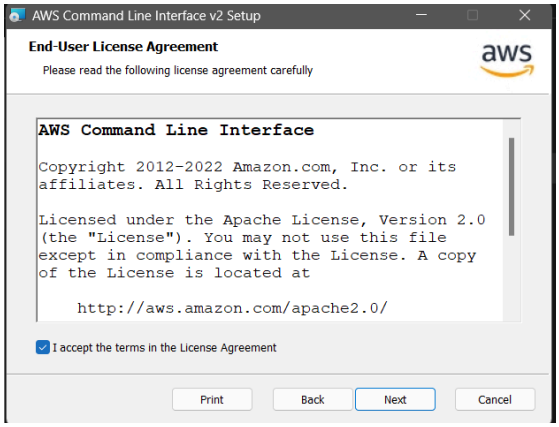


Adding MFA for the user via Root User

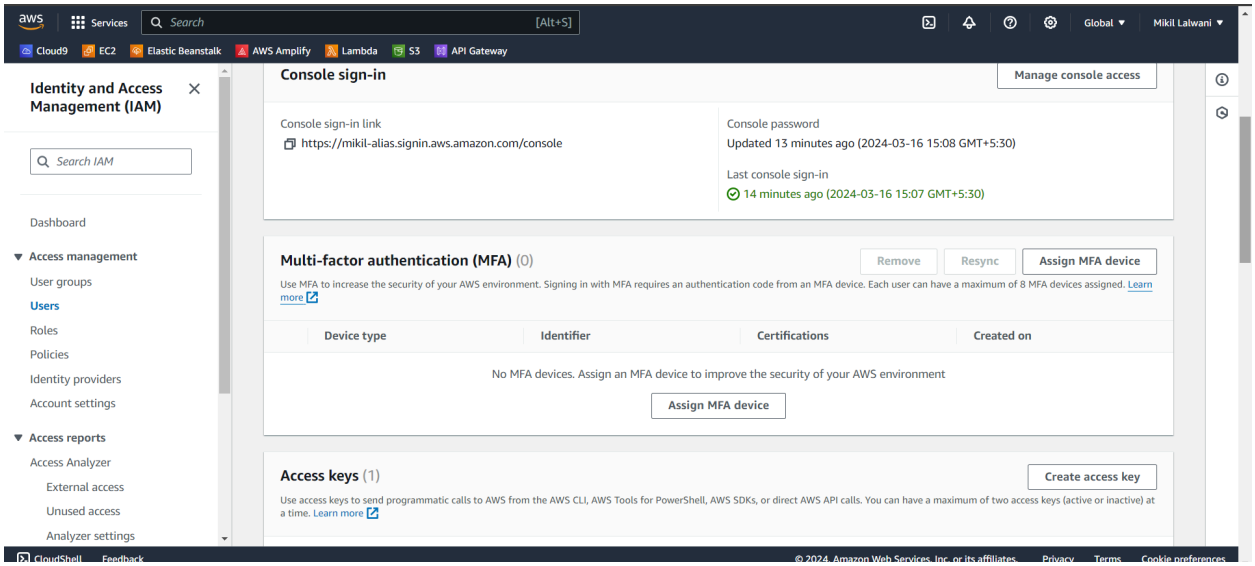
Type "AWS CLI" in a new window of any browser and go to it's the main page of AWS regarding the same.
Click on 64-bit hyperlink in the RHS column under the Windows section and download, install the AWS CLI.

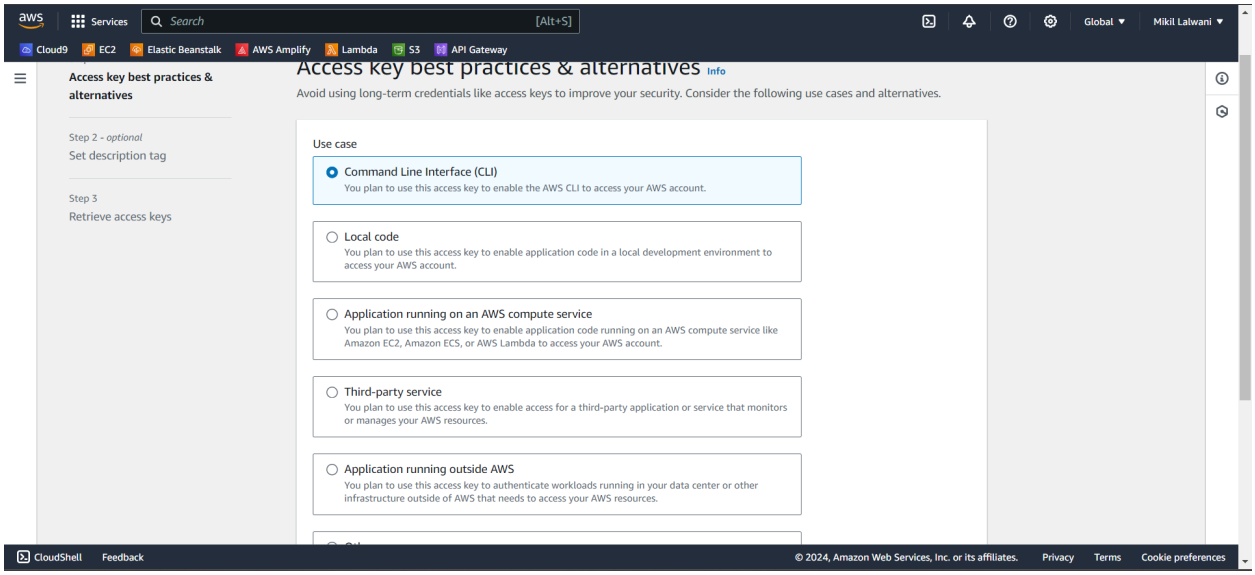


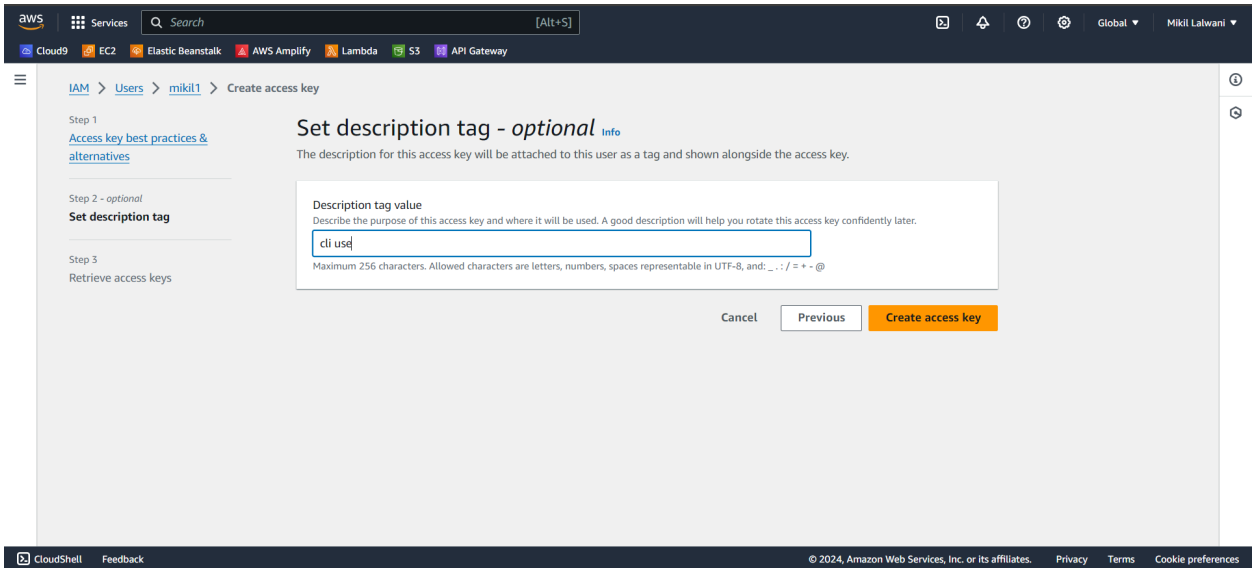




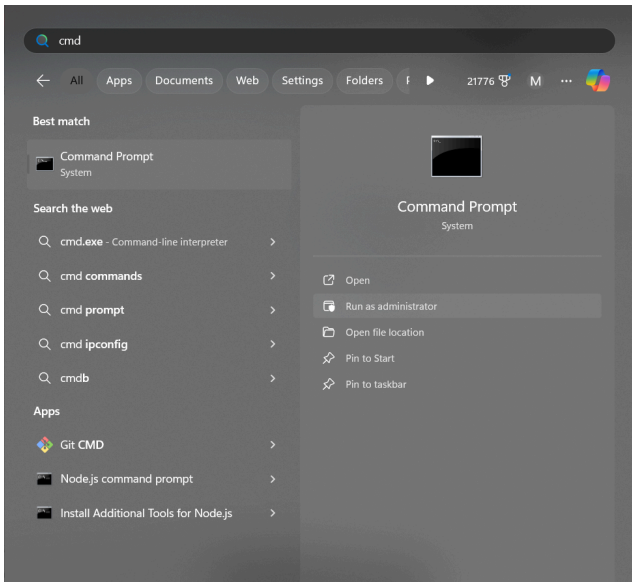
Generate the AccessKeyID and Key needed for AWS CLI.


aws
Services
Search
[Alt+S]
Global
Mikil Lalwani
Cloud9
EC2
Elastic Beanstalk
AWS Amplify
Lambda
S3
API Gateway
Identity and Access Management (IAM)
Search IAM
Dashboard
Access management
User groups
Users
Roles
Policies
Identity providers
Account settings
Access reports
Access Analyzer
External access
Unused access
Analyzer settings
Console sign-in
Manage console access
Console sign-in link
https://mikil-alias.signin.aws.amazon.com/console
Console password
Updated 13 minutes ago (2024-03-16 15:08 GMT+5:30)
Last console sign-in
14 minutes ago (2024-03-16 15:07 GMT+5:30)
Multi-factor authentication (MFA) (0)
Remove
Resync
Assign MFA device
Use MFA to increase the security of your AWS environment. Signing in with MFA requires an authentication code from an MFA device. Each user can have a maximum of 8 MFA devices assigned. Learn more
Device type
Identifier
Certifications
Created on
No MFA devices. Assign an MFA device to improve the security of your AWS environment
Assign MFA device
Access keys (1)
Create access key
Use access keys to send programmatic calls to AWS from the AWS CLI, AWS Tools for PowerShell, AWS SDKs, or direct AWS API calls. You can have a maximum of two access keys (active or inactive) at a time. Learn more
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


aws
Services
Search
[Alt+S]
Global
Mikil Lalwani
Cloud9
EC2
Elastic Beanstalk
AWS Amplify
Lambda
S3
API Gateway
Access key best practices & alternatives
Step 2 - optional
Set description tag
Step 3
Retrieve access keys
Access key best practices & alternatives Info
Avoid using long-term credentials like access keys to improve your security. Consider the following use cases and alternatives.
Use case
Command Line Interface (CLI)
You plan to use this access key to enable the AWS CLI to access your AWS account.
Local code
You plan to use this access key to enable application code in a local development environment to access your AWS account.
Application running on an AWS compute service
You plan to use this access key to enable application code running on an AWS compute service like Amazon EC2, Amazon ECS, or AWS Lambda to access your AWS account.
Third-party service
You plan to use this access key to enable access for a third-party application or service that monitors or manages your AWS resources.
Application running outside AWS
You plan to use this access key to authenticate workloads running in your data center or other infrastructure outside of AWS that needs to access your AWS resources.
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences



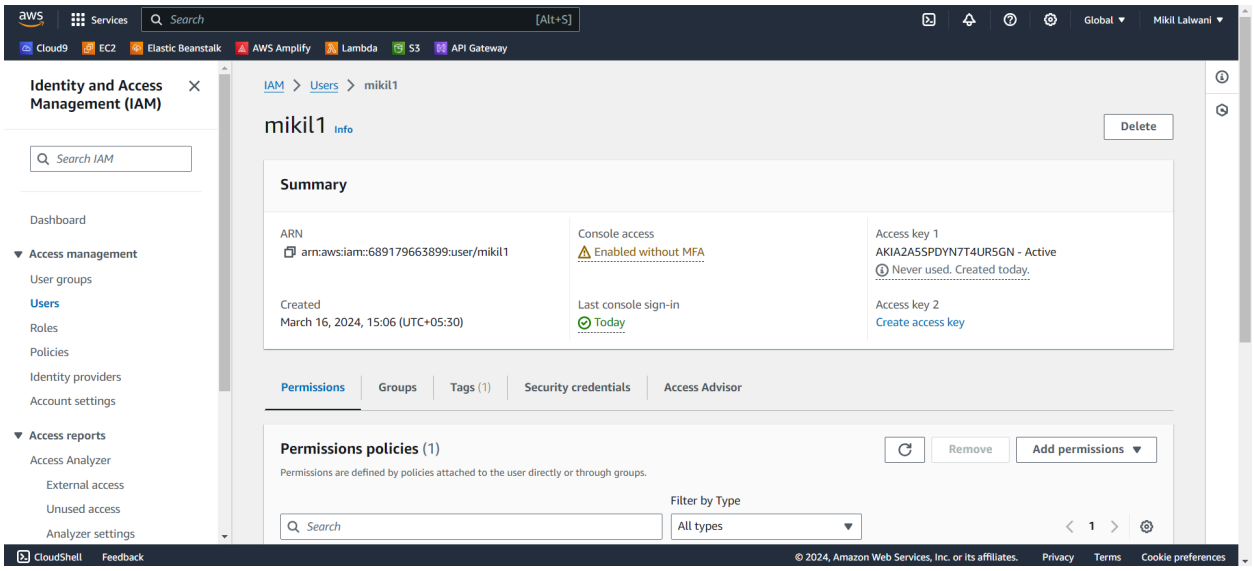
Type “cmd” in the windows search bar and run it as an administrator



