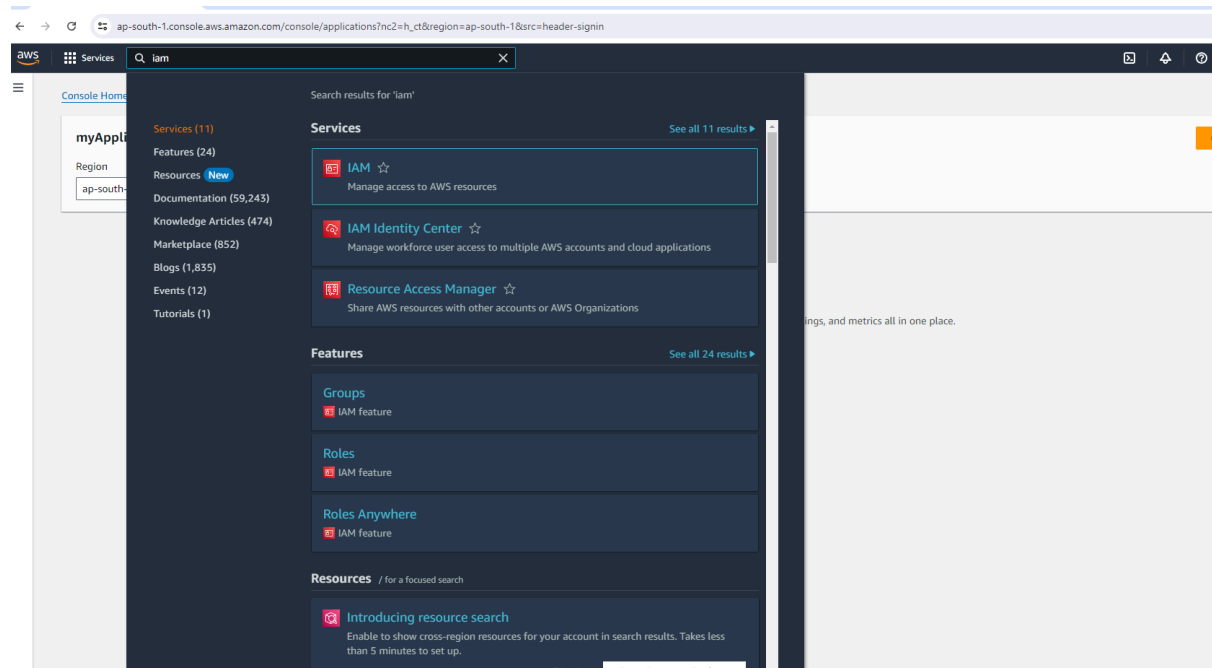


Name: Garima Joshi
Cloud Computing: Identity Access Management

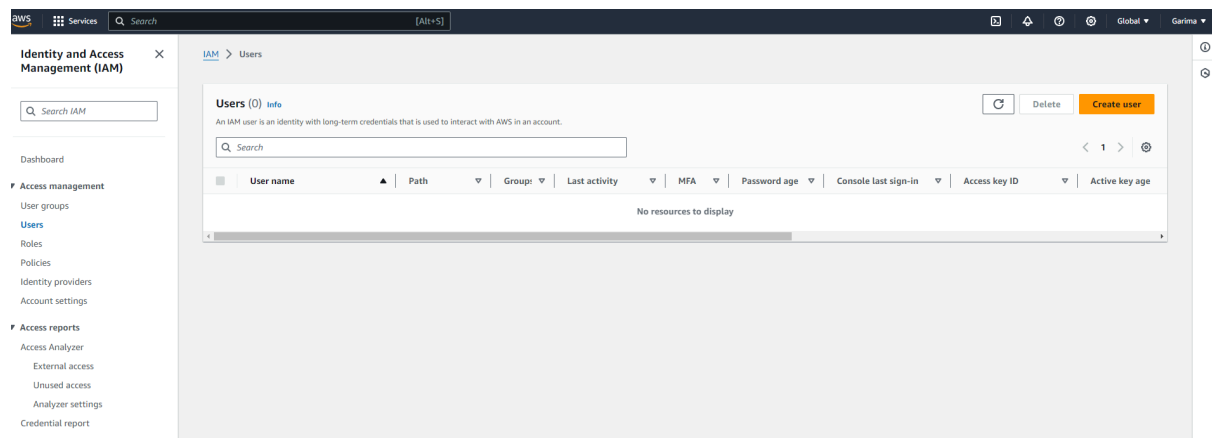
Q.Implementing policies for IAM users to access

1. S3 service
2. EC2 service

Ans:



Select IAM from services.



Select the “Create User” option

This screenshot shows the 'Specify user details' step of the AWS IAM 'Create user' process. The 'User name' field is populated with 'admin'. Below the field, there is a checkbox for 'Provide user access to the AWS Management Console - optional' and a note about generating credentials for programmatic access. The 'Next' button is highlighted in orange.

Specify user details

User details

User name
admin

The user name can have up to 64 characters. Valid characters: A-Z, a-z, 0-9, and ! ~ , @ _ - (hyphen)

☐ Provide user access to the AWS Management Console - optional
If you're providing console access to a person, it's a best practice to manage their access in IAM Identity Center.

[Learn more](#)

Cancel **Next**

Given name to the user.

This screenshot shows the 'Set permissions' step. Three options are available: 'Add user to group' (selected), 'Copy permissions', and 'Attach policies directly'. A 'Create group' button is visible next to the 'Get started with groups' section. The 'Next' button is highlighted in orange.

Set permissions

Add user to an existing group or create a new one. Using groups is a best-practice way to manage user's permissions by job functions. [Learn more](#)

Permissions options

☒ **Add user to group**
Add user to an existing group, or create a new group. We recommend using groups to manage user permissions by job function.

☐ **Copy permissions**
Copy all group memberships, attached managed policies, and inline policies from an existing user.

☐ **Attach policies directly**
Attach a managed policy directly to a user. As a best practice, we recommend attaching policies to a group instead. Then, add the user to the appropriate group.

Get started with groups
Create a group and select policies to attach to the group. We recommend using groups to manage user permissions by job function, AWS service access, or custom permissions. [Learn more](#)

Create group

Set permissions boundary - optional

Cancel **Previous** **Next**

Select the first option of adding the user to the group and hit the next button

This screenshot shows the 'Review and create' step. It displays a summary of the user details (name: admin, console password type: None, require password reset: No) and a permissions summary table. There are no resources listed in the permissions summary. The 'Create user' button is highlighted in orange.

Review and create

Review your choices. After you create the user, you can view and download the autogenerated password, if enabled.

User details

User name admin	Console password type None	Require password reset No
--------------------	-------------------------------	------------------------------

Permissions summary

Name	Type	Used as
No resources		

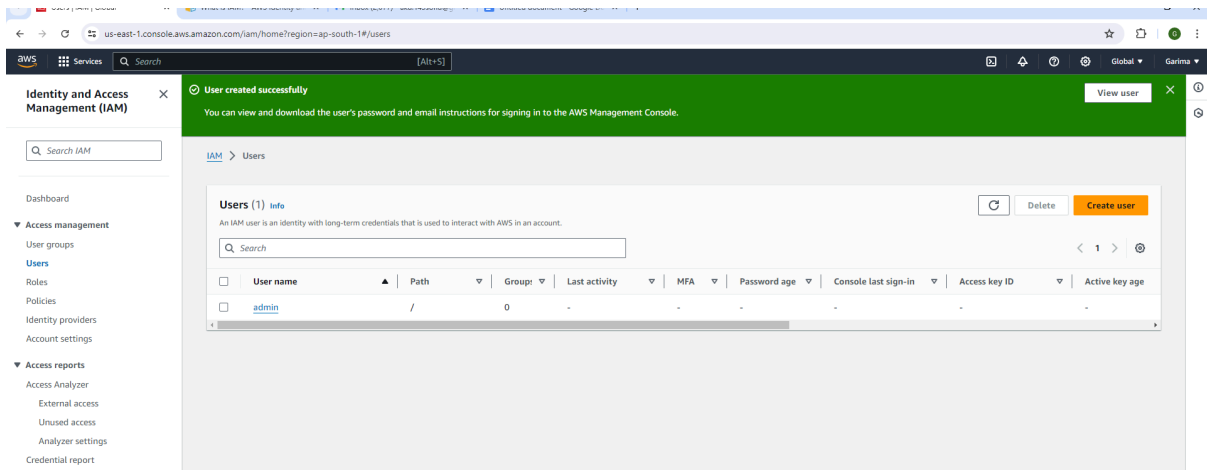
Tags - optional
Tags are key-value pairs you can add to AWS resources to help identify, organize, or search for resources. Choose any tags you want to associate with this user.

No tags associated with the resource.

