IAM : Users & Groups

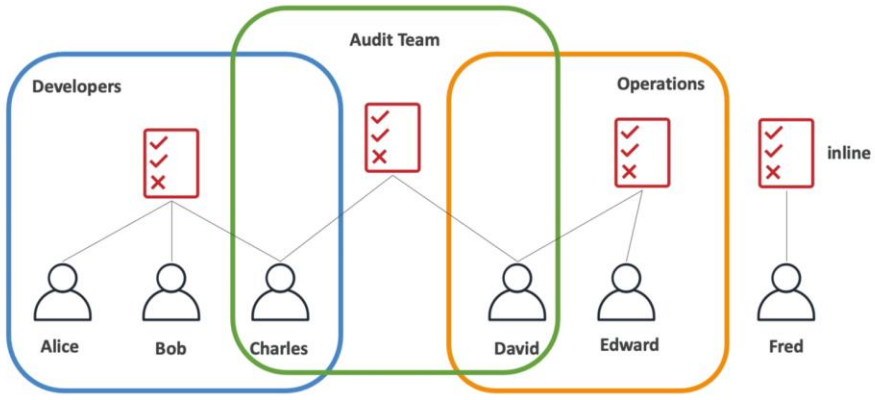
- IAM = Identity and Access Management, **Global** service
- **Root account** created by default, shouldn't be used or shared
- **Users** are people within your organization, and can be grouped
- **Groups** only contain users, not other groups
- Users don't have to belong to a group, and user can belong to multiple groups

IAM: Permissions

- **Users or Groups** can be assigned JSON documents called policies
- These policies define the **permissions** of the users
- In AWS you apply the **least privilege principle**: don't give more permissions than a user needs

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "ec2:Describe*",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "elasticloadbalancing:Describe*",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "cloudwatch:ListMetrics",
                "cloudwatch:GetMetricStatistics",
                "cloudwatch:Describe*"
            ],
            "Resource": "*"
        }
    ]
}
```

IAM Policies inheritance



IAM Policies Structure

- Consists of
  - Version: policy language version, always include "2012-10-17"
  - Id: an identifier for the policy (optional)
  - Statement: one or more individual statements (required)
- Statements consists of
  - Sid: an identifier for the statement (optional)
  - Effect: whether the statement allows or denies access (Allow, Deny)
  - Principal: account/user/role to which this policy applied to
  - Action: list of actions this policy allows or denies
  - Resource: list of resources to which the actions applied to
  - Condition: conditions for when this policy is in effect (optional)

```
{
    "Version": "2012-10-17",
    "Id": "S3-Account-Permissions",
    "Statement": [
        {
            "Sid": "1",
            "Effect": "Allow",
            "Principal": {
                "AWS": ["arn:aws:iam::123456789012:root"]
            },
            "Action": [
                "s3:GetObject",
                "s3:PutObject"
            ],
            "Resource": ["arn:aws:s3:::mybucket/*"]
        }
    ]
}
```

IAM: Password Policy

- Strong passwords = higher security for your account
- In AWS, you can setup a password policy:
  - Set a minimum password length
  - Require specific character types:
    - including uppercase letters
    - lowercase letters
    - numbers
    - non-alphanumeric characters
  - Allow all IAM users to change their own passwords
  - Require users to change their password after some time (password expiration)
  - Prevent password re-use

Multi Factor Authentication -MFA

- Users have access to your account and can possibly change configurations or delete resources in your AWS account
- You want to protect your Root Accounts and IAM users
- MFA = password *you know* + security device *you own*

- Main benefit of MFA:
  if a password is stolen or hacked, the account is not compromised

MFA Device options in AWS

Virtual MFA device

Google Authenticator
(phone only)

Authy
(multi-device)

Support for multiple tokens on a single device.