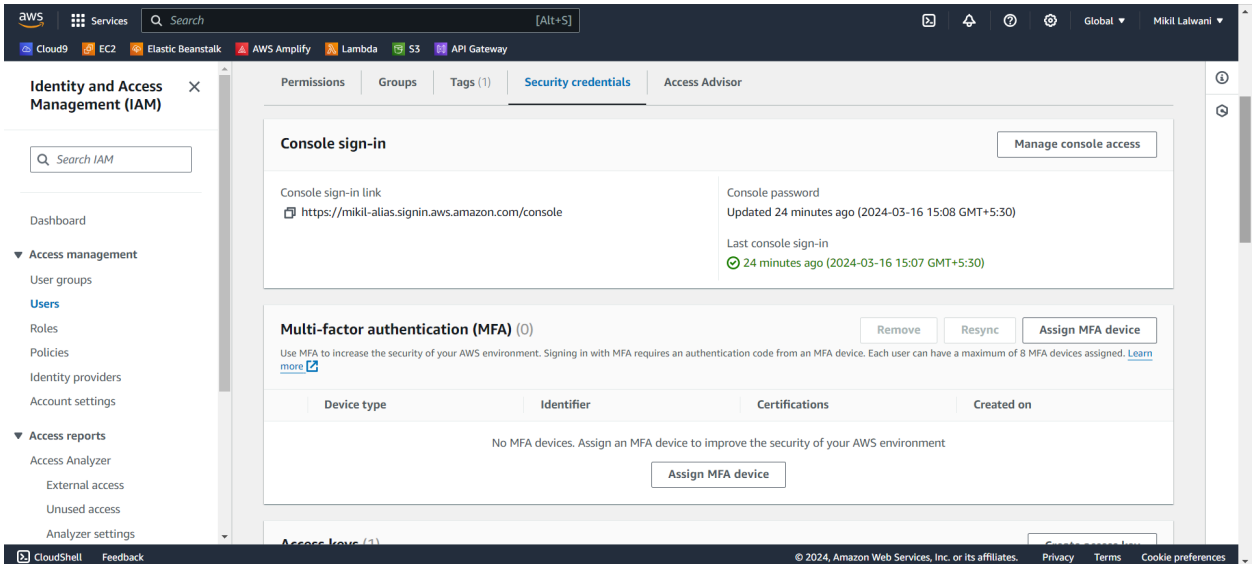
Type aws configure, it will ask for a few inputs;
AWS Access Key ID and Key are the ones which we saved earlier
Default region name is whichever region AWS you are using; in case of Mumbai, its: *ap-south-1*
The output format is *json* in our case

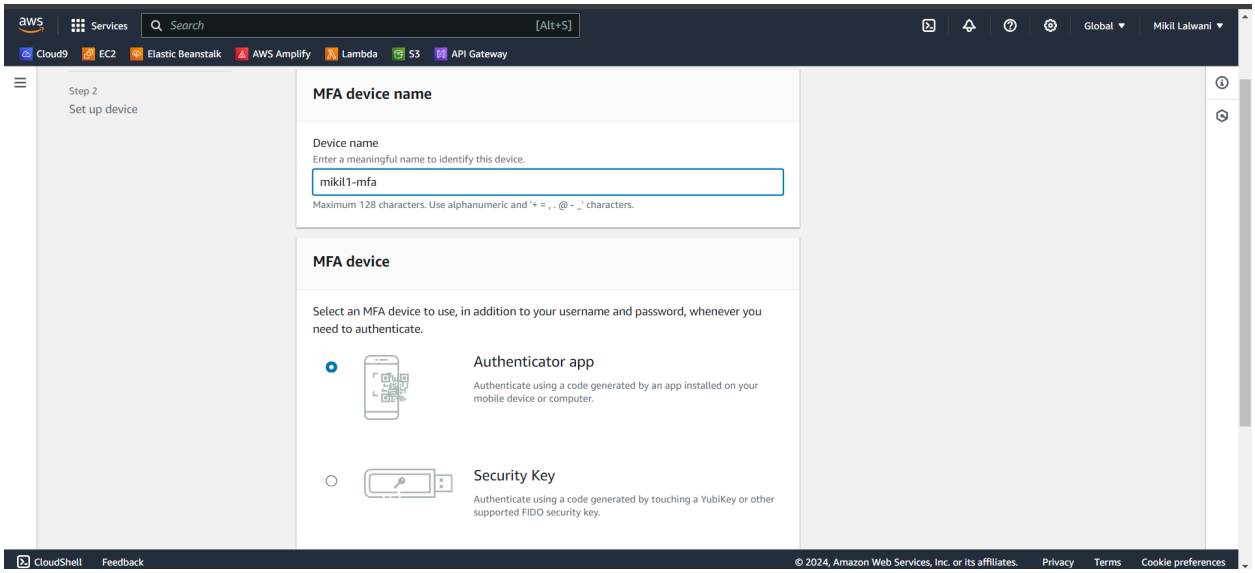
The next two steps are OPTIONAL:
aws –version
aws s3 ls

Go in the security credentials tab under Users of IAM Dashboard

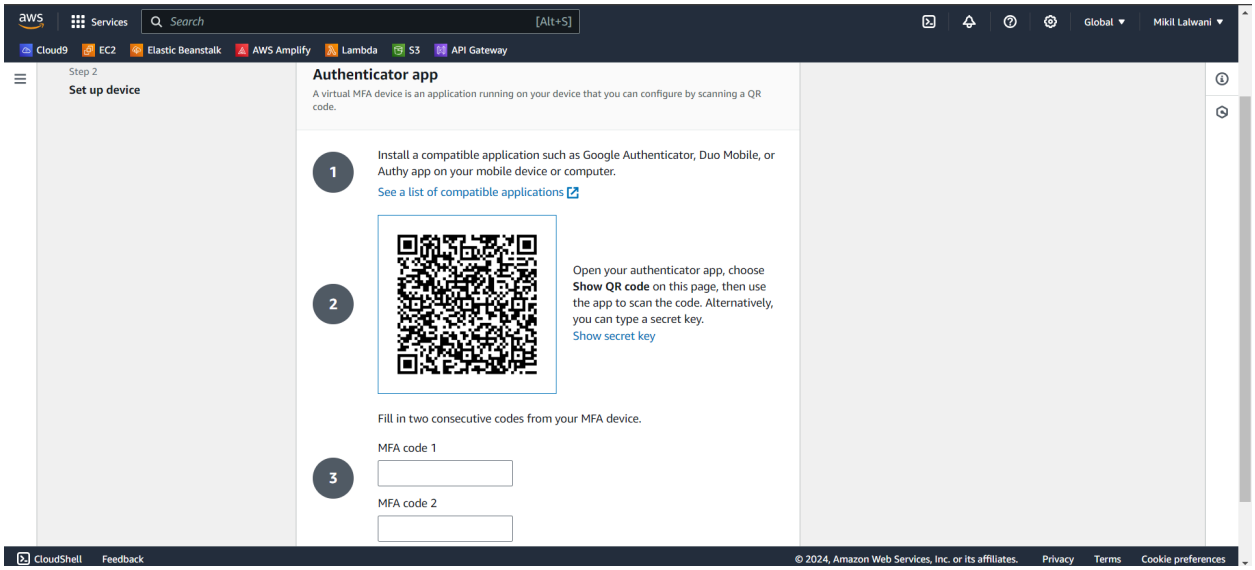


Click on the “Assign MFA Device” Hyperlink

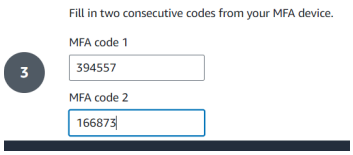


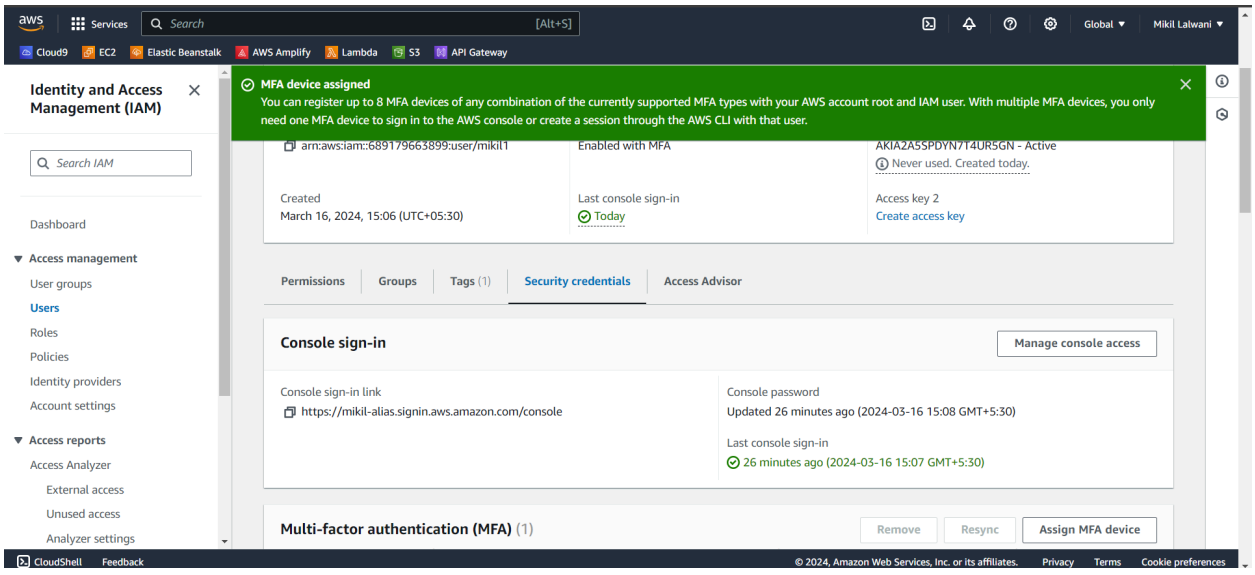


Use the Google Authenticator app downloaded earlier to scan the QR Code



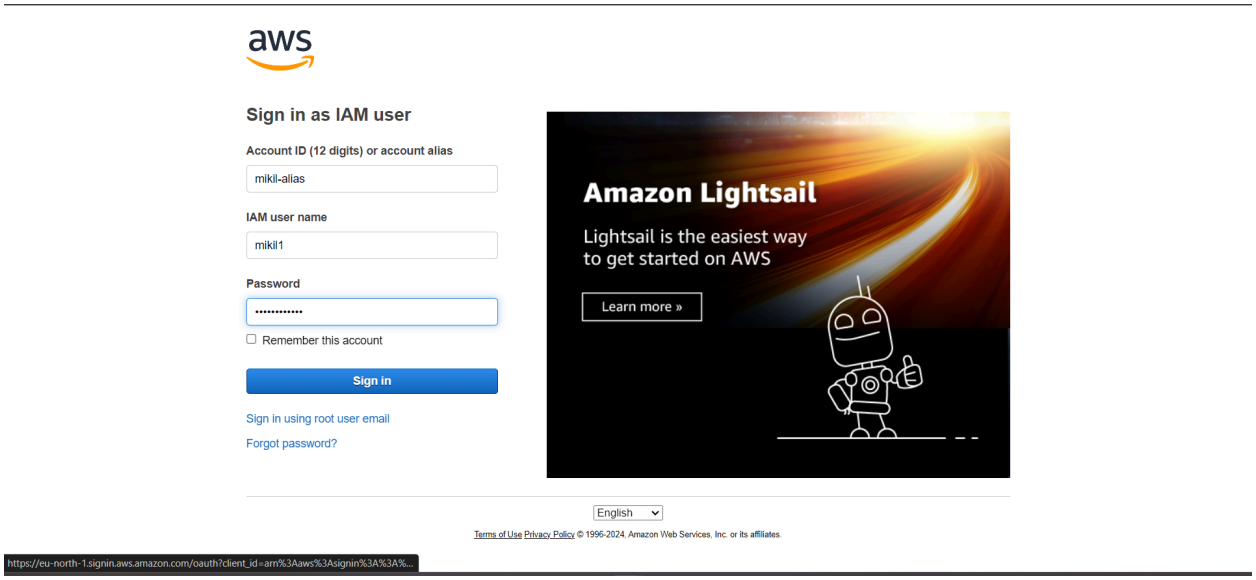
Enter two of the codes which are shown in the Google Authenticator App over a span of 30 secs each; click on Assign MFA Button