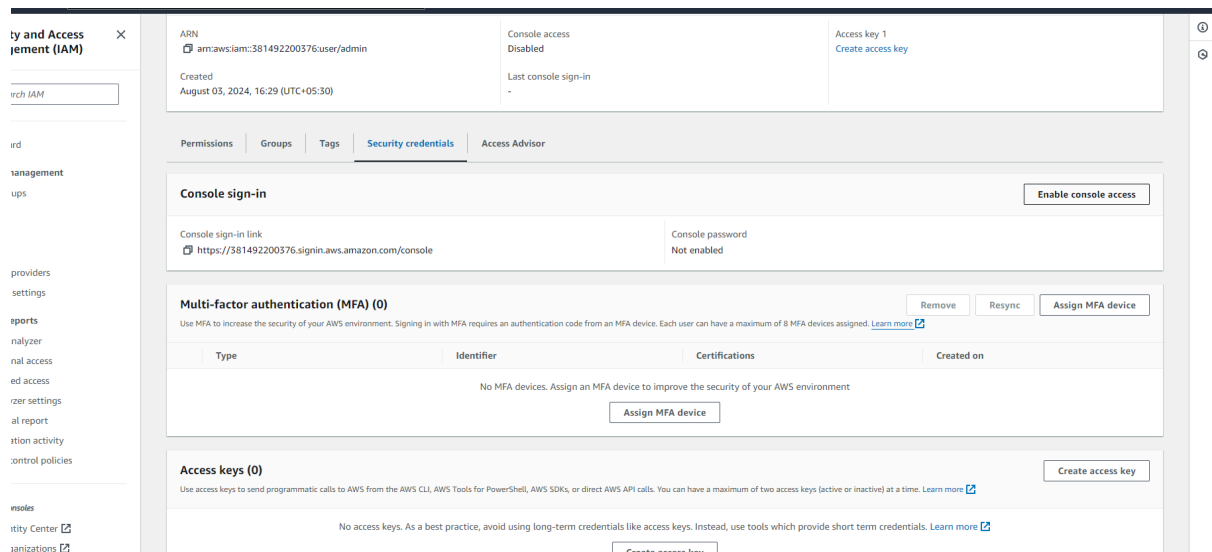
Add new tag
You can add up to 50 more tags.

Cancel **Previous** **Create user**

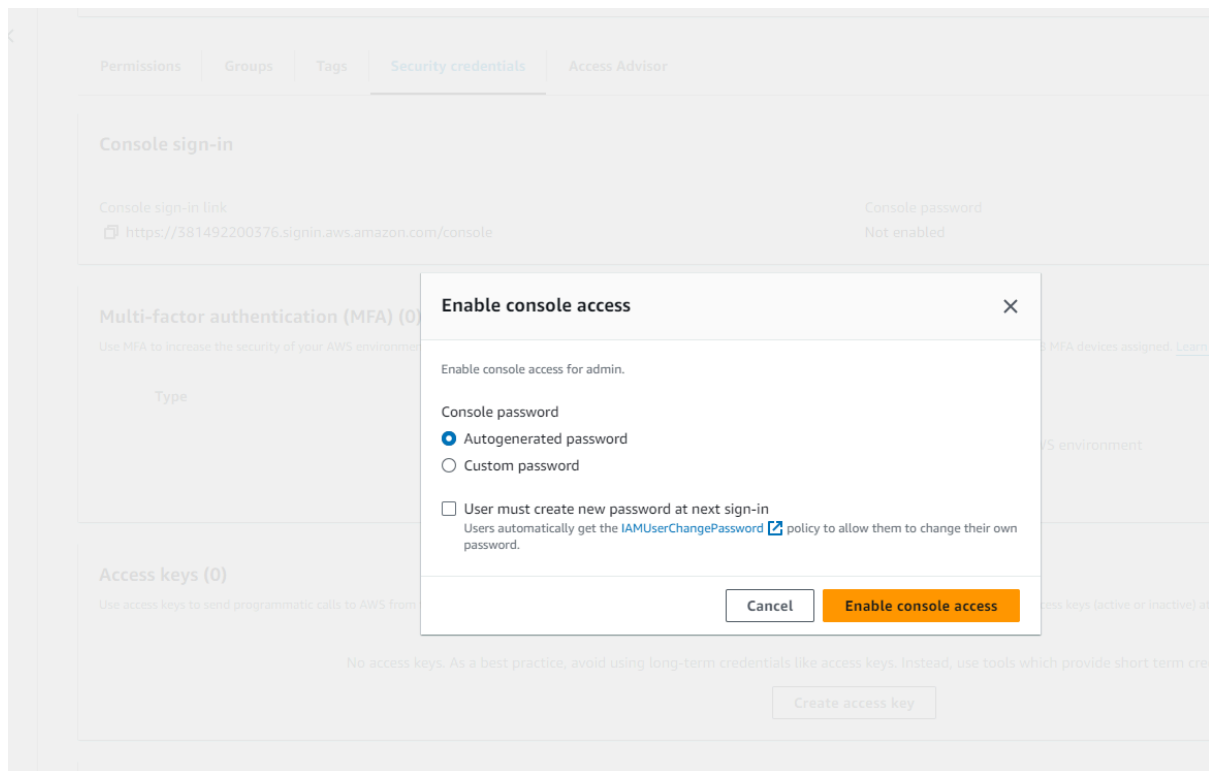
Review the details and hit the "Create User" button



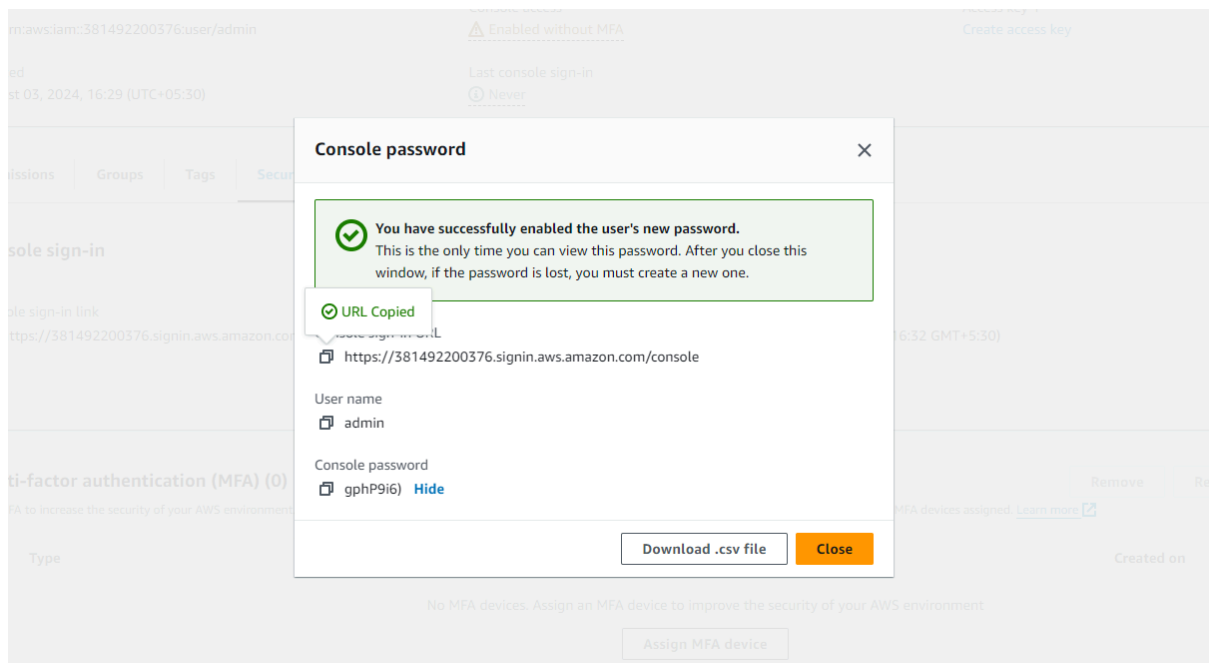
Open the user account “admin”.



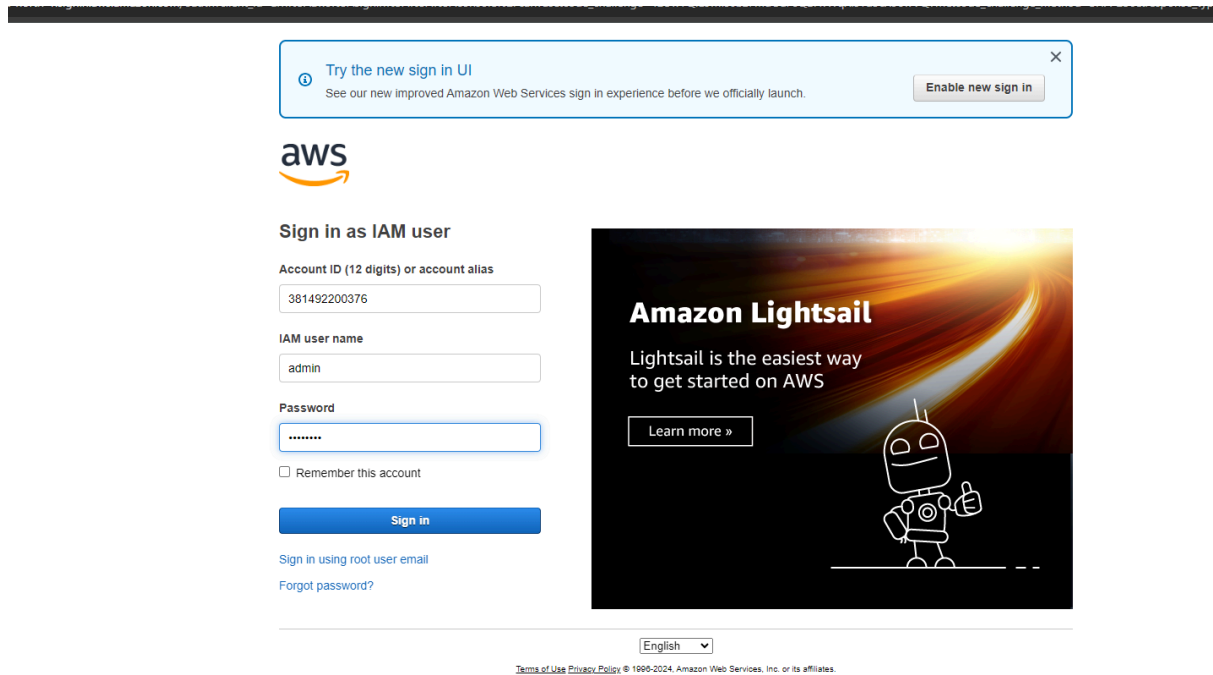
In the “Security credentials” section, in the “Console sign-in” section, click on “Enable console access”