Universal 2nd Factor (U2F) Security Key

YubiKey by Yubico (3rd party)

Support for multiple root and IAM users using a single security key

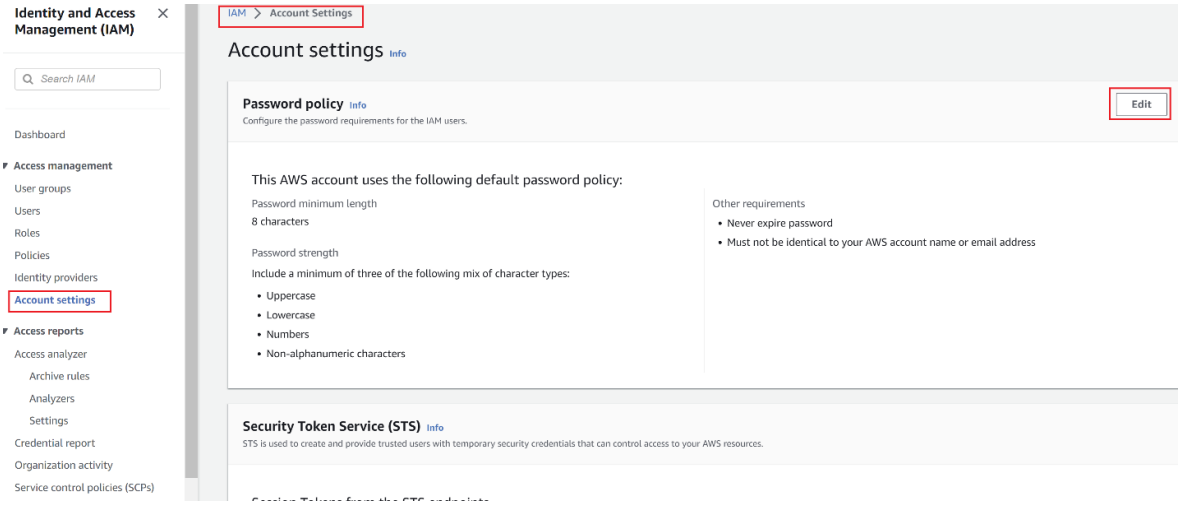
Hardware Key Fob MFA Device

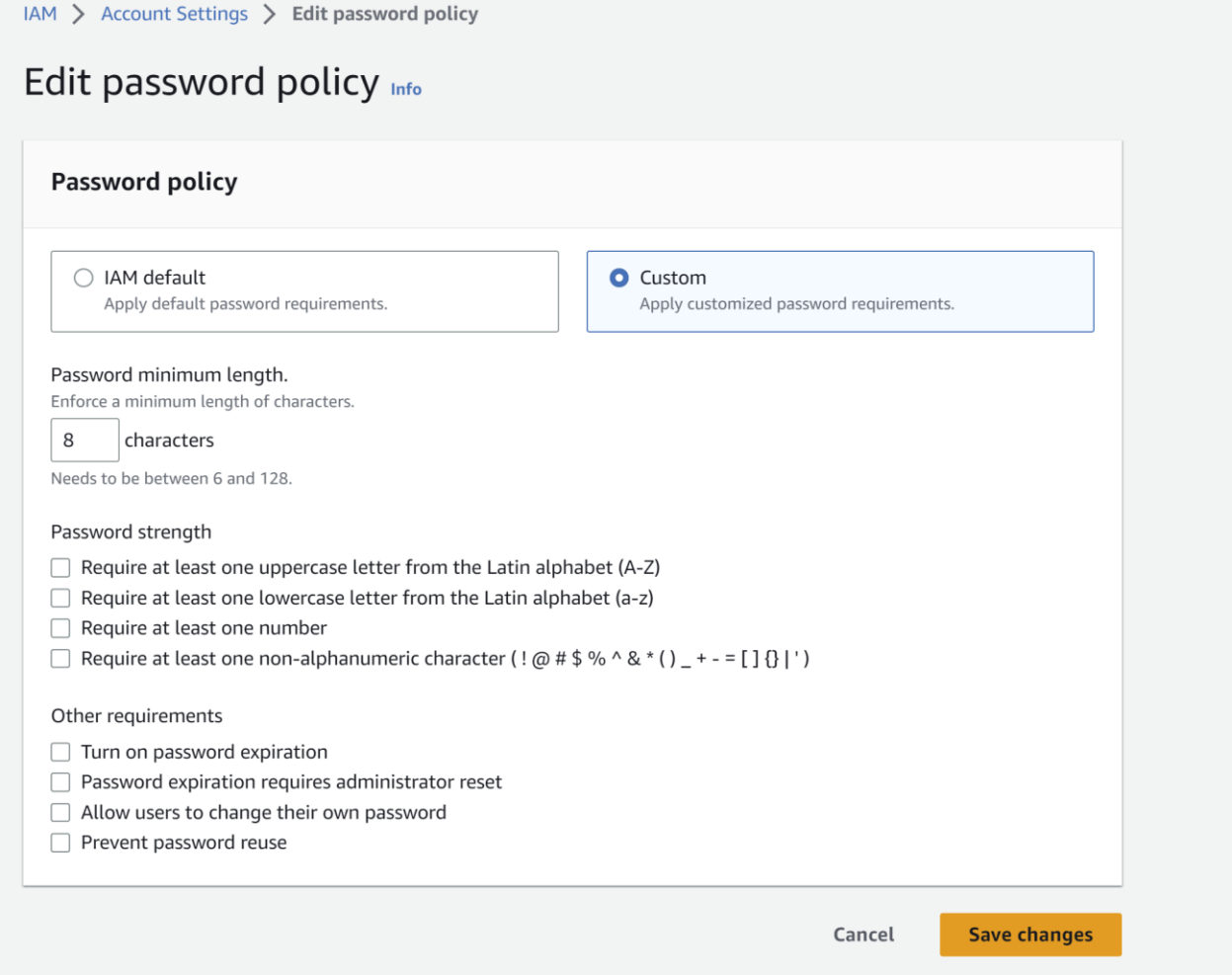
Provided by Gemalto (3rd party)

Hardware Key Fob MFA Device for AWS GovCloud (US)

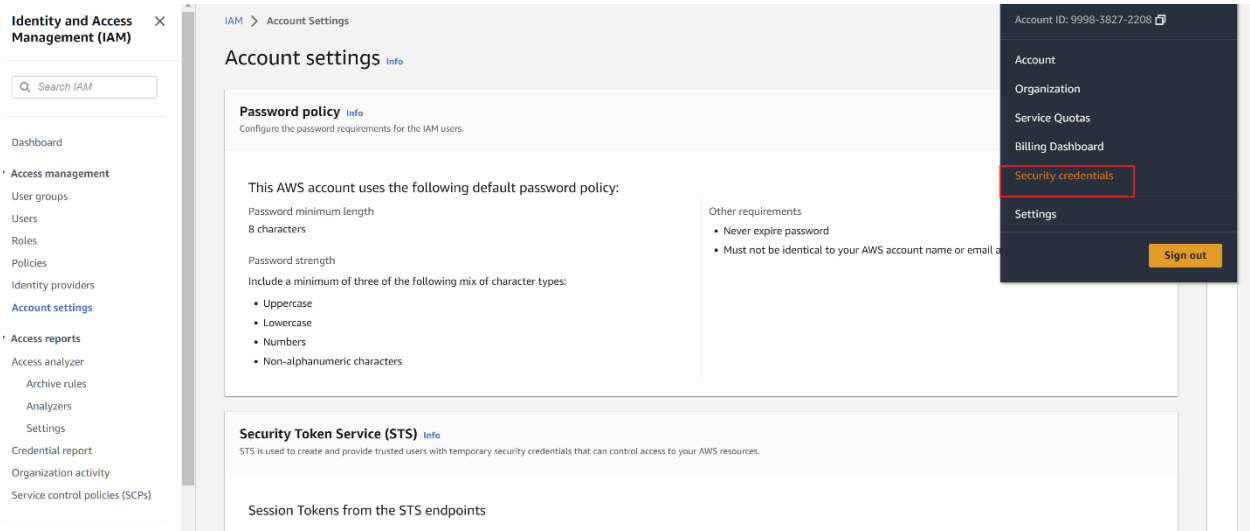
Provided by SurePassID (3rd party)

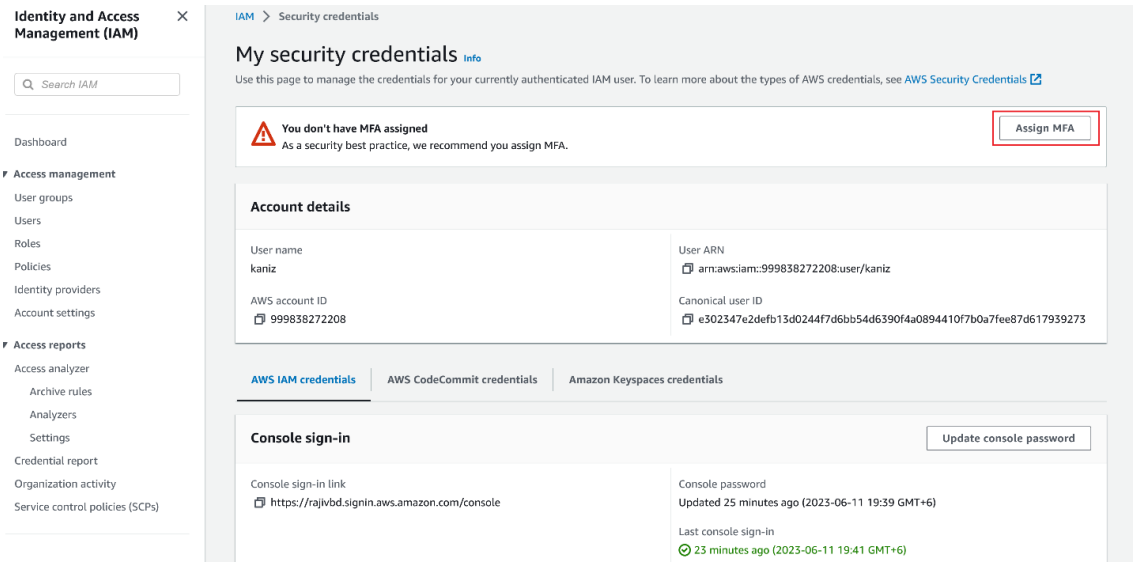
LAB: Password policy setting

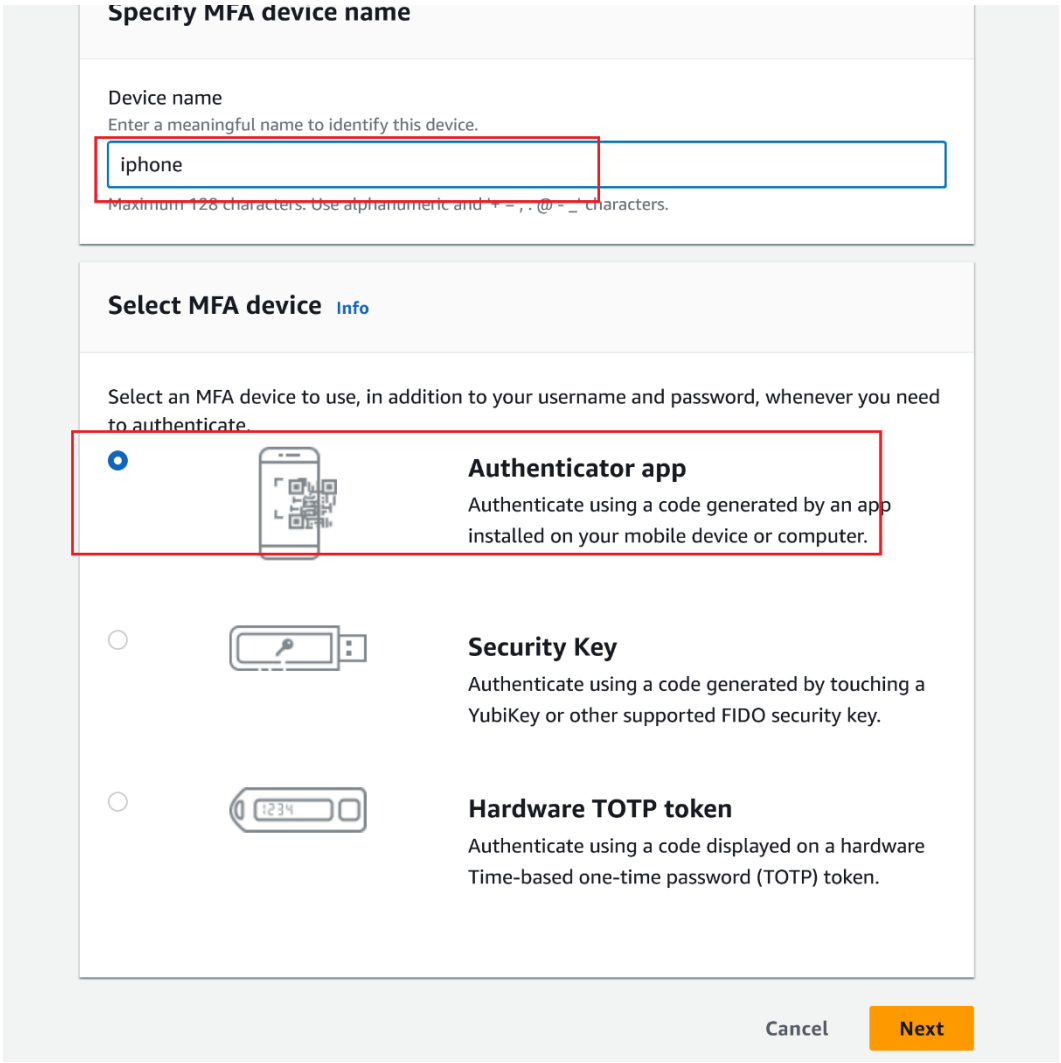




LAB: MFA Setting




Identity and Access Management (IAM)
Search IAM
Dashboard
Access management
User groups
Users
Roles
Policies
Identity providers
Account settings
Access reports
Access analyzer
Archive rules
Analyzers
Settings
Credential report
Organization activity
Service control policies (SCPs)
IAM > Security credentials
My security credentials Info
Use this page to manage the credentials for your currently authenticated IAM user. To learn more about the types of AWS credentials, see AWS Security Credentials