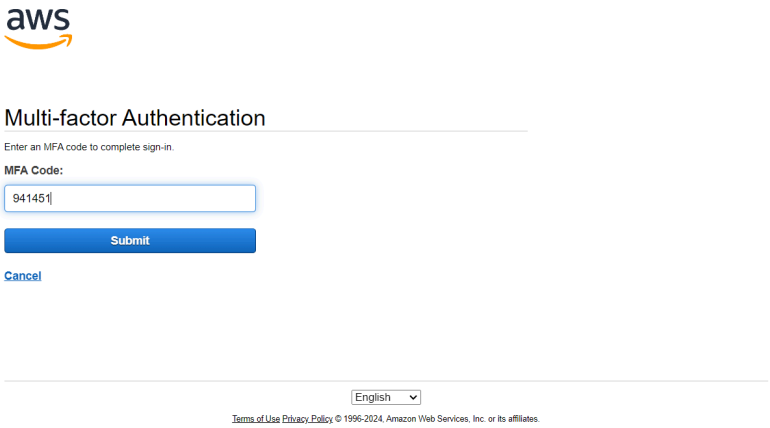


Logging in as the new user after MFA

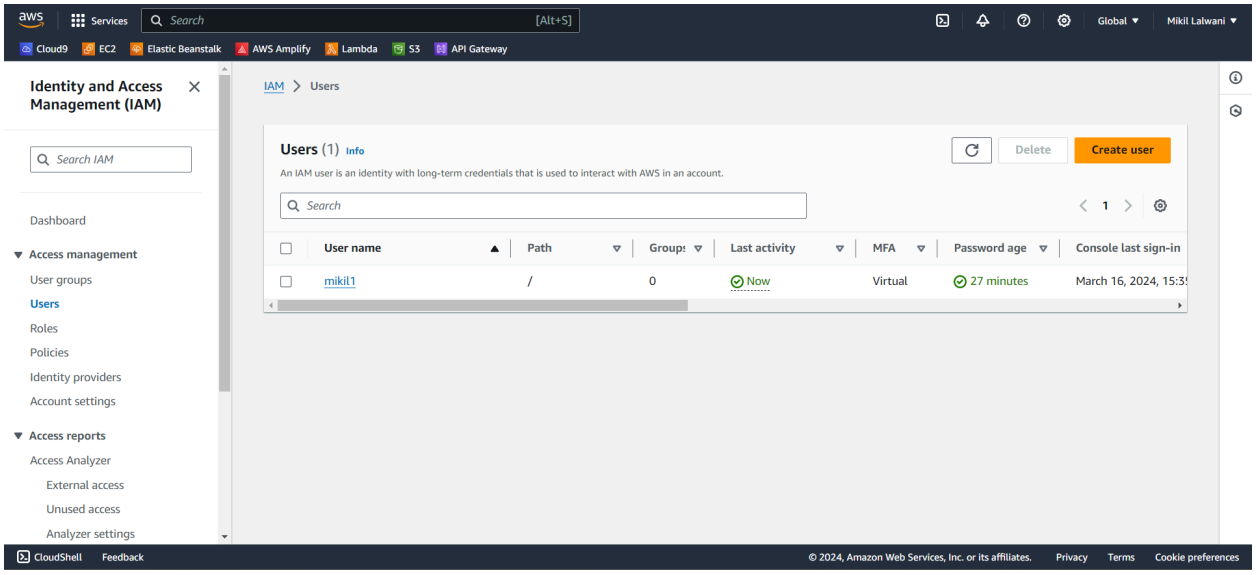
Again try logging in via the new user created earlier; this time it will ask for MFA after you click on Sign In



Use the code being shown in the Google Authenticator



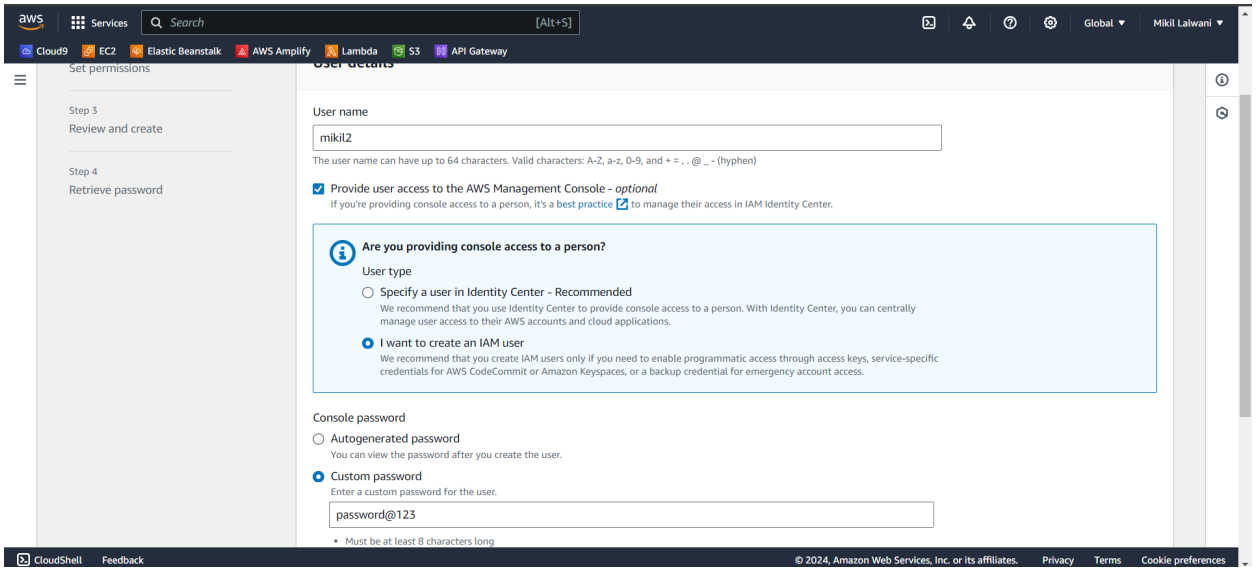
Now, after opening the root user window again
After going in the Users section of IAM Dashboard, we can see that MFA has been activated for the new user



Adding 2 More Users and Giving them permissions

Now, Adding 2 More Users

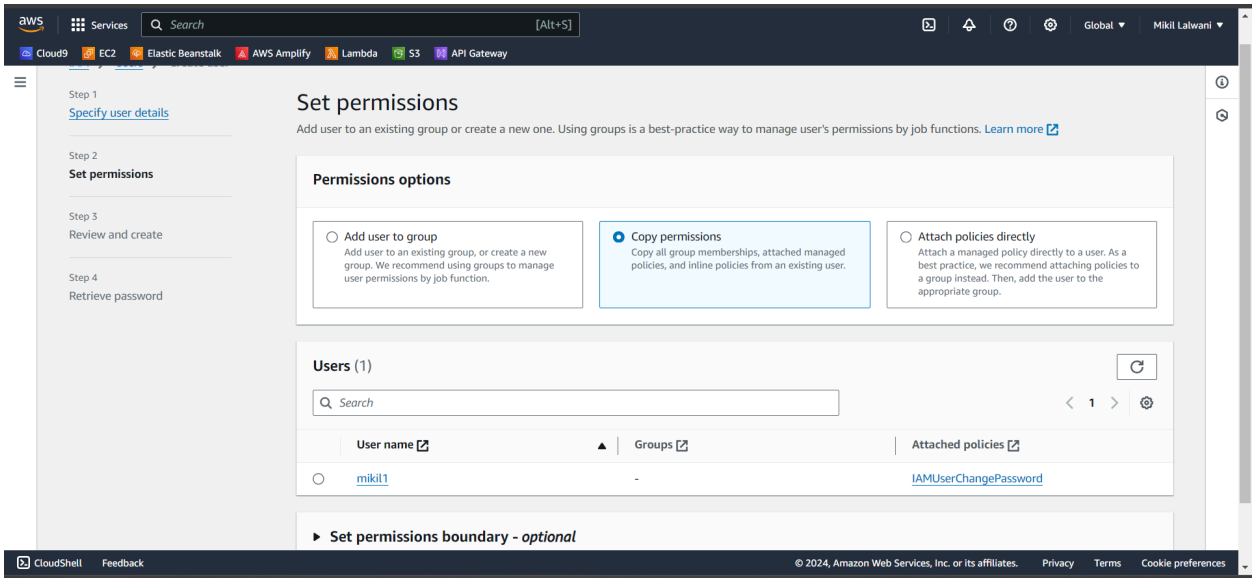
Not giving them an Access key and not checking the Psw Reset Checkbox; Click on the Next: Permissions



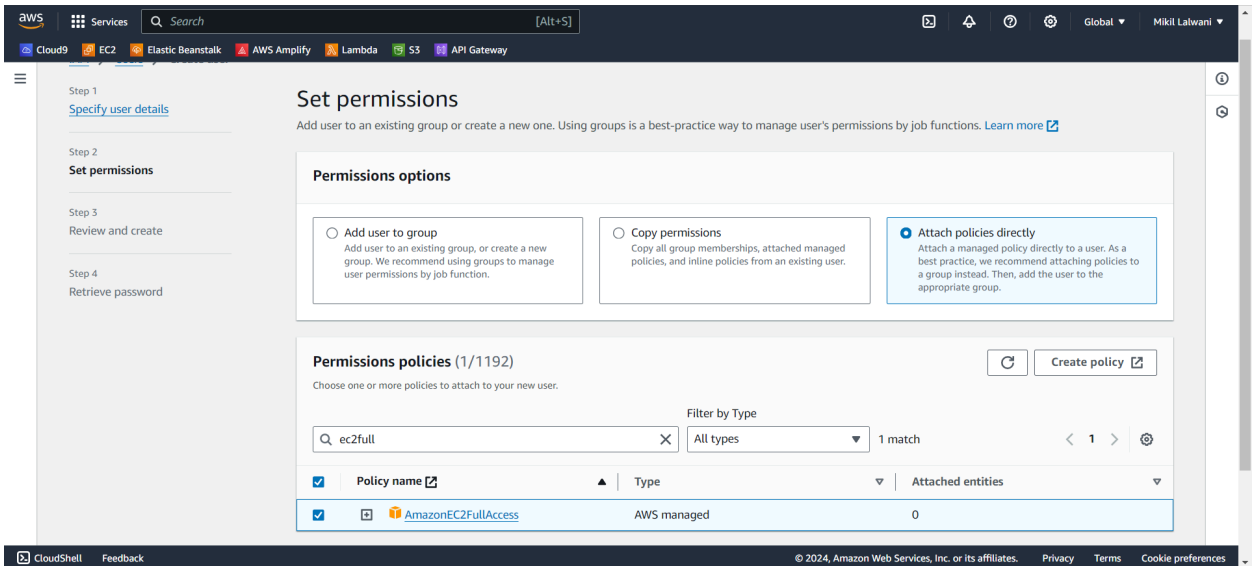
We will create a group later

We can see the previous user listed under the copy “permission from existing user” section (just for observation purpose)

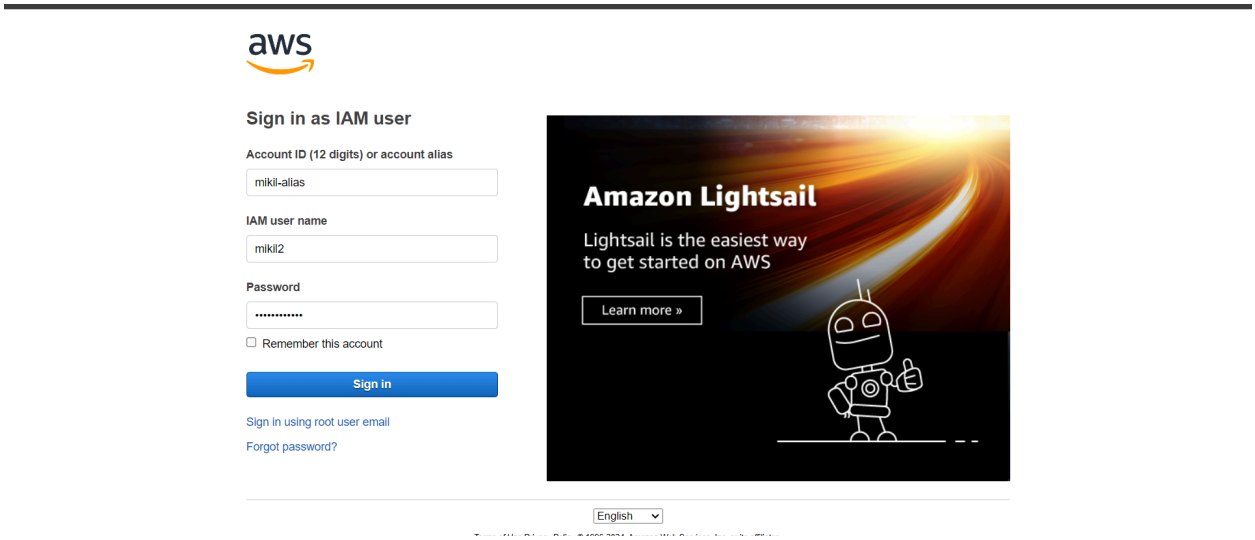
Click on the third section: Attach existing policies directly