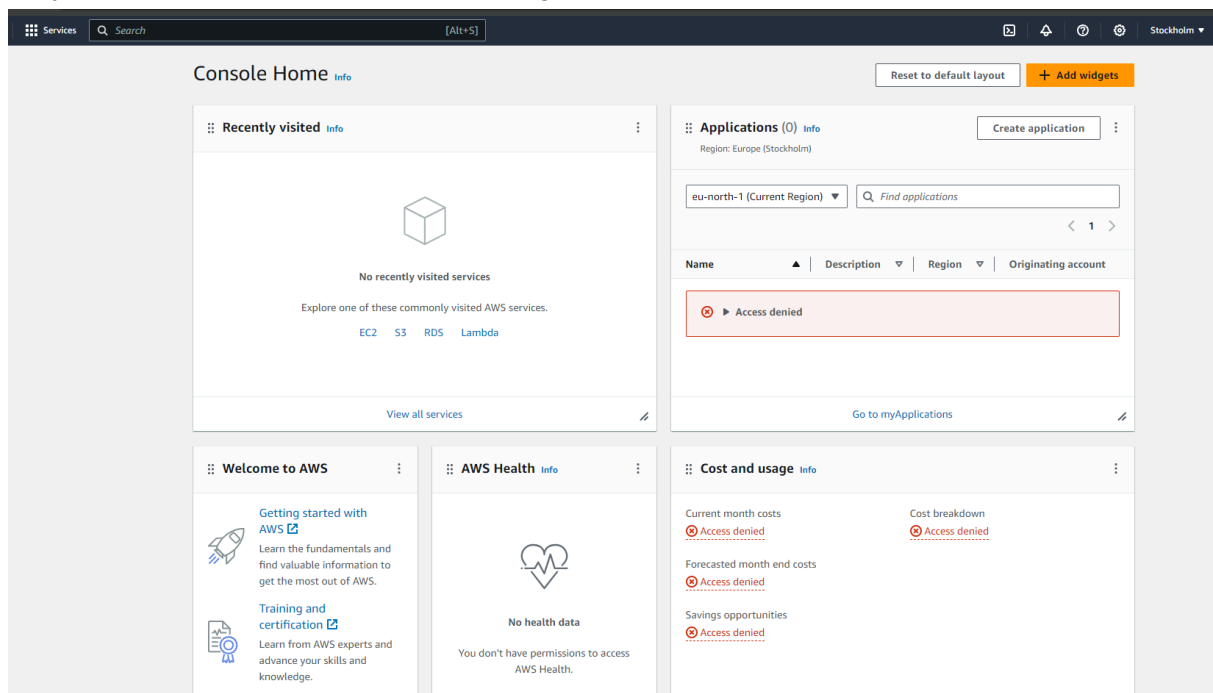
Here, select “Autogenerated password” and then click on the “Enable console access” button.



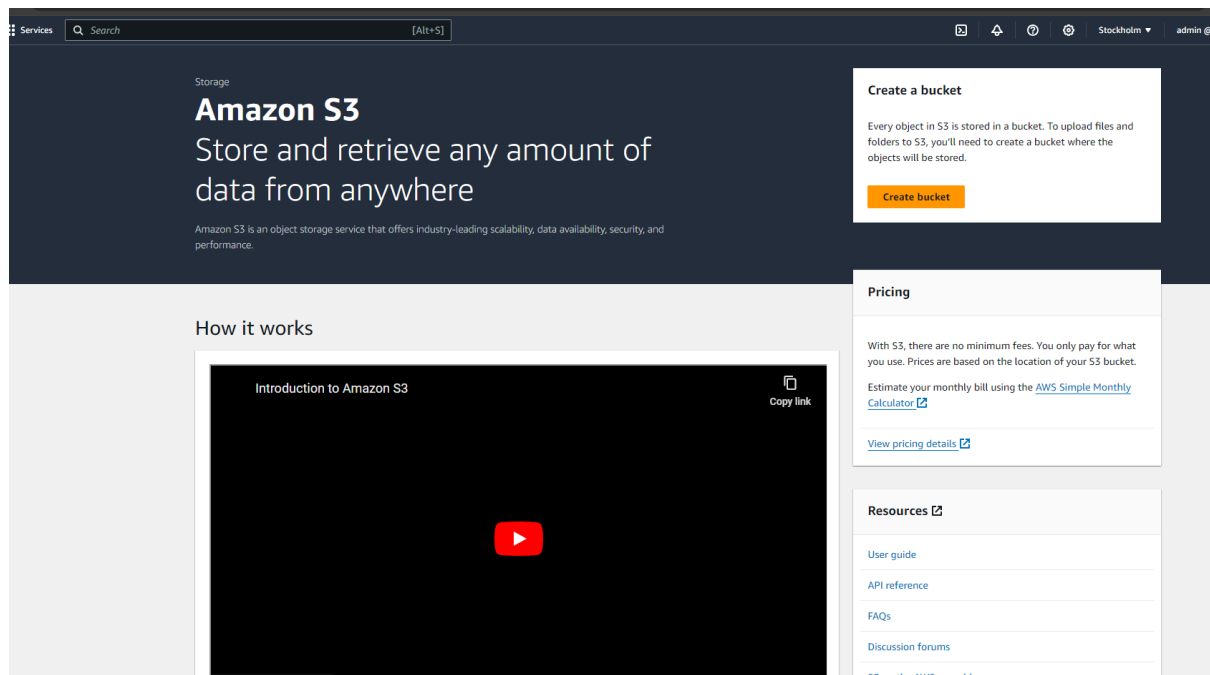
After creating the password, we get the password and the link for the admin user’s account. Download the .csv file. Copy the link and open the new account. Not to close this dialog box.



This is the page to sign in to the admin user's AWS account. From the before dialog box, copy the password and paste it here to sign in.



After signing in, this screen shows that the user cannot access the services, so the costs and applications section shows “Access denied”.
Eg, we try to create a bucket using the S3 service.



Click on the “Create Bucket” option.

[Amazon S3](#) > [Buckets](#) > [Create bucket](#)

Create bucket [Info](#)

Buckets are containers for data stored in S3.

General configuration

AWS Region
Europe (Stockholm) eu-north-1

Bucket type [Info](#)

☒ **General purpose**
Recommended for most use cases and access patterns. General purpose buckets are the original S3 bucket type. They allow a mix of storage classes that redundantly store objects across multiple Availability Zones.

☐ **Directory - New**
Recommended for low-latency use cases. These buckets use only the S3 Express One Zone storage class, which provides faster processing of data within a single Availability Zone.

Bucket name [Info](#)

Bucket name must be unique within the global namespace and follow the bucket naming rules. [See rules for bucket naming](#)

Copy settings from existing bucket - *optional*
Only the bucket settings in the following configuration are copied.

Format: s3://bucket/prefix

Object Ownership [Info](#)

Control ownership of objects written to this bucket from other AWS accounts and the use of access control lists (ACLs). Object ownership determines who can specify access to objects.

☒ **ACLs disabled (recommended)**
All objects in this bucket are owned by this account. Access to this bucket and its objects is specified using only policies.

☐ **ACLs enabled**
Objects in this bucket can be owned by other AWS accounts. Access to this bucket and its objects can be specified using ACLs.

Select appropriate options for creating a bucket.

Default encryption Info

Server-side encryption is automatically applied to new objects stored in this bucket.

Encryption type Info

☒ Server-side encryption with Amazon S3 managed keys (SSE-S3)
☐ Server-side encryption with AWS Key Management Service keys (SSE-KMS)
☐ Dual-layer server-side encryption with AWS Key Management Service keys (DSSE-KMS)
 Secure your objects with two separate layers of encryption. For details on pricing, see [DSSE-KMS pricing](#) on the [Storage](#) tab of the [Amazon S3 pricing page](#).

Bucket Key

Using an S3 Bucket Key for SSE-KMS reduces encryption costs by lowering calls to AWS KMS. S3 Bucket Keys aren't supported for DSSE-KMS. [Learn more](#)

☐ Disable
☒ Enable

► Advanced settings

After creating the bucket, you can upload files and folders to the bucket, and configure additional bucket settings.

Failed to create bucket

To create a bucket, the `s3:CreateBucket` permission is required.

View your permissions in the [IAM console](#). [Identity and Access Management in Amazon S3](#)

► API response

Cancel

Create bucket

In the end, when we try to create the bucket, we get the message that the bucket cannot be made as the user “admin” doesn’t have access to it.

Identity and Access Management (IAM)

Q Search IAM

Dashboard

Access management

User groups

Users

Roles

Policies

Identity providers

Account settings

Access reports

Access Analyzer

External access

Unused access

Analyzer settings

Credential report

Organization activity

Service control policies

Related consoles

IAM Identity Center

AWS Organizations

Policies (1221) Info

A policy is an object in AWS that defines permissions.