You don't have MFA assigned
As a security best practice, we recommend you assign MFA.
Assign MFA
Account details
User name
kaniz
User ARN
arn:aws:iam::999838272208:user/kaniz
AWS account ID
999838272208
Canonical user ID
e302347e2defb13d0244f7d6bb54d6390f4a0894410f7b0a7fee87d617939273
AWS IAM credentials
AWS CodeCommit credentials
Amazon Keyspaces credentials
Console sign-in
Update console password
Console sign-in link
https://rajivbd.signin.aws.amazon.com/console
Console password
Updated 25 minutes ago (2023-06-11 19:39 GMT+6)
Last console sign-in
23 minutes ago (2023-06-11 19:41 GMT+6)


Specify MFA device name
Device name
Enter a meaningful name to identify this device.
iphone
Maximum 128 characters. Use alphanumeric and '+ = , . @ - _' characters.
Select MFA device Info
Select an MFA device to use, in addition to your username and password, whenever you need to authenticate.
Authenticator app
Authenticate using a code generated by an app installed on your mobile device or computer.
Security Key
Authenticate using a code generated by touching a YubiKey or other supported FIDO security key.
Hardware TOTP token
Authenticate using a code displayed on a hardware Time-based one-time password (TOTP) token.
Cancel
Next

The following app will allow you to set MFA in AWS

Virtual authenticator apps

Virtual authenticator apps implement the time-based one-time password (TOTP) algorithm and support multiple tokens on a single device. Virtual authenticators are supported for IAM users in the AWS GovCloud (US) Regions and in other AWS Regions. For more information about enabling virtual authenticators, see Enabling a virtual multi-factor authentication (MFA) device.

You can install apps for your smartphone from the app store that is specific to your type of smartphone. Some app providers also have web and desktop applications available. See the following table for examples.