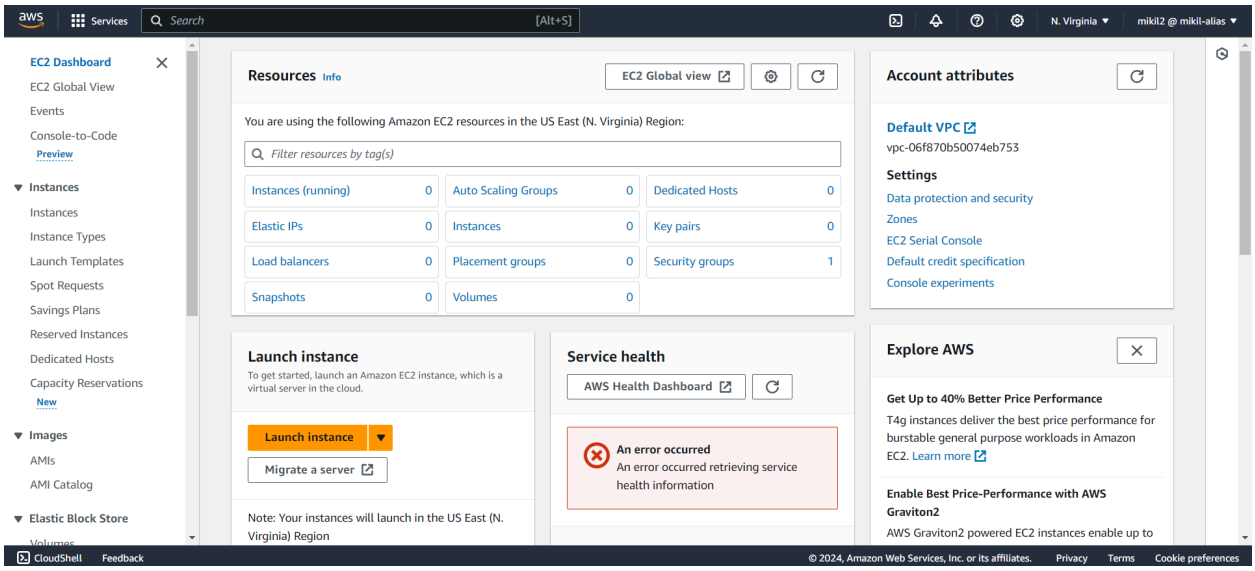
Type in *ec2fullaccess* in the search box and click the check box for it; click on Next

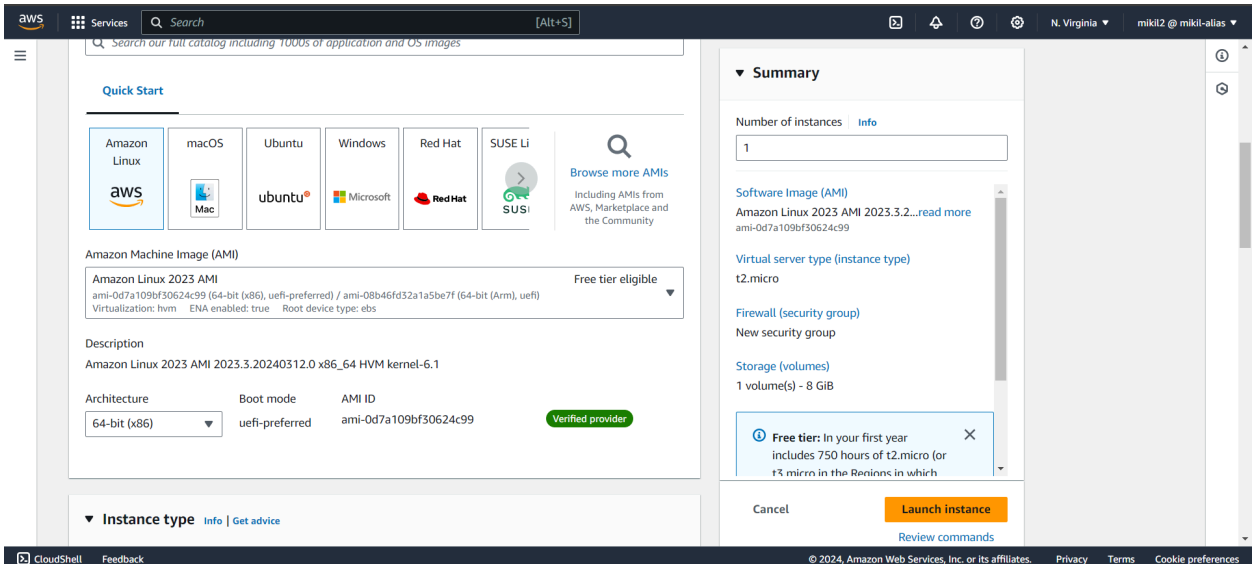


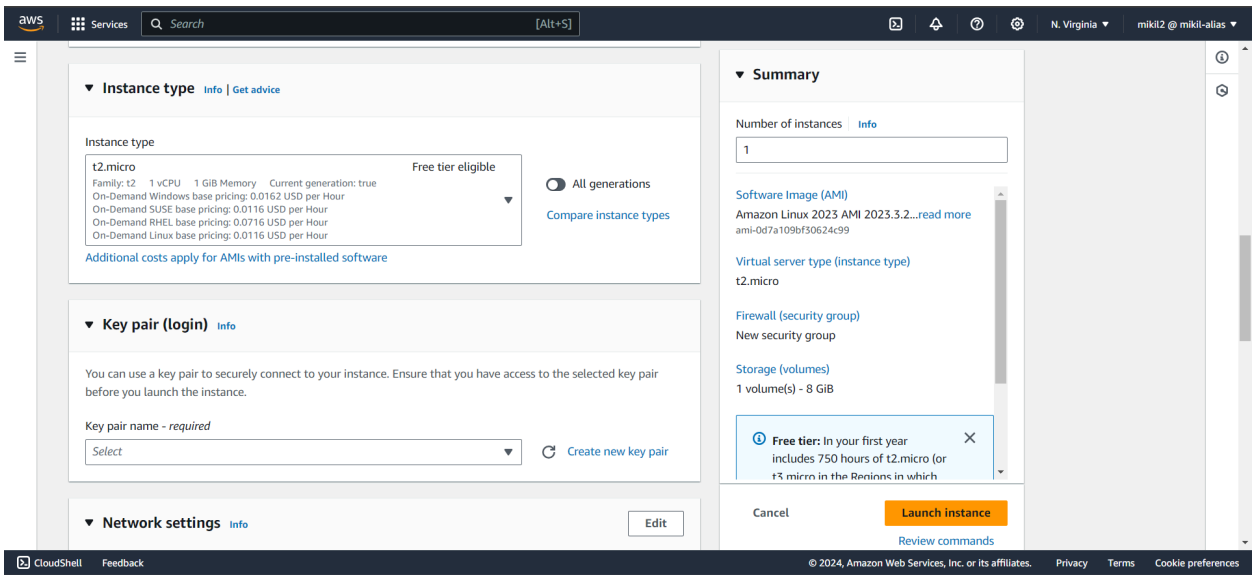
Logging in as one of the 2 new Users and Checking their permissions

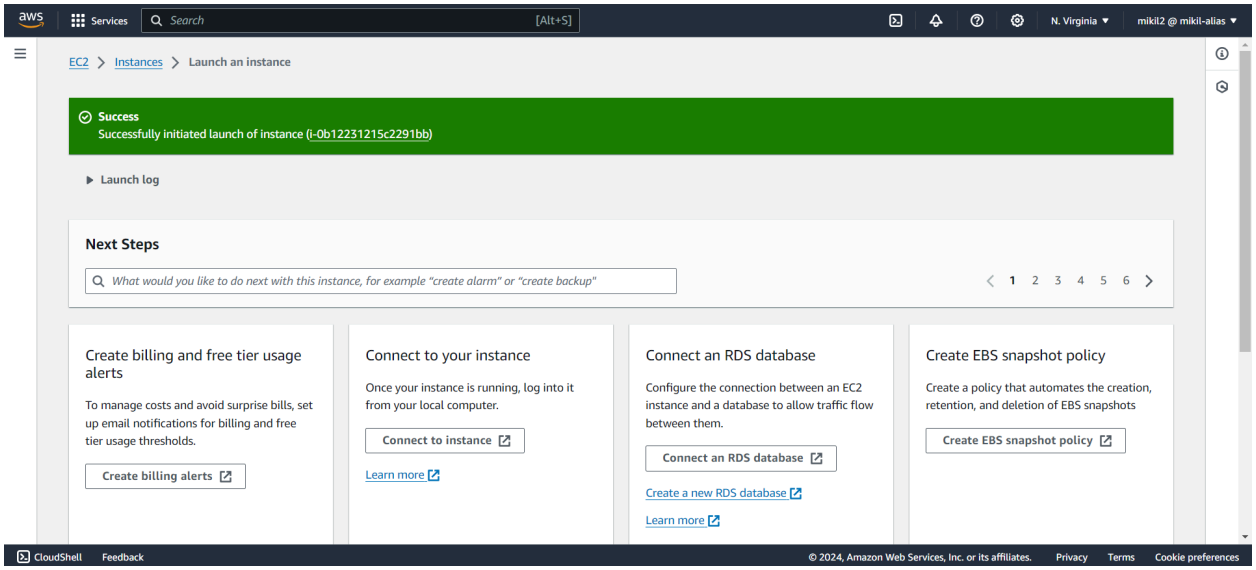


Try launching an EC2 instance via the new user

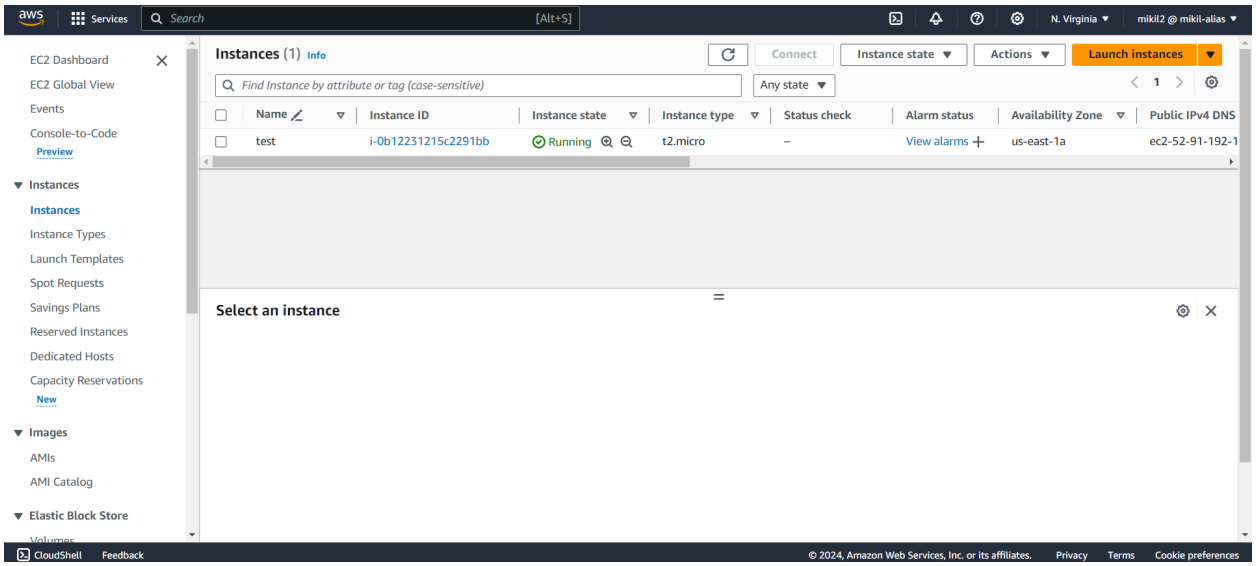



aws
Services
Search
[Alt+S]
N. Virginia
mikil2 @ mikil-alias
Search our full catalog including 1000s of application and OS images
Quick Start
Amazon Linux
macOS
Ubuntu
Windows
Red Hat
SUSE Li
Browse more AMIs
Including AMIs from AWS, Marketplace and the Community
Amazon Machine Image (AMI)
Amazon Linux 2023 AMI
Free tier eligible
ami-0d7a109bf30624c99 (64-bit (x86), uefi-preferred) / ami-08b46fd32a1a5be7f (64-bit (Arm), uefi)
Virtualization: hvm ENA enabled: true Root device type: ebs
Description
Amazon Linux 2023 AMI 2023.3.20240312.0 x86_64 HVM kernel-6.1
Architecture
Boot mode
AMI ID
64-bit (x86)
uefi-preferred
ami-0d7a109bf30624c99
Verified provider
Instance type Info | Get advice
Summary
Number of instances Info
1
Software Image (AMI)
Amazon Linux 2023 AMI 2023.3.2...read more
ami-0d7a109bf30624c99
Virtual server type (instance type)
t2.micro
Firewall (security group)
New security group
Storage (volumes)
1 volume(s) - 8 GiB
Free tier: In your first year includes 750 hours of t2.micro (or
Cancel
Launch instance
Review commands
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


aws
Services
Search
[Alt+S]
N. Virginia
mikil2 @ mikil-alias
Instance type Info | Get advice
Instance type
t2.micro
Free tier eligible
Family: t2 1 vCPU 1 GiB Memory Current generation: true
On-Demand Windows base pricing: 0.0162 USD per Hour
On-Demand SUSE base pricing: 0.0116 USD per Hour
On-Demand RHEL base pricing: 0.0716 USD per Hour
On-Demand Linux base pricing: 0.0116 USD per Hour
All generations
Compare instance types
Additional costs apply for AMIs with pre-installed software
Key pair (login) Info
You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.
Key pair name - required
Select
Create new key pair
Network settings Info
Edit
Summary
Number of instances Info
1
Software Image (AMI)
Amazon Linux 2023 AMI 2023.3.2...read more
ami-0d7a109bf30624c99
Virtual server type (instance type)
t2.micro
Firewall (security group)
New security group
Storage (volumes)
1 volume(s) - 8 GiB
Free tier: In your first year includes 750 hours of t2.micro (or
Cancel
Launch instance
Review commands
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences



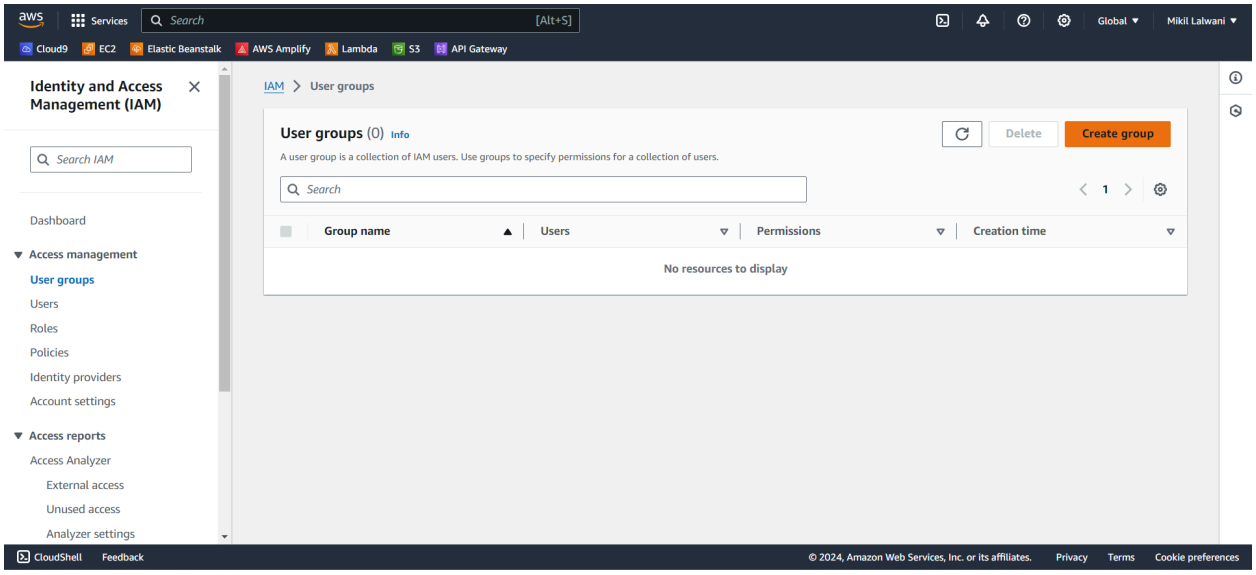
Hence, an instance has been created



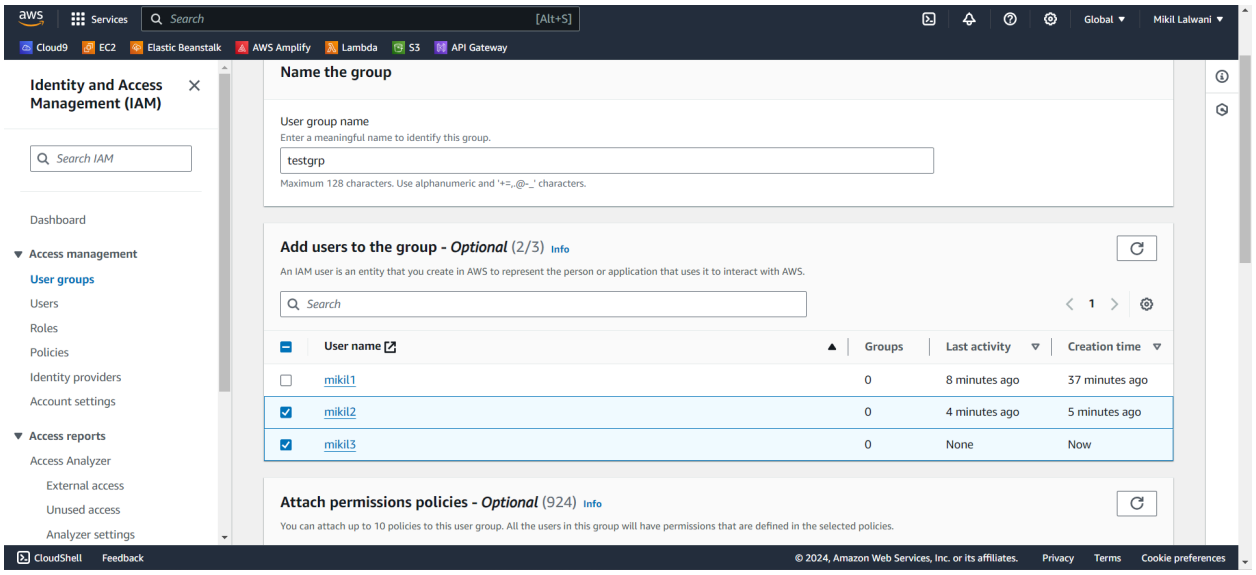
Delete the bucket when done with your work

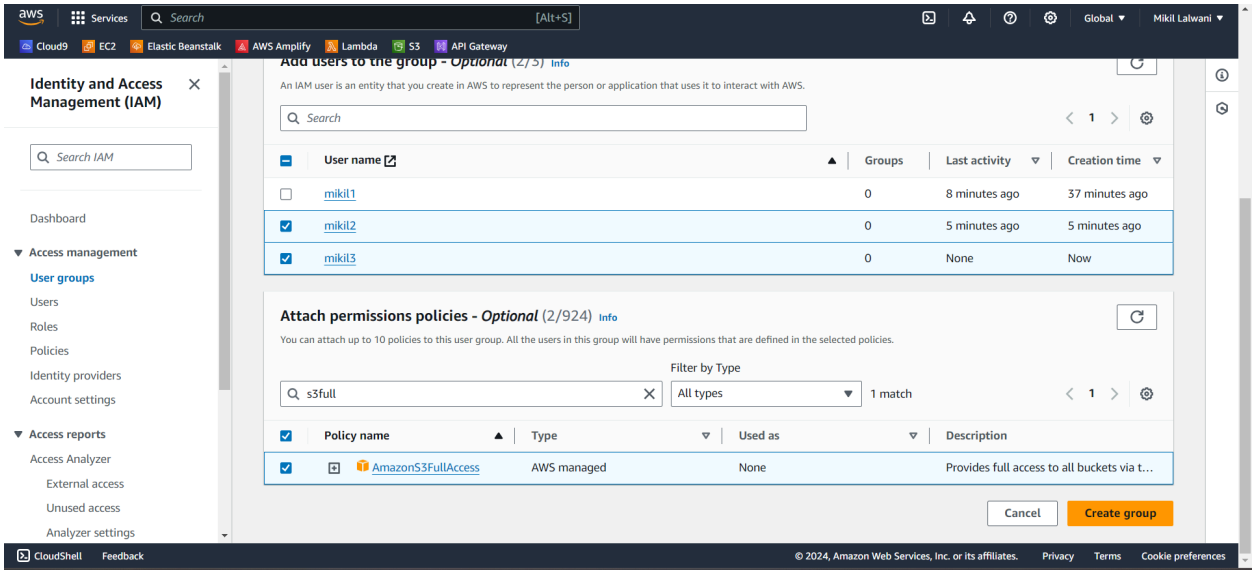
Creating a new Group and Giving it permissions

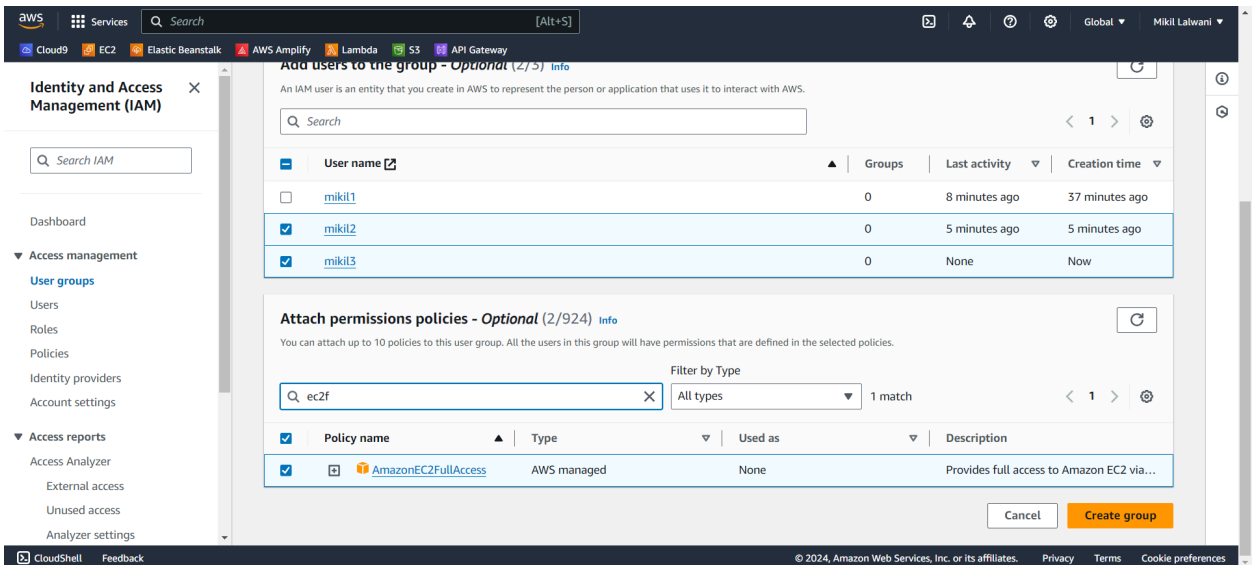
Select the members to be present in the group (max 4 per group)

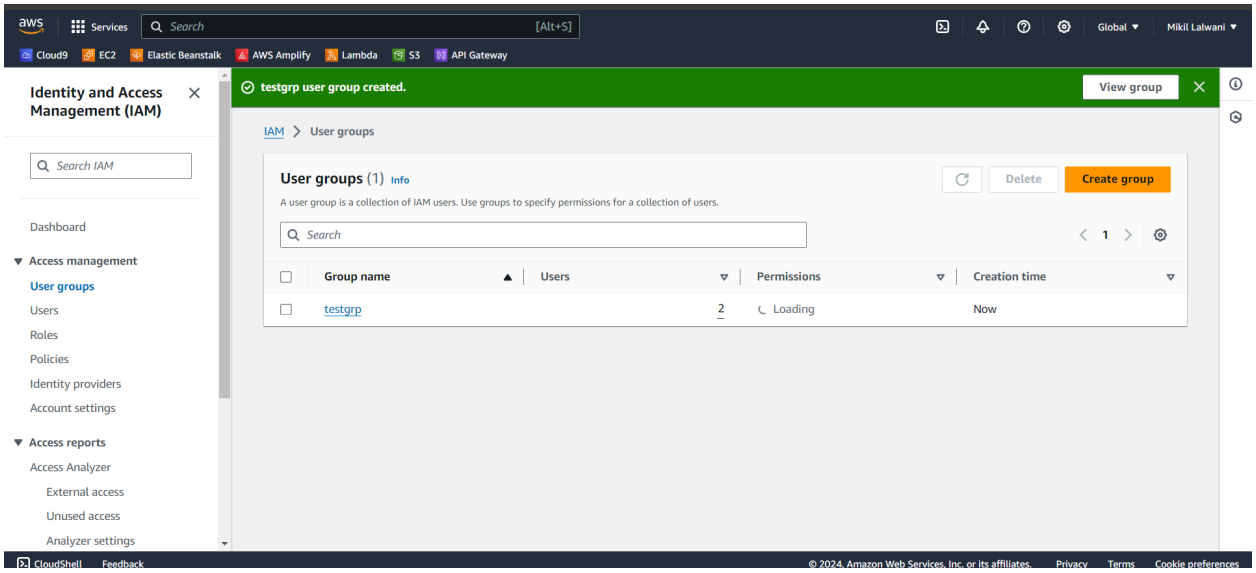


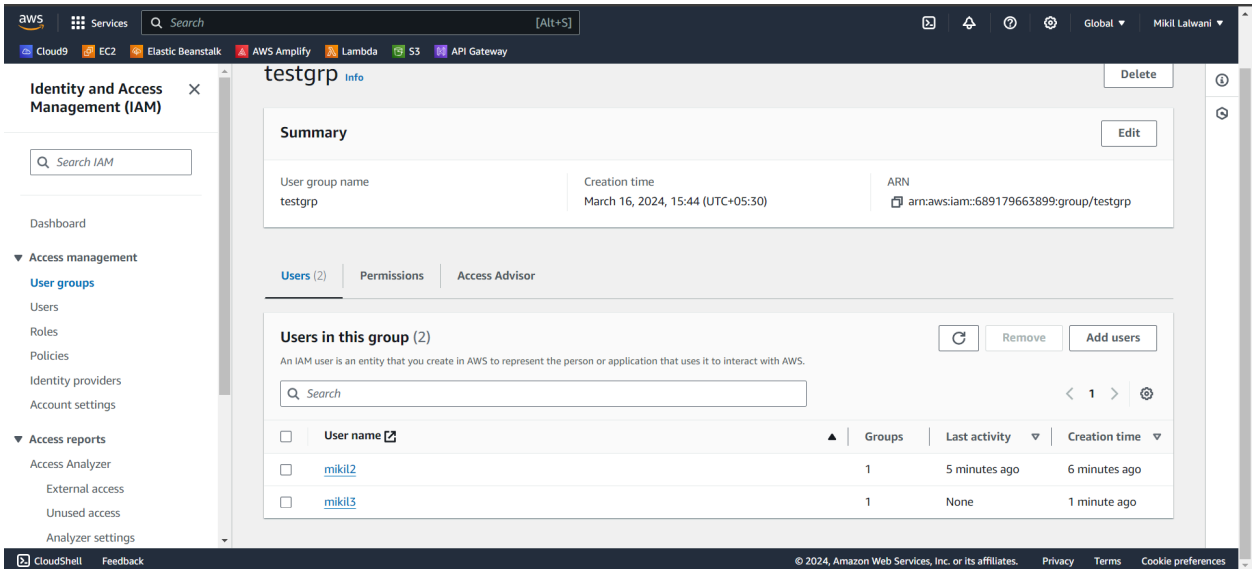
Giving this group *ec2fullaccess* and *s3fullaccess*