Q Search

Filter by Type

All types

< 1 2 3 4 5 6 7 ... 62 >

Policy name

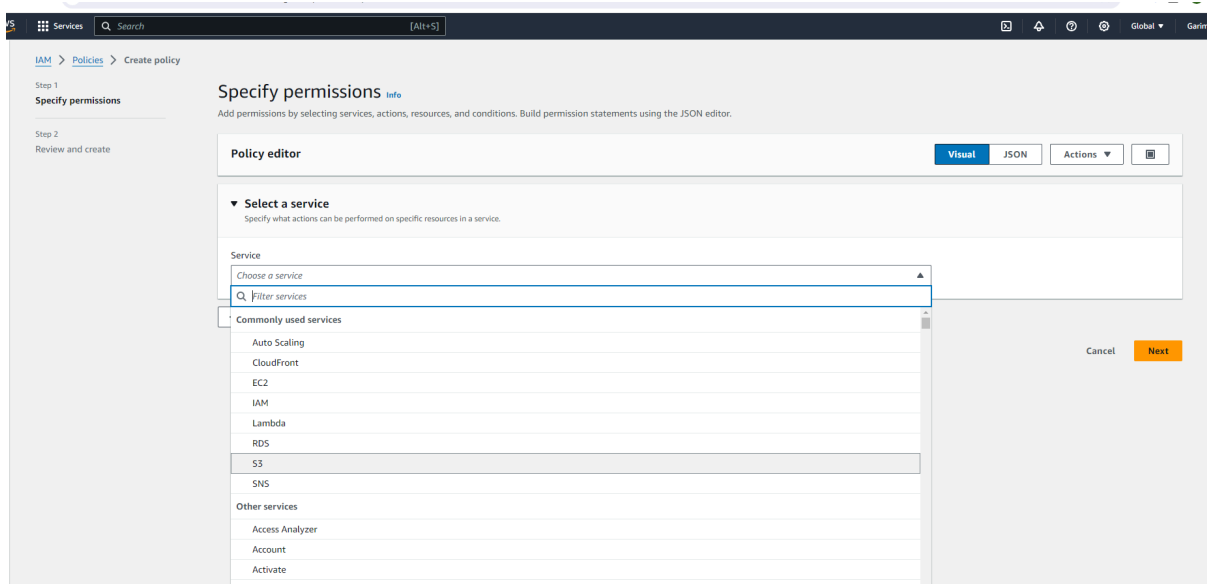
Type

Used as

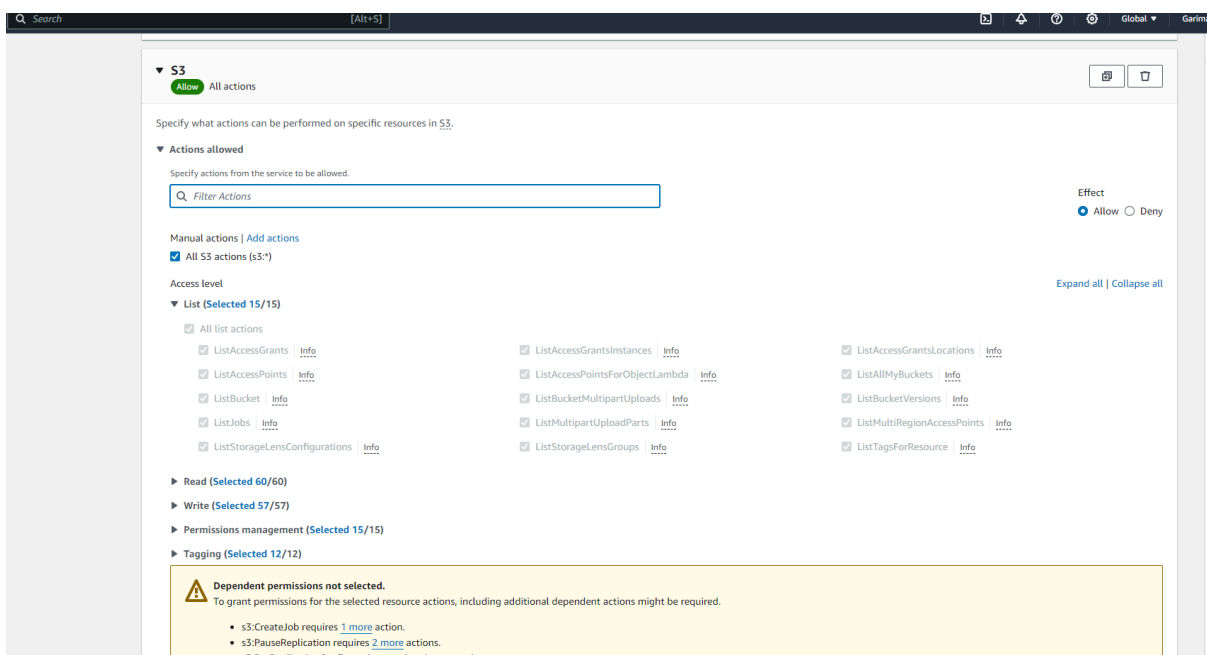
Description

<input type="radio"/>	AdministratorAccess-AWSElasticBeanstalk	AWS managed	None	Grants account administrative permis...
<input type="radio"/>	AlexaForBusinessDeviceSetup	AWS managed	None	Provide device setup access to AlexaFo...
<input type="radio"/>	AlexaForBusinessFullAccess	AWS managed	None	Grants full access to AlexaForBusiness ...
<input type="radio"/>	AlexaForBusinessGatewayExecution	AWS managed	None	Provide gateway execution access to A...
<input type="radio"/>	AlexaForBusinessLifsizeDelegatedAccess...	AWS managed	None	Provide access to Lifesize AVS devices
<input type="radio"/>	AlexaForBusinessNetworkProfileServicePo...	AWS managed	None	-
<input type="radio"/>	AlexaForBusinessPolyDelegatedAccessPolicy	AWS managed	None	Provide access to Poly AVS devices
<input type="radio"/>	AlexaForBusinessReadOnlyAccess	AWS managed	None	Provide read only access to AlexaForB...
<input type="radio"/>	AmazonAPIGatewayAdministrator	AWS managed	None	Provides full access to create/edit/dele...
<input type="radio"/>	AmazonAPIGatewayInvokeFullAccess	AWS managed	None	Provides full access to invoke APIs in A...
<input type="radio"/>	AmazonAPIGatewayPushToCloudWatchLogs	AWS managed	None	Allows API Gateway to push logs to us...
<input type="radio"/>	AmazonAppFlowFullAccess	AWS managed	None	Provides full access to Amazon AppFlo...
<input type="radio"/>	AmazonAppFlowReadOnlyAccess	AWS managed	None	Provides read only access to Amazon A...
<input type="radio"/>	AmazonAppStreamFullAccess	AWS managed	None	Provides full access to Amazon AppStr...
<input type="radio"/>	AmazonAppStreamPCAAccess	AWS managed	None	Amazon AppStream 2.0 access to AWS...
<input type="radio"/>	AmazonAppStreamReadOnlyAccess	AWS managed	None	Provides read only access to Amazon A...
<input type="radio"/>	AmazonAppStreamServiceAccess	AWS managed	None	Default policy for Amazon AppStream ...

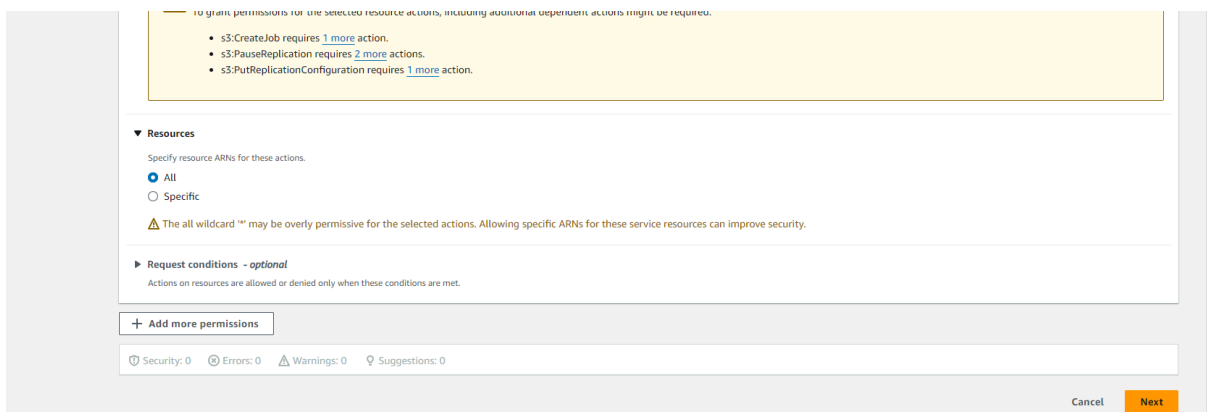
Go back to the root user’s account. On the left side, select “Policies” from the Access Management section. We get the above screen where we click on the “Create Policy” option.



We get the above page. In the “Visual” section in the “Select a service” section, select “S3” from the drop-down list.



Select the checkbox for “All S3 actions”



Select “All” resources. Click on Next.

IAM > Policies > Create policy

Step 1
Specify permissions

Step 2
Review and create

Specify permissions [Info](#)

Add permissions by selecting services, actions, resources, and conditions. Build permission statements using the JSON editor.