Android	Twilio Authy Authenticator, Duo Mobile, LastPass Authenticator, Microsoft Authenticator, Google Authenticator, Symantec VIP
iOS	Twilio Authy Authenticator, Duo Mobile, LastPass Authenticator, Microsoft Authenticator, Google Authenticator, Symantec VIP

Set up your authenticator app

A virtual MFA device is an application running on your device that you can configure by scanning a QR code.

1. Install a compatible application such as Google Authenticator, Duo Mobile, or Authy app on your mobile device or computer.
   See a list of compatible applications
2. Show QR code
   Open your authenticator app, choose **Show QR code** on this page, then use the app to scan the code. Alternatively, you can type a secret key. Show secret key
3. Fill in two consecutive codes from your MFA device.
   MFA code 1: 213213
   MFA code 2: 312312

Cancel | Previous | Add MFA

How Can Users access AWS

- To access AWS, you have three options:
  - **AWS Management Console** (protected by password + MFA)
  - **AWS Command Line Interface (CLI):** protected by access keys
  - **AWS Software Developer Kit (SDK)** - for code: protected by access keys
- Access Keys are generated through the AWS Console
- Users manage their own access keys
- Access Keys are secret, just like a password. Don't share them
- Access Key ID ~= username
- Secret Access Key ~= password

AWS CLI

- A tool that enables you to interact with AWS services using commands in your command-line shell
- Direct access to the public APIs of AWS services
- You can develop scripts to manage your resources
- It's open-source https://github.com/aws/aws-cli
- Alternative to using AWS Management Console

```
➜  ~ aws s3 cp myfile.txt s3://ccp-mybucket/myfile.txt