Services
Search
[Alt+S]
Global
Mikil Lalwani
Cloud9
EC2
Elastic Beanstalk
AWS Amplify
Lambda
S3
API Gateway
Identity and Access Management (IAM)
Search IAM
Dashboard
Access management
User groups
Users
Roles
Policies
Identity providers
Account settings
Access reports
Access Analyzer
External access
Unused access
Analyzer settings
Add users to the group - Optional (2/5) Info
An IAM user is an entity that you create in AWS to represent the person or application that uses it to interact with AWS.
Search
1
User name
Groups
Last activity
Creation time
mikil1
0
8 minutes ago
37 minutes ago
mikil2
0
5 minutes ago
5 minutes ago
mikil3
0
None
Now
Attach permissions policies - Optional (2/924) Info
You can attach up to 10 policies to this user group. All the users in this group will have permissions that are defined in the selected policies.
Filter by Type
s3full
All types
1 match
Policy name
Type
Used as
Description
AmazonS3FullAccess
AWS managed
None
Provides full access to all buckets via t...
Cancel
Create group
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

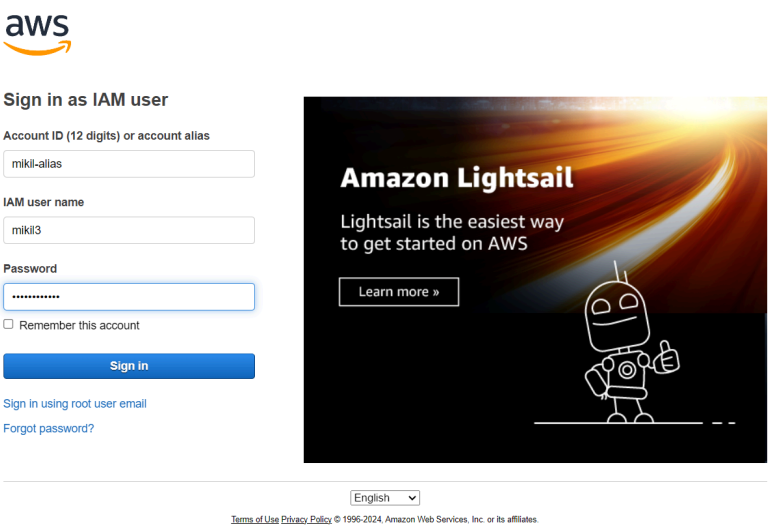

Services
Search
[Alt+S]
Global
Mikil Lalwani
Cloud9
EC2
Elastic Beanstalk
AWS Amplify
Lambda
S3
API Gateway
Identity and Access Management (IAM)
Search IAM
Dashboard
Access management
User groups
Users
Roles
Policies
Identity providers
Account settings
Access reports
Access Analyzer
External access
Unused access
Analyzer settings
Add users to the group - Optional (2/5) Info
An IAM user is an entity that you create in AWS to represent the person or application that uses it to interact with AWS.
Search
1
User name
Groups
Last activity
Creation time
mikil1
0
8 minutes ago
37 minutes ago
mikil2
0
5 minutes ago
5 minutes ago
mikil3
0
None
Now
Attach permissions policies - Optional (2/924) Info
You can attach up to 10 policies to this user group. All the users in this group will have permissions that are defined in the selected policies.
Filter by Type
ec2f
All types
1 match
Policy name
Type
Used as
Description
AmazonEC2FullAccess
AWS managed
None
Provides full access to Amazon EC2 via...
Cancel
Create group
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

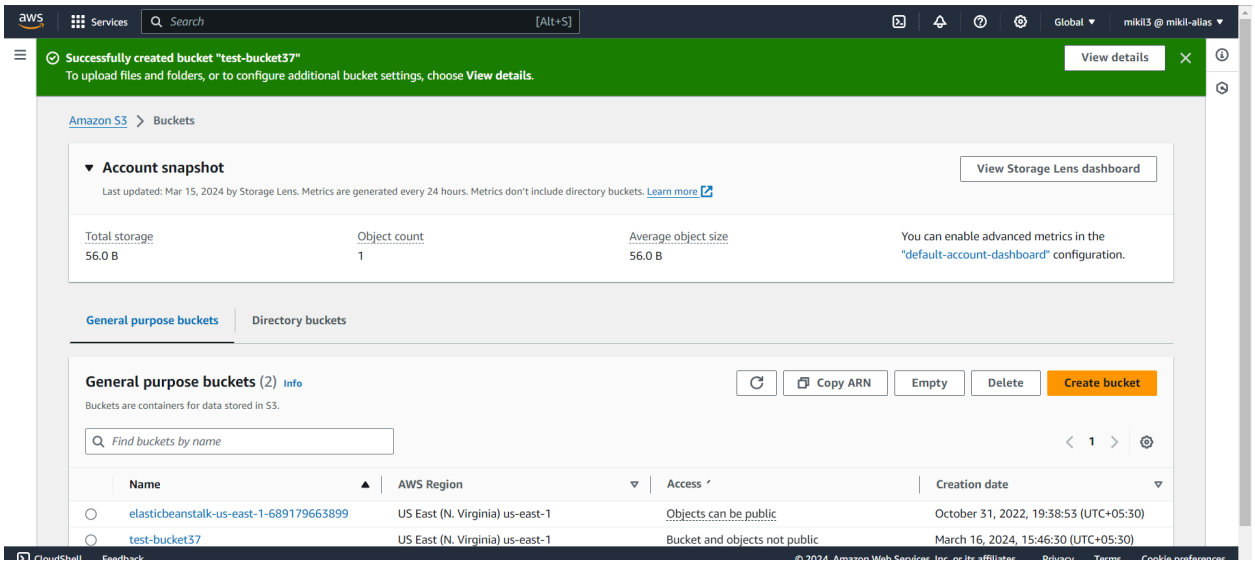




Logging in as a member of the Group & Checking their permissions

Now, login as one of the users from the group and try creating a S3 bucket

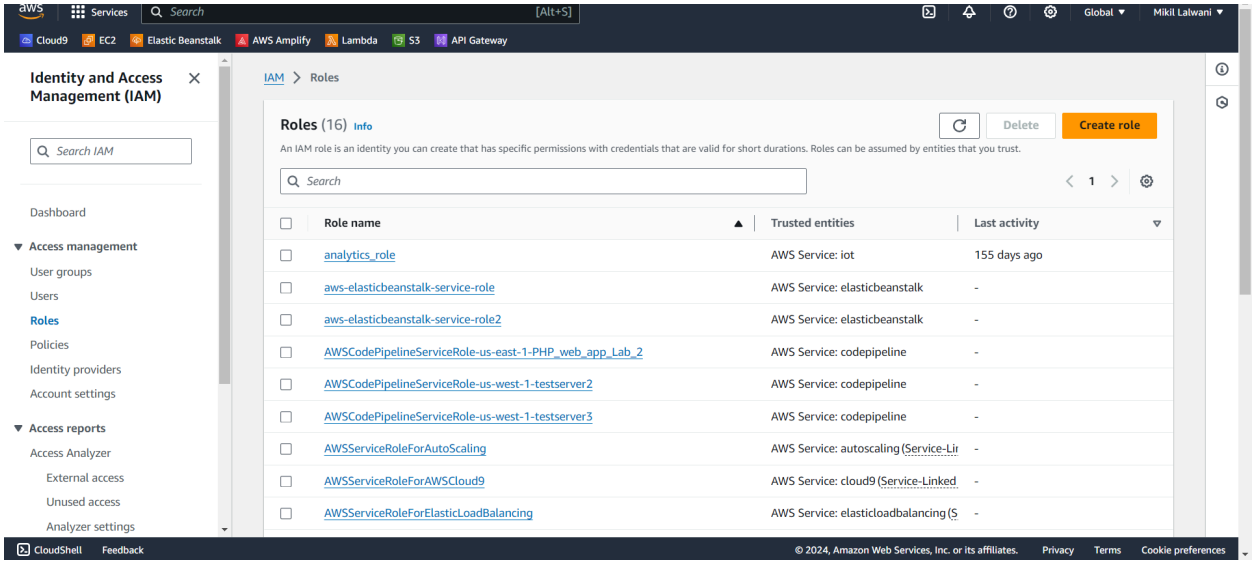




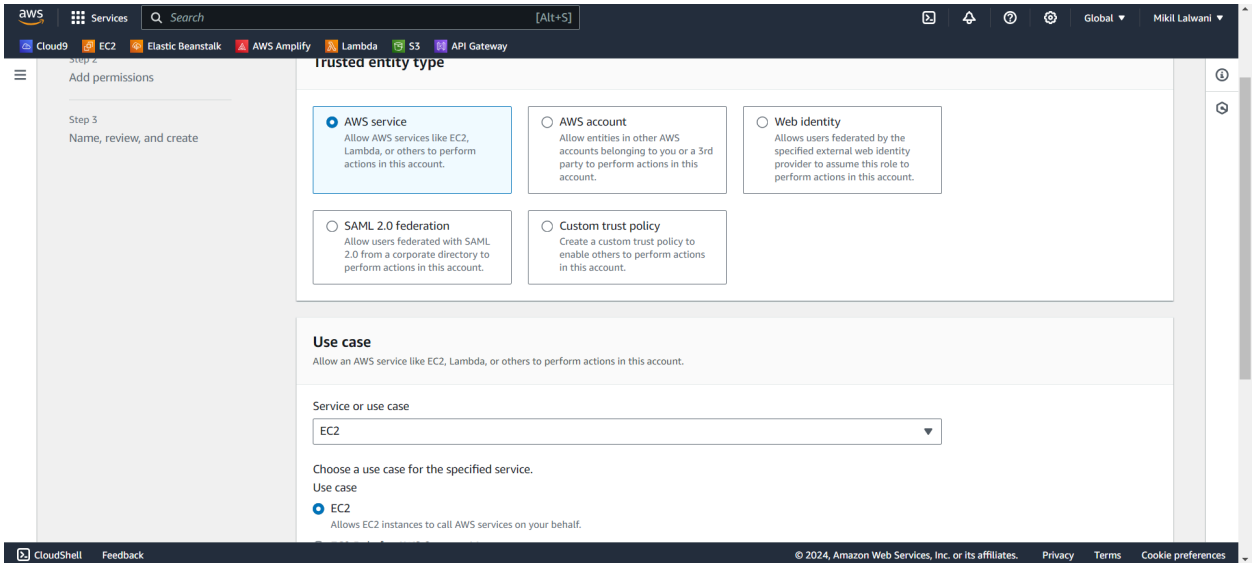
Delete the bucket when done with your work

Creating a new Role

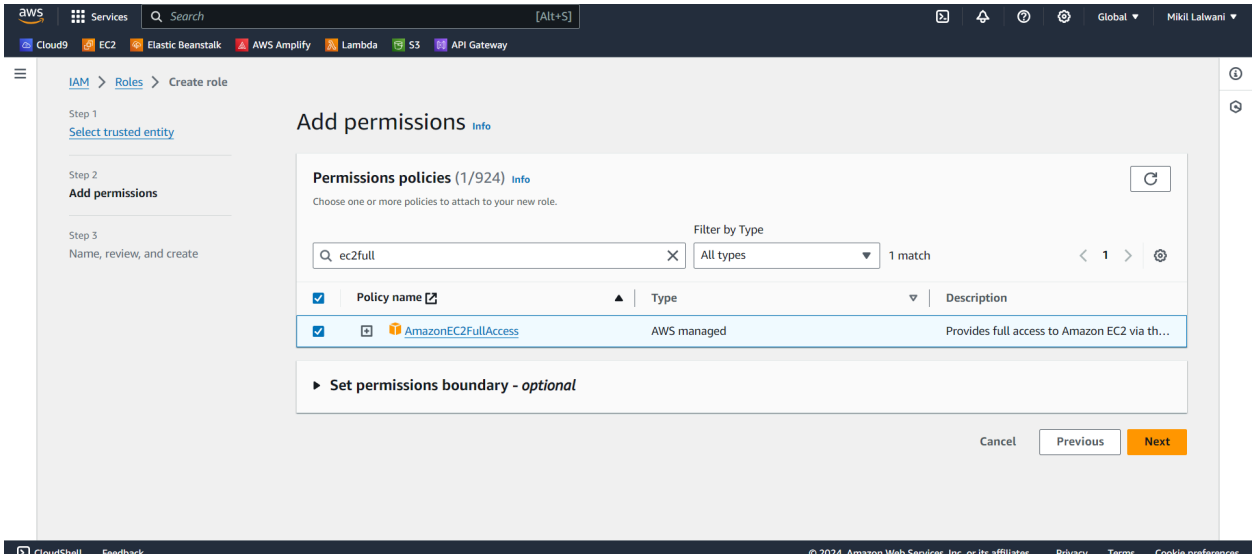
Go in the root user window and click on “create role” button in the “Roles” section of IAM Dashboard