Policy editor

Visual **JSON** Actions

```
1 {  
2   "Version": "2012-10-17",  
3   "Statement": [  
4     {  
5       "Sid": "VisualEditor0",  
6       "Effect": "Allow",  
7       "Action": "s3:*",  
8       "Resource": "*" }  
9   ]  
10 }  
11 }
```

+ Add new statement

Edit statement
Statement1 Remove

Add actions

Choose a service
Filter services

Included
S3

Available
AMP
API Gateway
API Gateway V2
ASC
Access Analyzer
Account

Add a resource Add

Add a condition (optional) Add

JSON Ln 7, Col 14 6034 of 6144 characters remaining

In the JSON section, change the name of “Sid” to the name of the user i.e. admin.

Policy name
Enter a meaningful name to identify this policy.
policy1
Maximum 128 characters. Use alphanumeric and '+', '@', '-' characters.

Description - optional
Add a short explanation for this policy.
all access to s3
Maximum 1,000 characters. Use alphanumeric and '+', '@', '-' characters.

Permissions defined in this policy [Info](#)

Permissions defined in this policy document specify which actions are allowed or denied. To define permissions for an IAM identity (user, user group, or role), attach a policy to it

Search

Allow (1 of 420 services) Show remaining 419 services

Service	Access level	Resource	Request condition
S3	Full access	All resources	None

Add tags - optional [Info](#)

Tags are key-value pairs that you can add to AWS resources to help identify, organize, or search for resources.

No tags associated with the resource.

Add new tag
You can add up to 50 more tags.

Cancel Previous **Create policy**

Give the policy a name and give a short description of the policy. Check the permissions defined for the policy. Then click on “Create policy”

Services Search [Alt+S]

entity and Access management (IAM)

Search IAM

Dashboard
Access management
User groups
Users
Roles
Policies
Identity providers
Account settings

Policy policy1 created. View policy

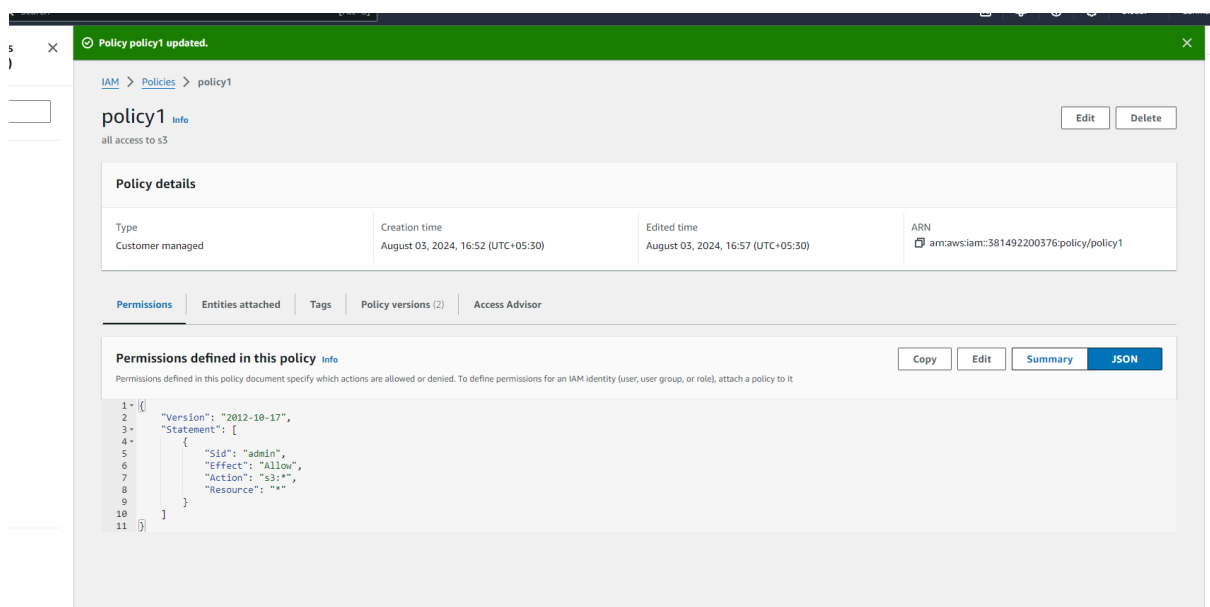
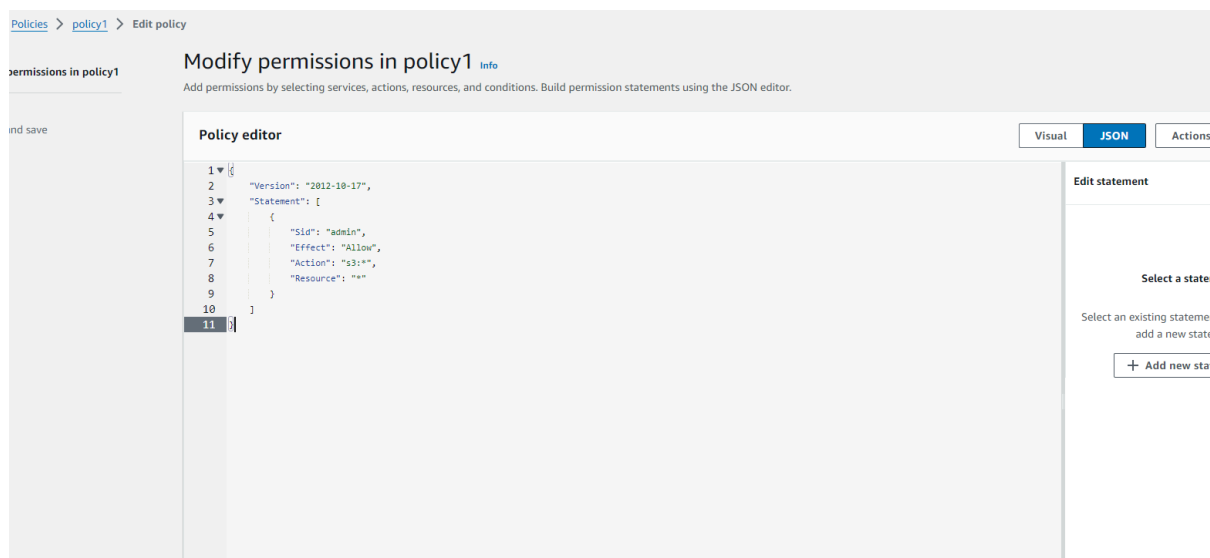
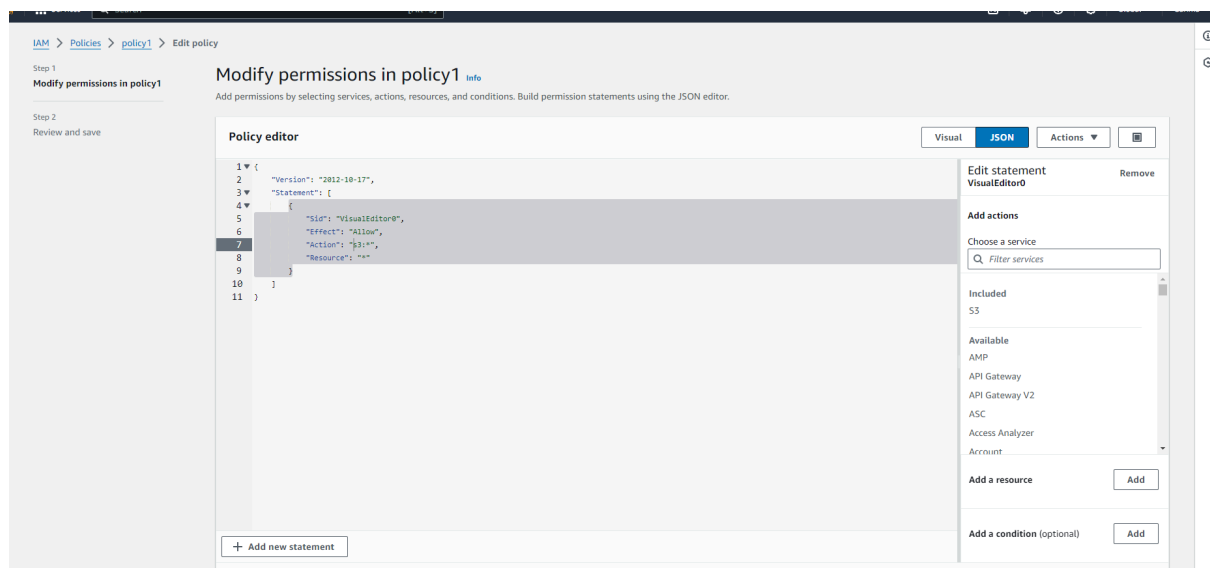
Policies (1222) [Info](#)

A policy is an object in AWS that defines permissions.