upload: ./myfile.txt to s3://ccp-mybucket/myfile.txt
➜  ~ aws s3 ls s3://ccp-mybucket
2021-05-14 03:22:52          0 myfile.txt
➜  ~
```

AWS SDK

- AWS Software Development Kit (AWS SDK)
- Language-specific APIs (set of libraries)
- Enables you to access and manage AWS services programmatically
- Embedded within your application
- Supports
  - SDKs (JavaScript, Python, PHP, .NET, Ruby, Java, Go, Node.js, C++)
  - Mobile SDKs (Android, iOS, …)
  - IoT Device SDKs (Embedded C, Arduino, …)
- Example: AWS CLI is built on AWS SDK for Python

Your Application

LAB

Install AWS CLI

https://docs.aws.amazon.com/cli/latest/userguide/getting-started-install.html

after install to check run the command
aws cli --version

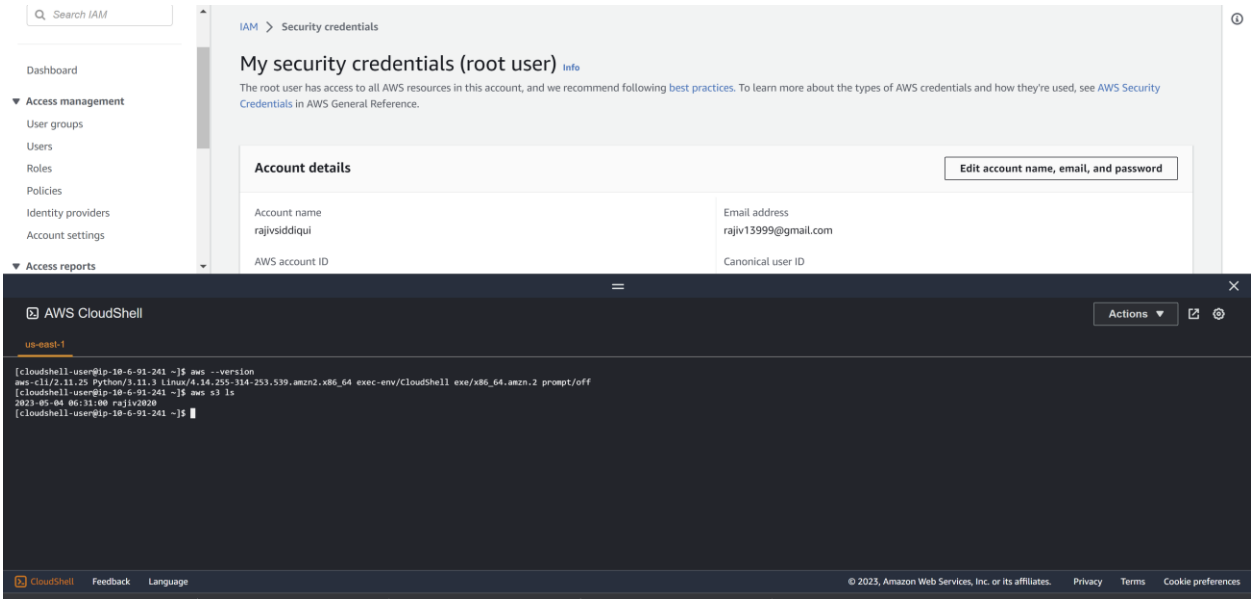
```
aweso@RAJIV-XPS17 MINGW64 /d/aws
$ aws --version