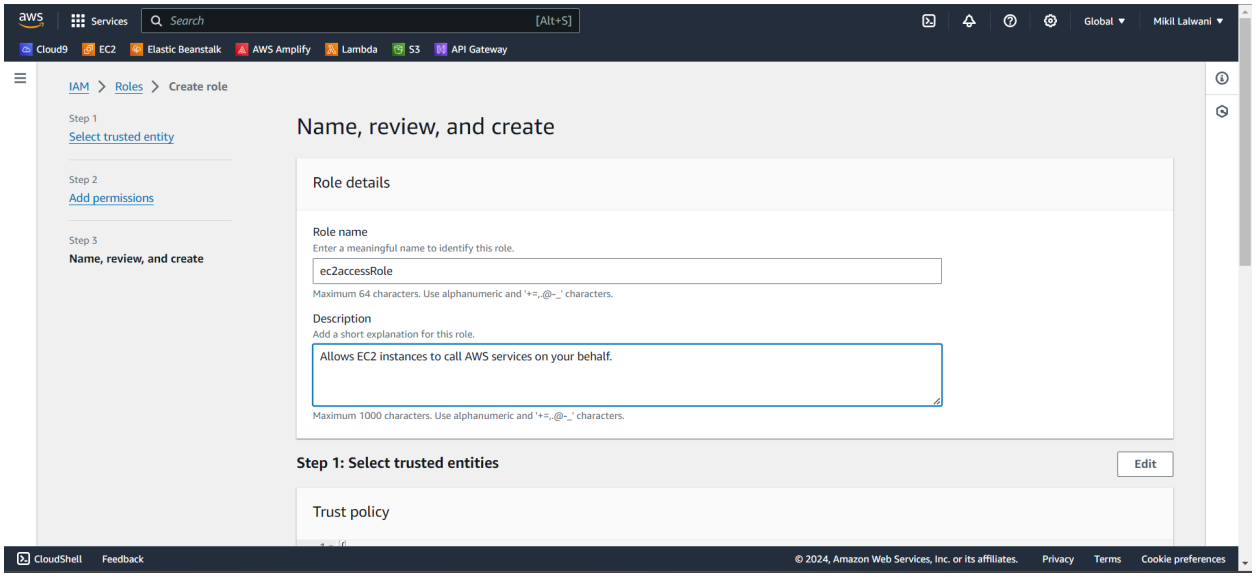
Let it be the default options (you can choose any use case you like)
Click in Next button



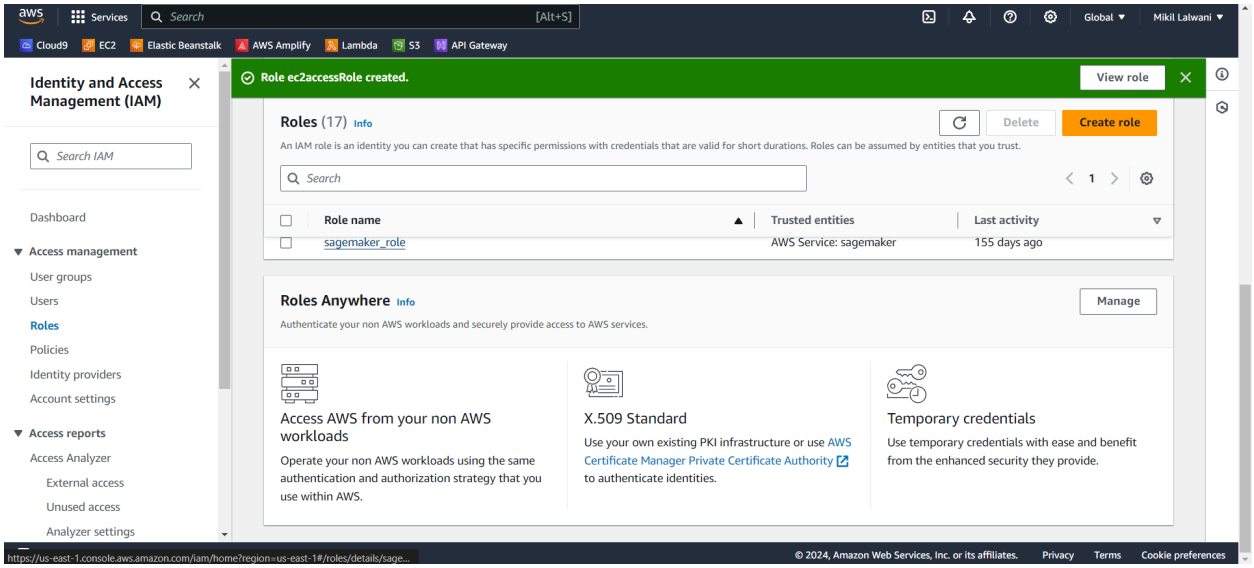
Give the permission suitable to the use case chosen



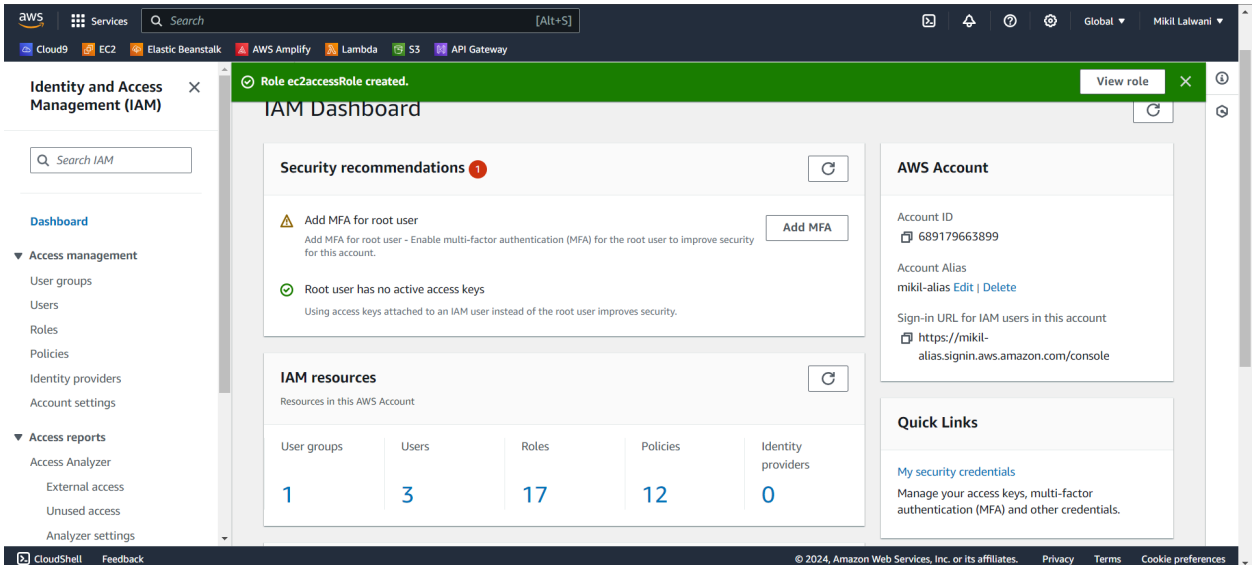
Give a suitable Role name and description; rest would remain as default. click on Create Role button



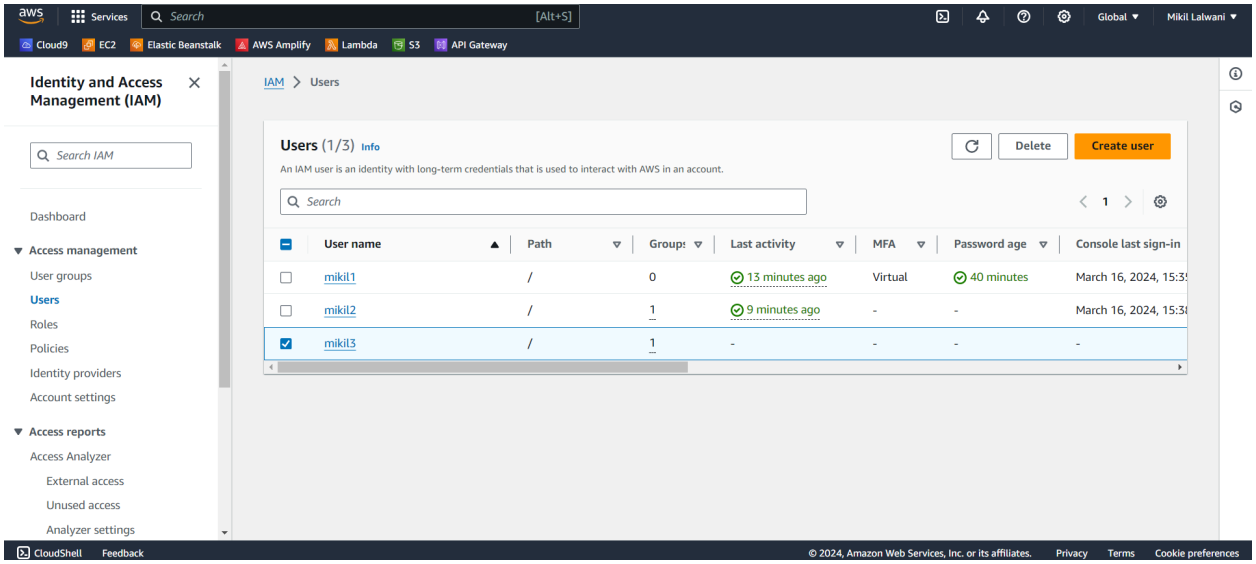
The role has been successfully created

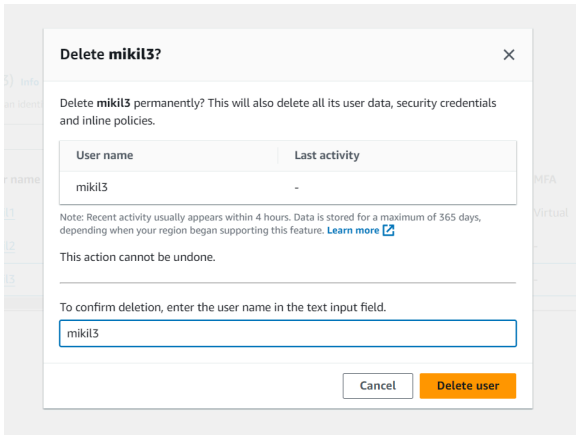


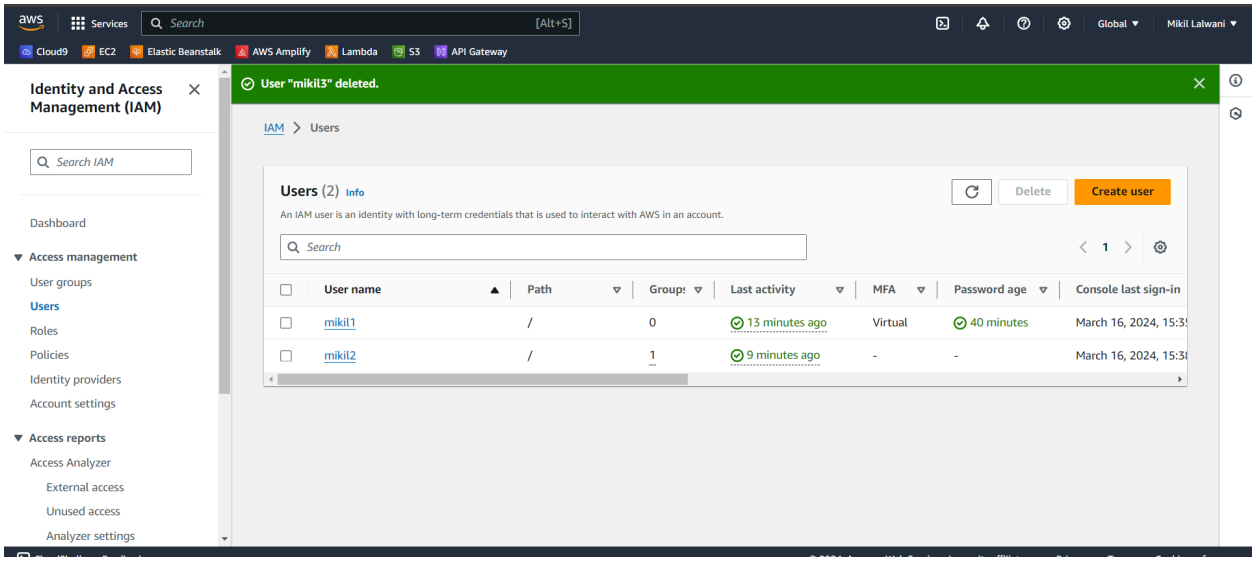
Just to check the overall users, groups and roles, you can check out the IAM Dashboard