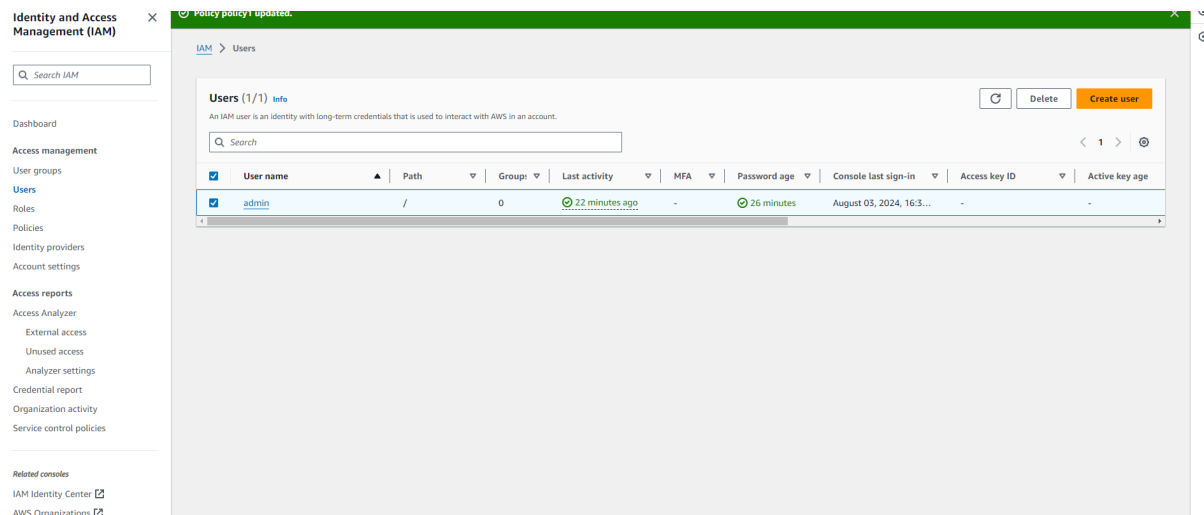
Filter by Type
policy1 All types 1 match

Policy name	Type	Used as	Description
policy1	Customer managed	None	all access to s3

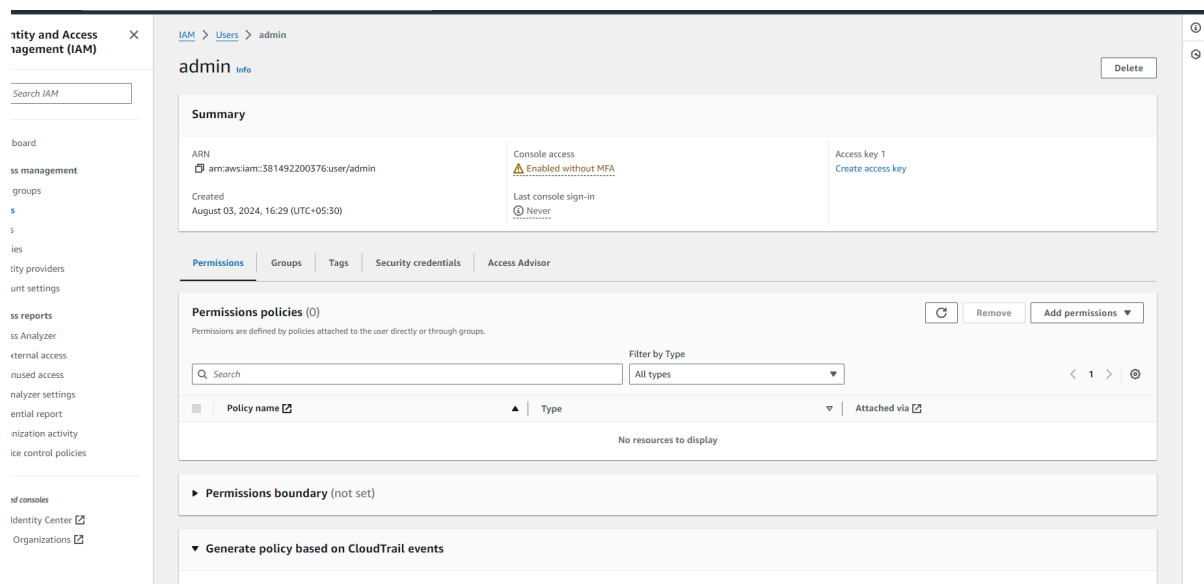
The policy is created.



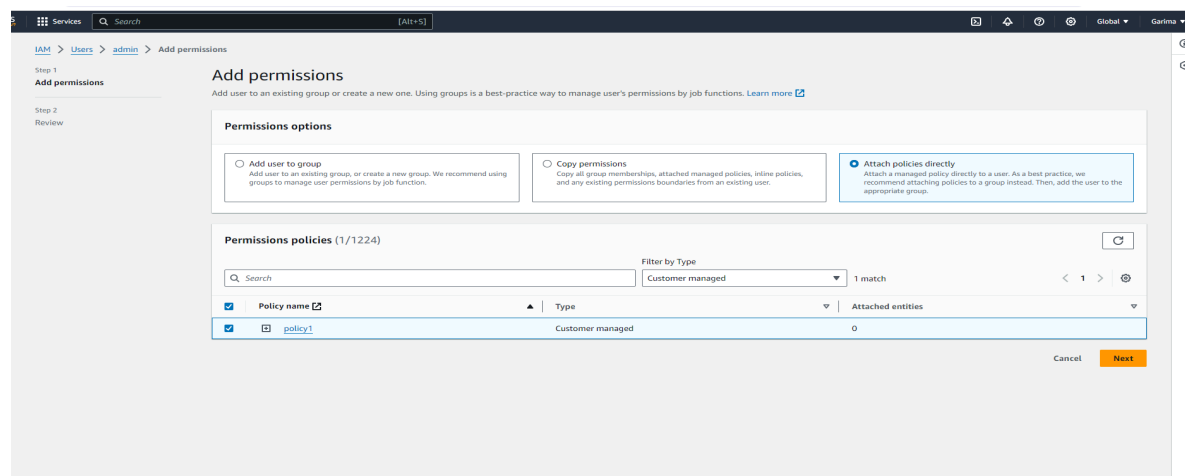
In case the policy needs to be edited, select the policy, select edit, make the edits, and update the policy.



Go to users and select the admin user created before.



In the “Permissions policies” section, select “Add permissions” from the drop-down at the right corner.

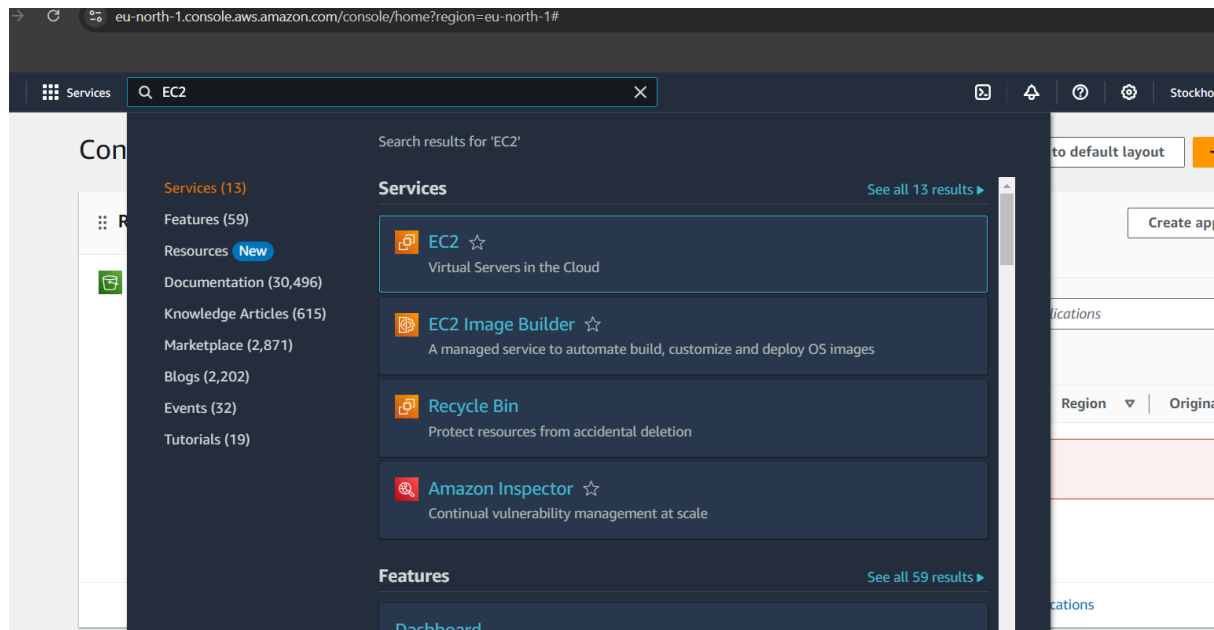


Select the policy created and above from the options, and select “Attach policies directly”. Click on “Next”

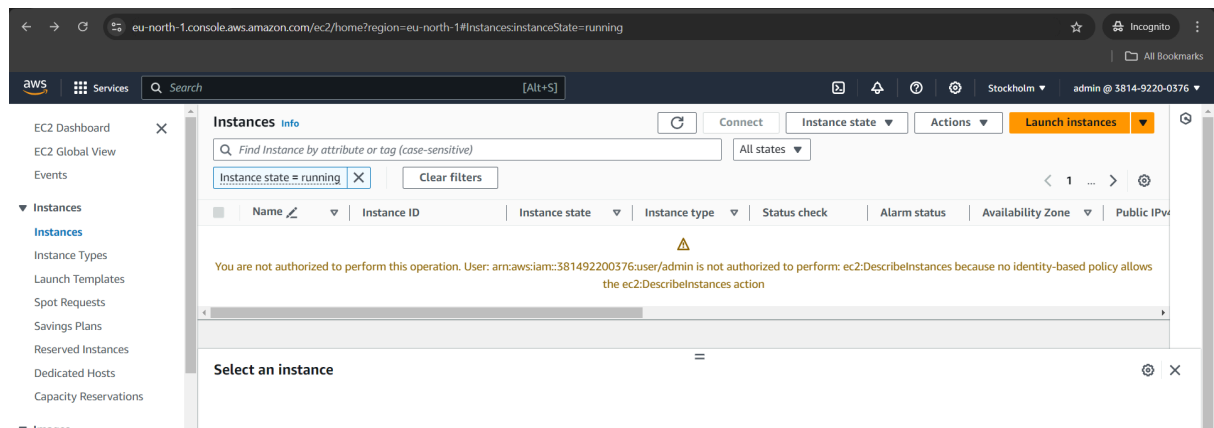
Check if the correct policy has been selected and then click on “Add permissions”.

Then go back to the incognito page of the user “admin” account, and click on create bucket. This time a bucket will be created.

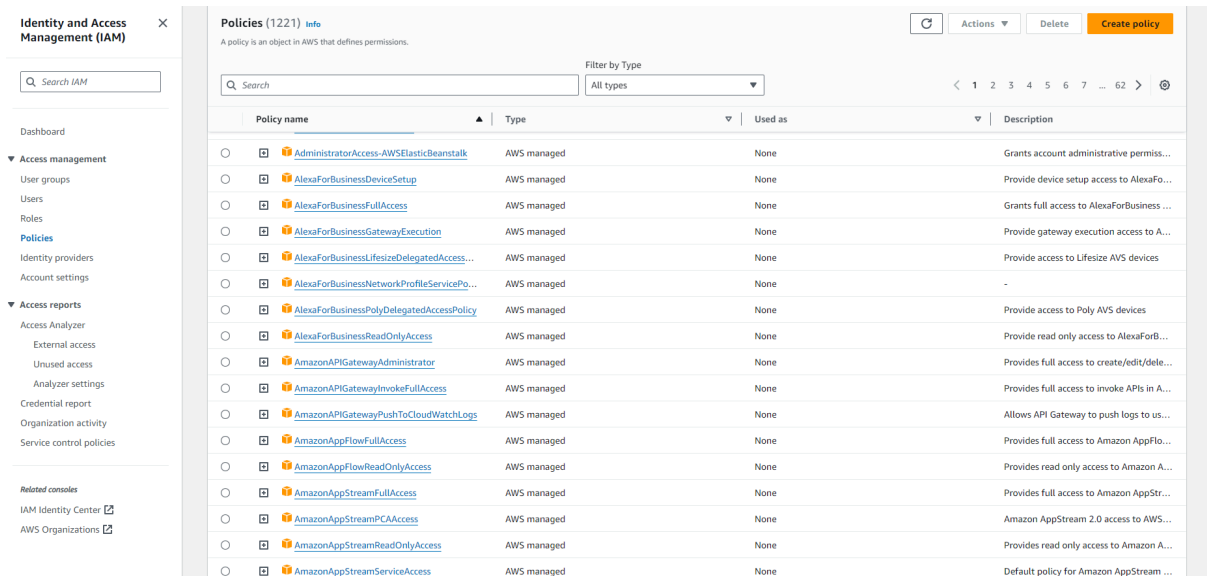
Now, we try to create an instance using the EC2 service.



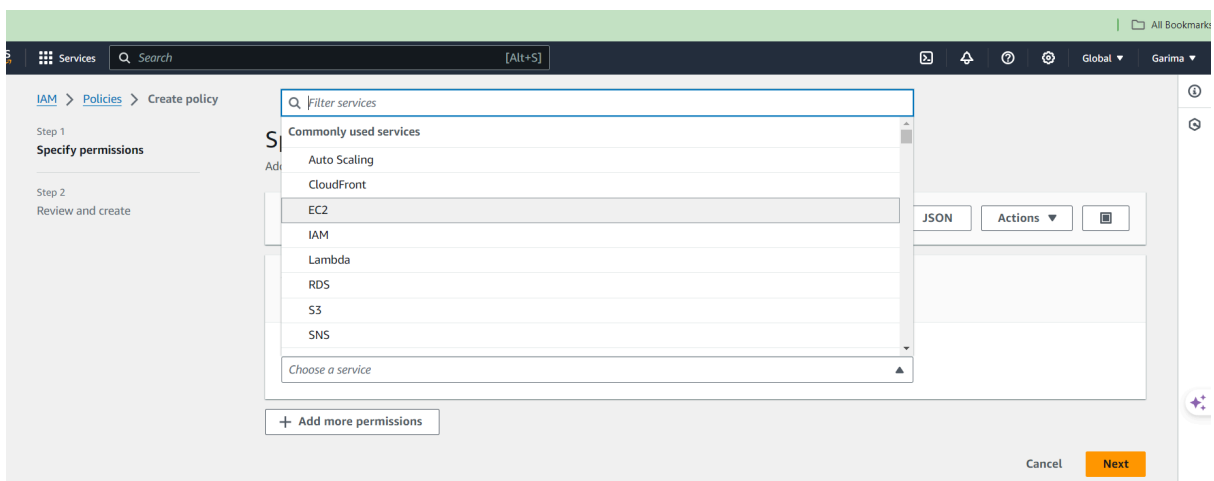
Select EC2 service.



When trying to create an instance, we get the above message stating that no policy permits the user “admin” to create an instance.



Go back to the root user's account. On the left side, select "Policies" from the Access Management section. We get the above screen where we click on the "Create Policy" option.



We get the above page. In the "Visual" section in the "Select a service" section, select "EC2" from the drop-down list.

▼ **EC2** Allow All actions

Specify what actions can be performed on specific resources in EC2.

▼ **Actions allowed**

Specify actions from the service to be allowed.

Effect
☒ Allow ☐ Deny

Manual actions | [Add actions](#)

☒ All EC2 actions (ec2:*)

Access level

- ▶ List (Selected 176/176)
- ▶ Read (Selected 36/36)
- ▶ Write (Selected 422/422)
- ▶ Permissions management (Selected 5/5)
- ▶ Tagging (Selected 2/2)

[Expand all](#) | [Collapse all](#)

⚠ Dependent permissions not selected.
 To grant permissions for the selected resource actions, including additional dependent actions might be required.

Select All EC2 actions to permit the user to perform all actions in the EC2 service.

▼ **Resources**

Specify resource ARNs for these actions.

☒ All
☐ Specific

⚠ The all wildcard "*" may be overly permissive for the selected actions. Allowing specific ARNs for these service resources can improve security.

▶ **Request conditions - optional**
 Actions on resources are allowed or denied only when these conditions are met.

[+ Add more permissions](#)

🛡 Security: 0 ❌ Errors: 0 ⚠ Warnings: 0 💡 Suggestions: 0

[Cancel](#) [Next](#)

In the “Resources” section, select the “All” option. Then click “Next”.

Add permissions by selecting services, actions, resources, and conditions. Build permission statements using the JSON editor.

Policy editor Visual **JSON** Actions ▼ 🛡

```

1 {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Sid": "VisualEditor0",
6       "Effect": "Allow",
7       "Action": "ec2:*",
8       "Resource": "*"
9     }
10  ]
11 }
  
```

Edit statement
VisualEditor0 [Remove](#)

Add actions

Choose a service

Included

EC2

Available

AMP
 API Gateway
 API Gateway V2
 ASC

Then in the “JSON” section, change the name of Sid from “VisualEditor” to the user’s name i.e. “admin”.

+ Add new statement

Available
AMP
API Gateway
API Gateway V2
ASC
Access Analyzer
Account

Add a resource
Add

Add a condition (optional)
Add

JSON Ln 5, Col 16
6041 of 6144 characters remaining

Security: 0
Errors: 0
Warnings: 0
Suggestions: 0
Check for new access

Cancel
Next

Click Next.

IAM > Policies > Create policy

Step 1
Specify permissions

Step 2
Review and create

Review and create

Review the permissions, specify details, and tags.

Policy details

Policy name
Enter a meaningful name to identify this policy.
EC2
Maximum 128 characters. Use alphanumeric and '+=, @-.' characters.

Description - *optional*
Add a short explanation for this policy.
all actions of EC2
Maximum 1,000 characters. Use alphanumeric and '+=, @-.' characters.

Give the policy a name and a short description.

Permissions defined in this policy
Edit

Permissions defined in this policy document specify which actions are allowed or denied. To define permissions for an IAM identity (user, user group, or role), attach a policy to it

Search

Allow (1 of 420 services)
Show remaining 419 services

Service	Access level	Resource	Request condition
EC2	Full access	All resources	None

Add tags - optional

Tags are key-value pairs that you can add to AWS resources to help identify, organize, or search for resources.