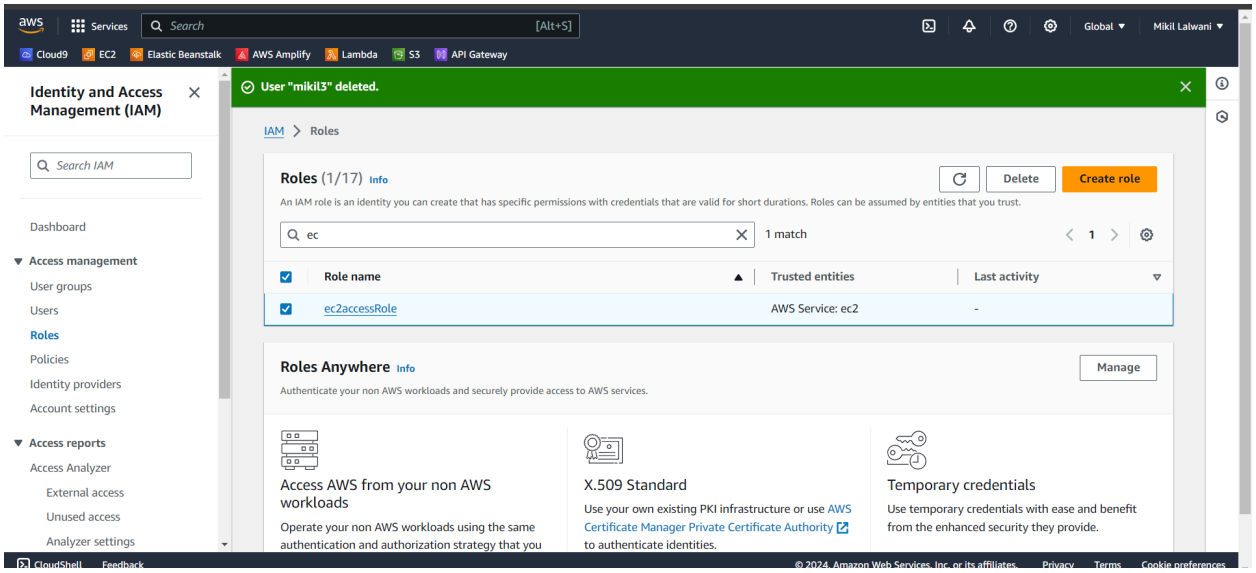
Deleting a User

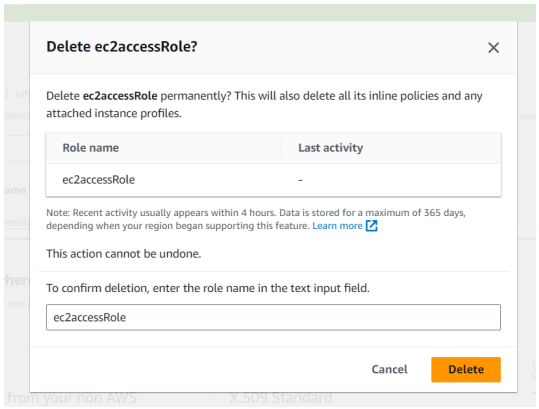


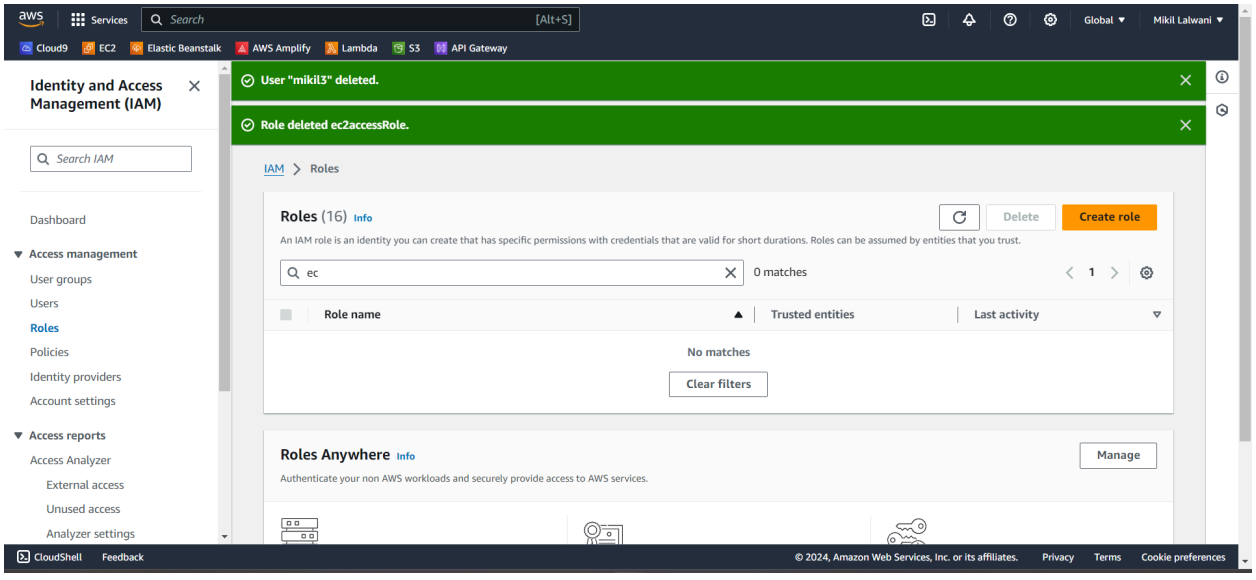




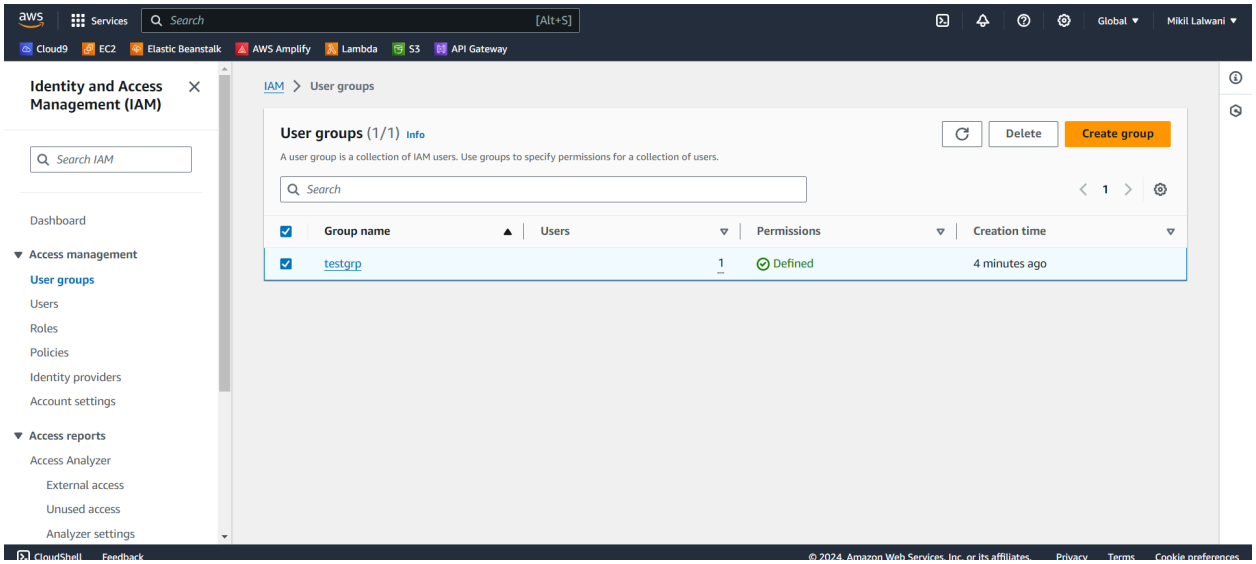
Deleting a Role

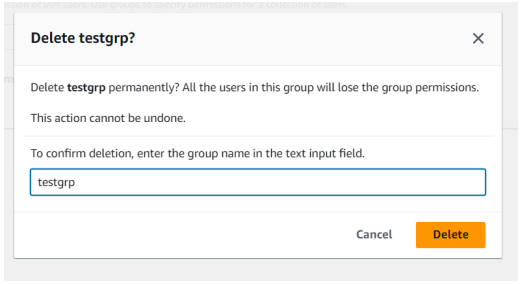




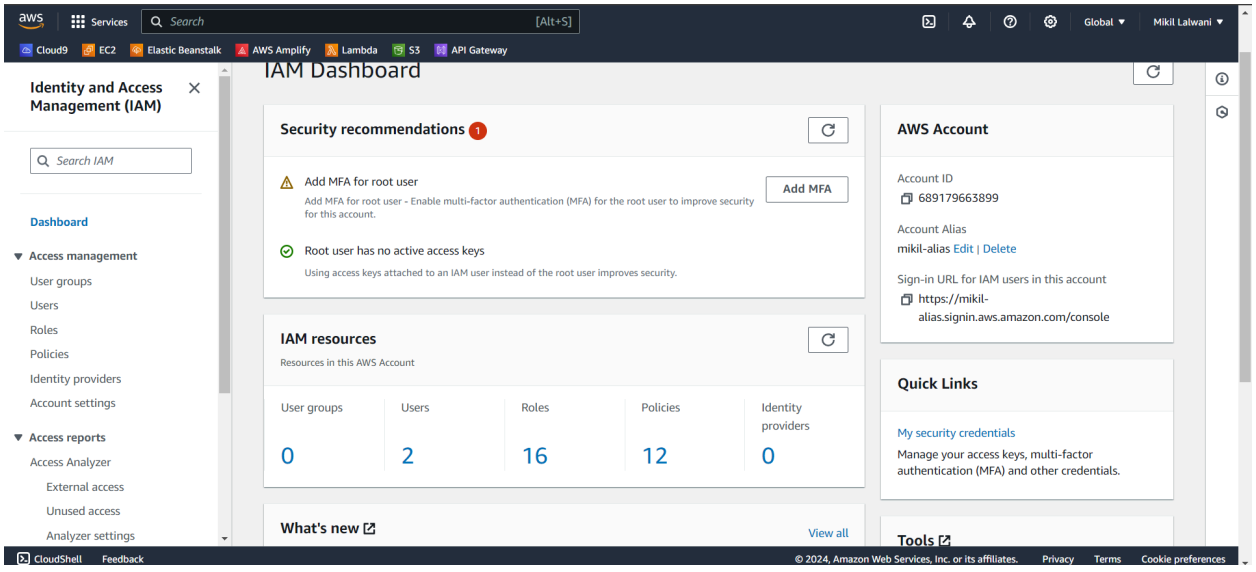


Deleting a Group

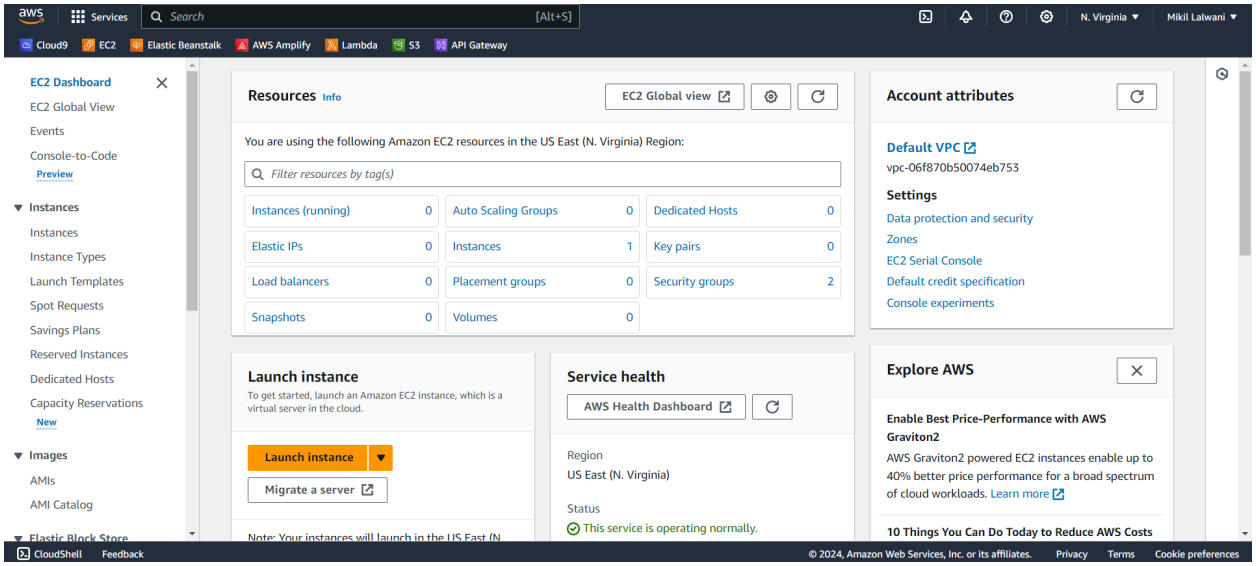




Check the IAM dashboard to see the results after deletion activities



Check the ec2 dashboard in case there are any running instances



Conclusion -

Thus we have successfully studied and implemented Identity and Access Management (IAM) practices on AWS.