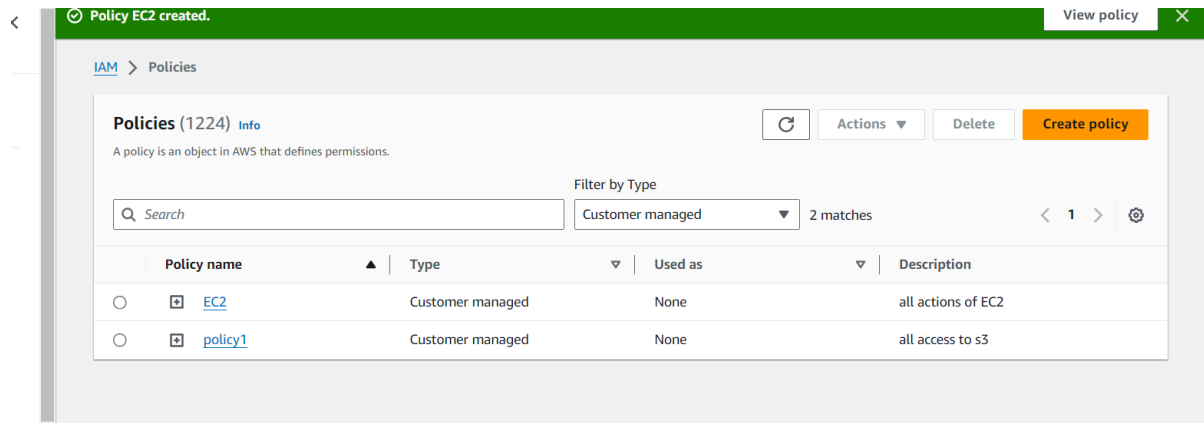
No tags associated with the resource.

Add new tag

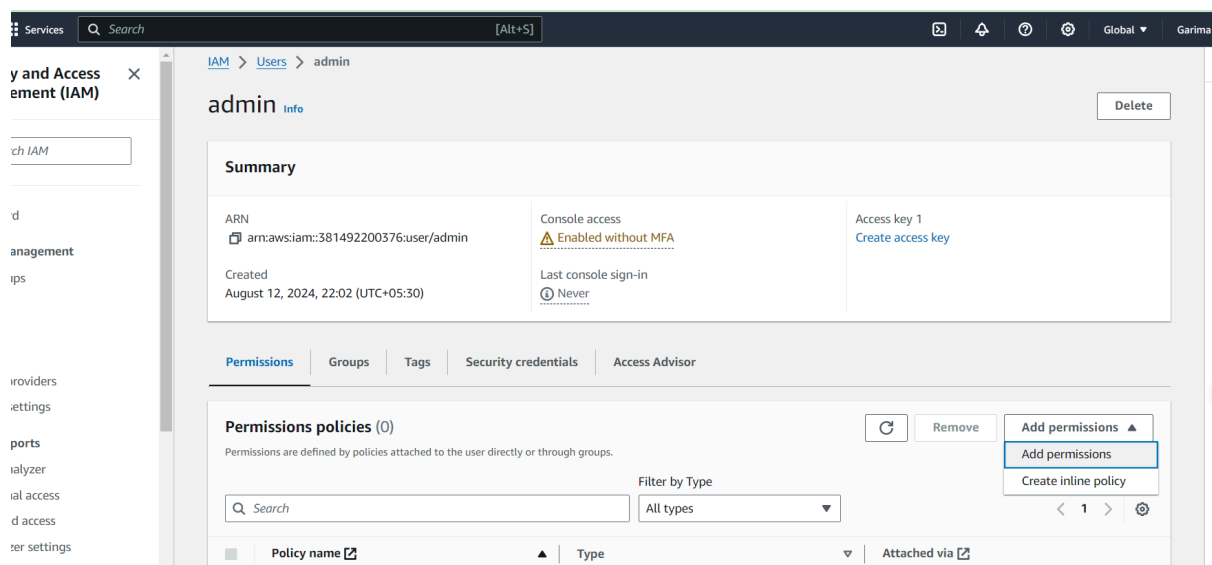
You can add up to 50 more tags.

Cancel
Previous
Create policy

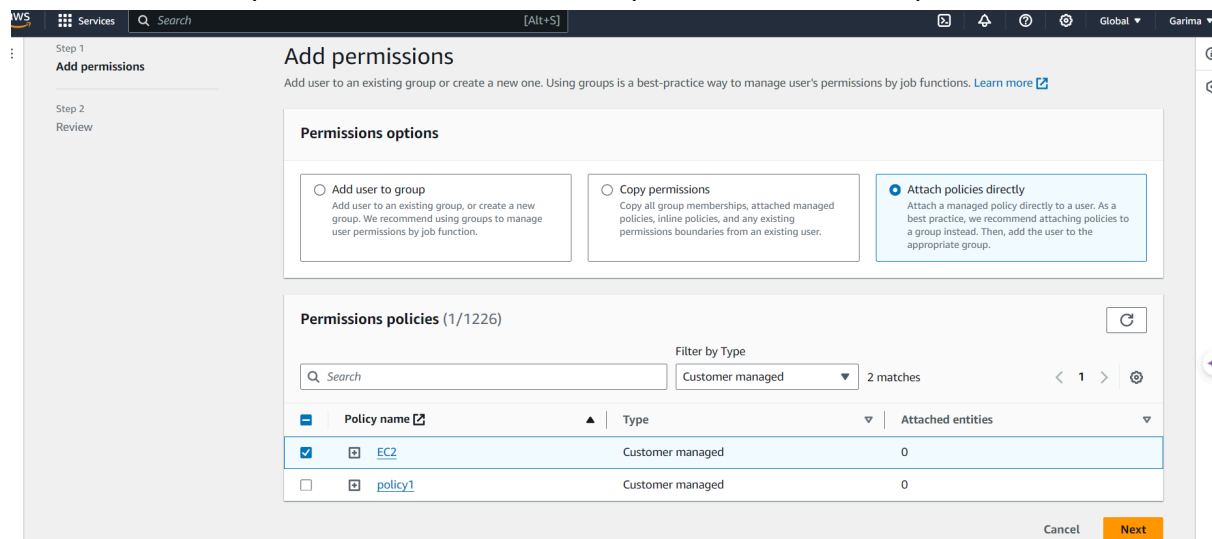
Click Create Policy.



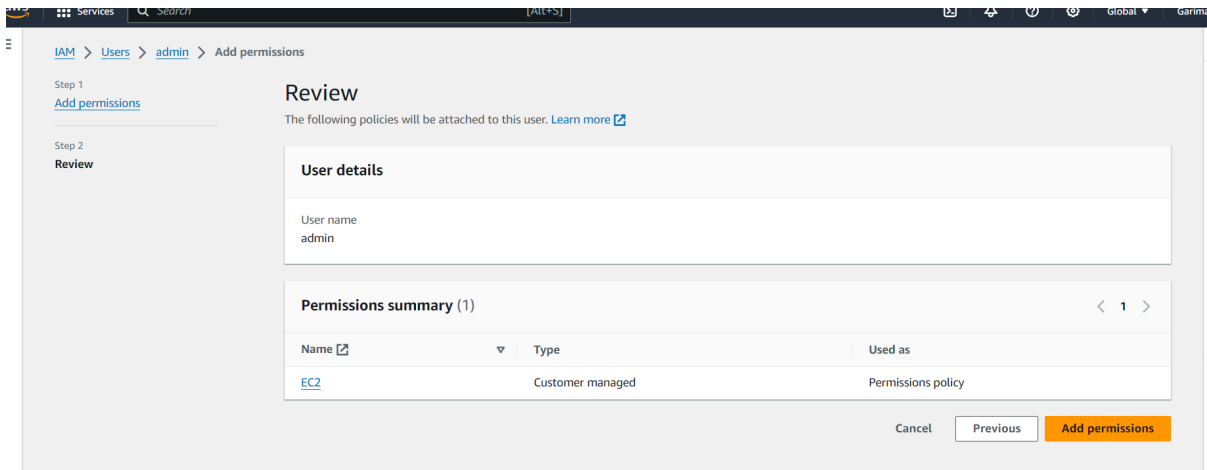
Here, we get to see the policy we created.
Go to users and select admin user.



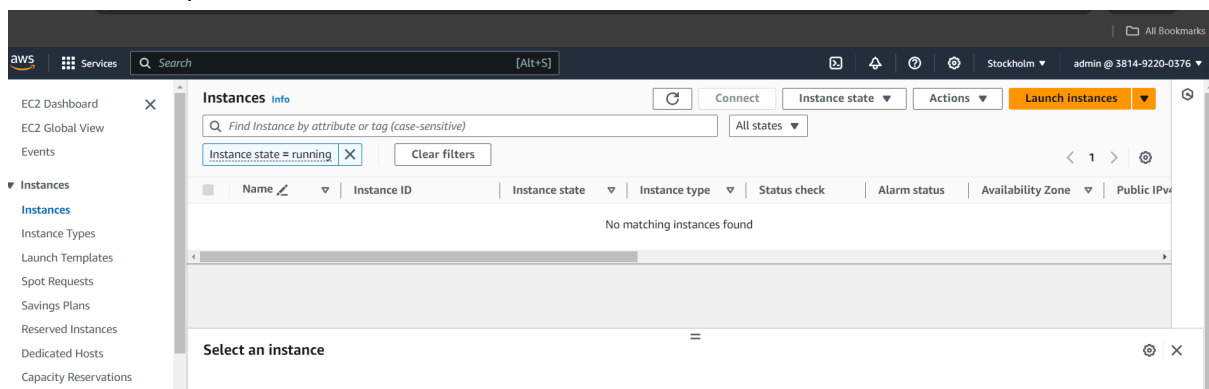
In the “Permission policies” section, select “Add permissions” from drop-down.



Select the “Attach policies directly” option. In “Permission policies” filter as “Customer managed” and select the policy “EC2”. Click “Next”.



Click on “Add permissions”.



After going back to the incognito tab where the admin user's account is open, refresh the page. Now no message regarding permission issues is shown, indicating we can create an instance now from this account.