aws-cli/2.11.16 Python/3.11.3 Windows/10 exe/AMD64 prompt/off
```

Install aws cli in linux:

curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscliv2.zip"
unzip awscliv2.zip
sudo ./aws/install

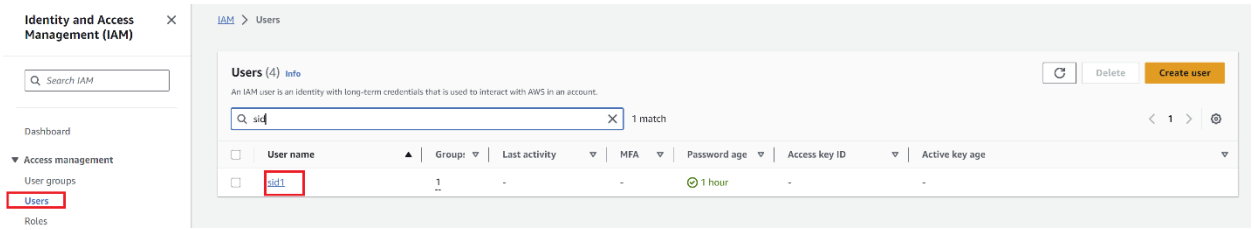
now check
aws --version

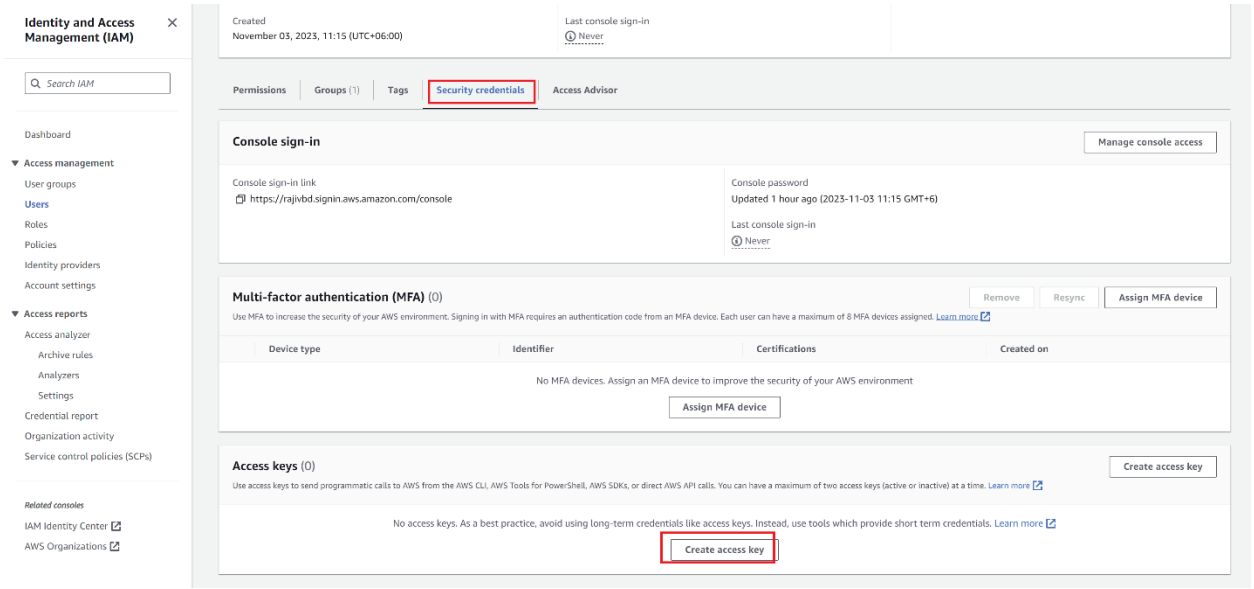
OR
We can use AWS CloudShell for using aws cli

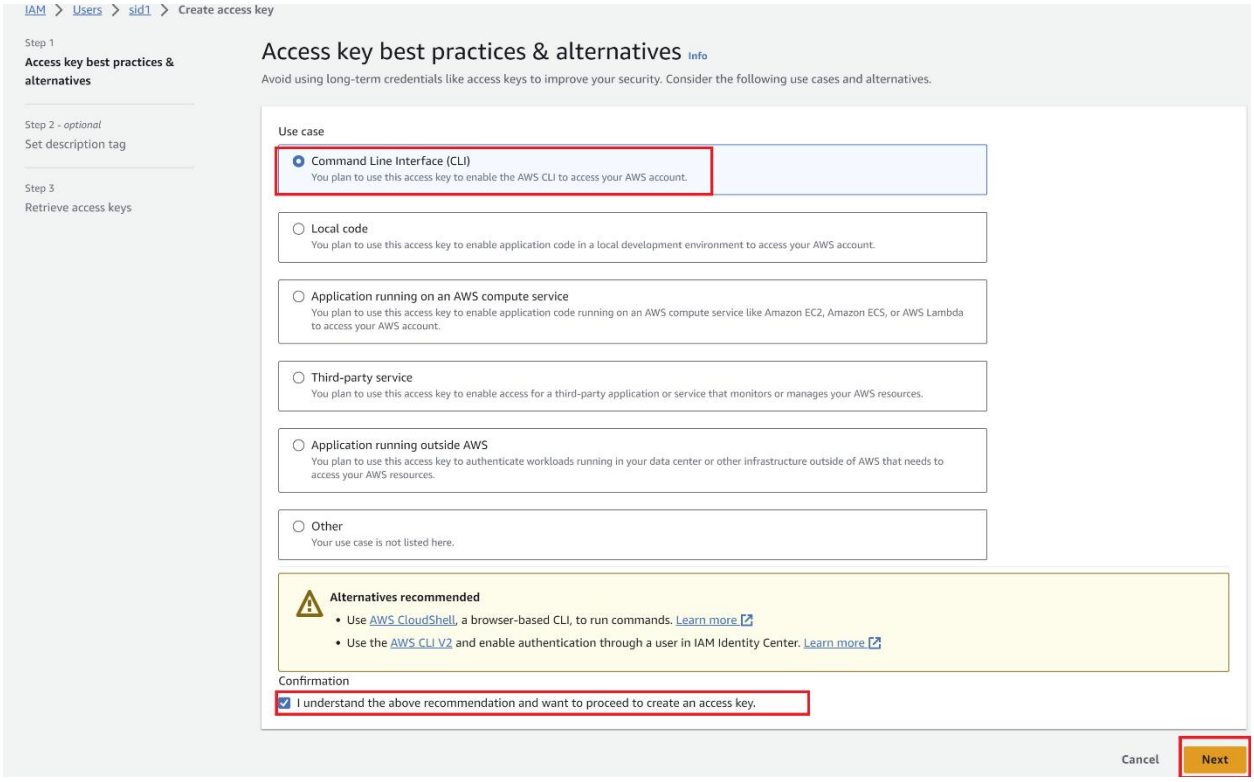


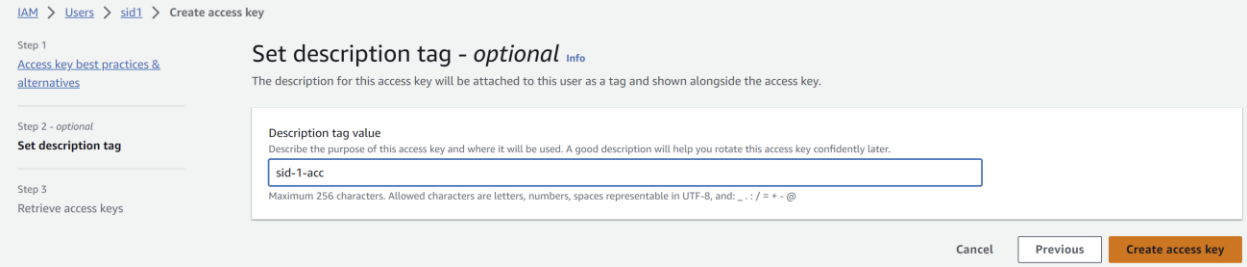
LAB: install and configure AWS cli

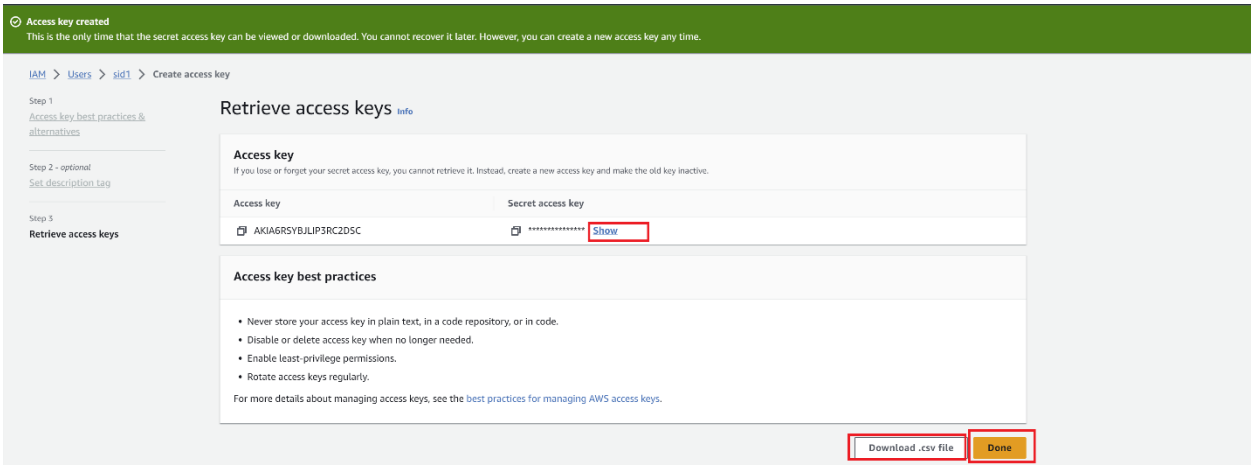
create a access key for a user







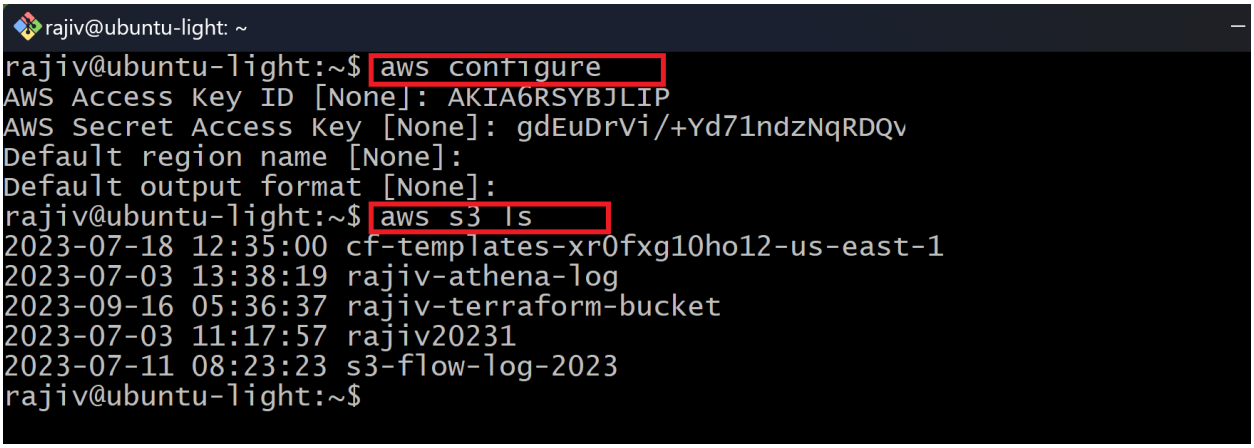




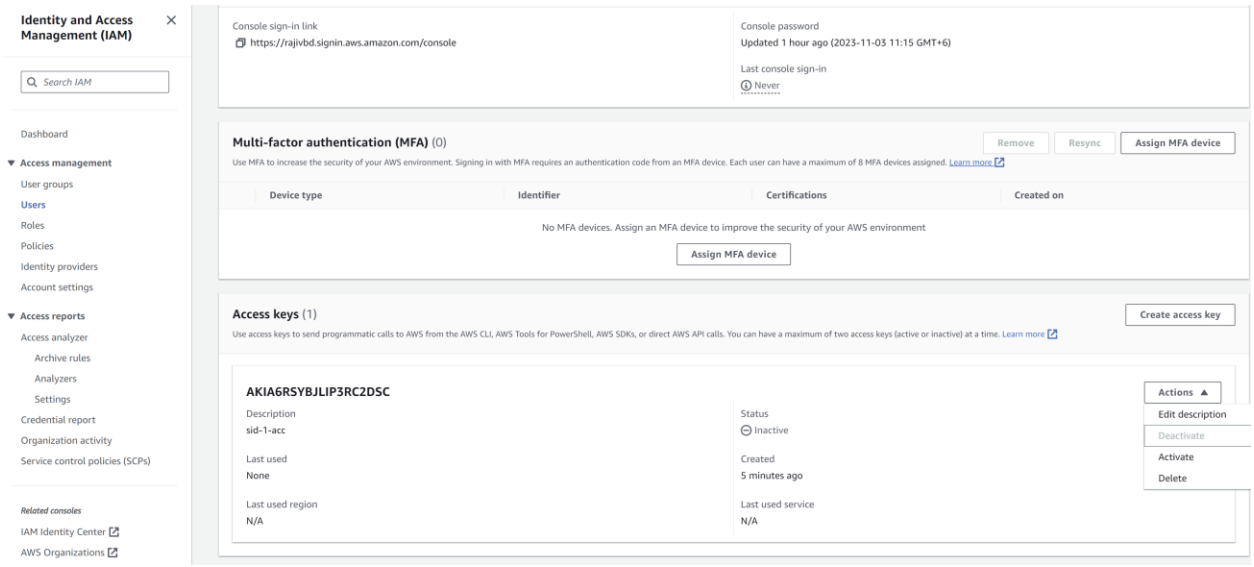
Now go to your linux pc command line and type access key and secret access key

#aws configure

#aws s3 ls - see your all bucket list

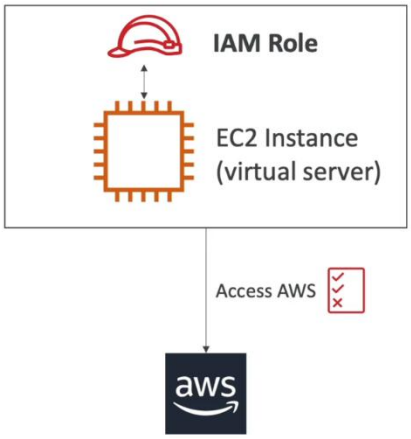


You can deactivate it inactive your access key and secret access key

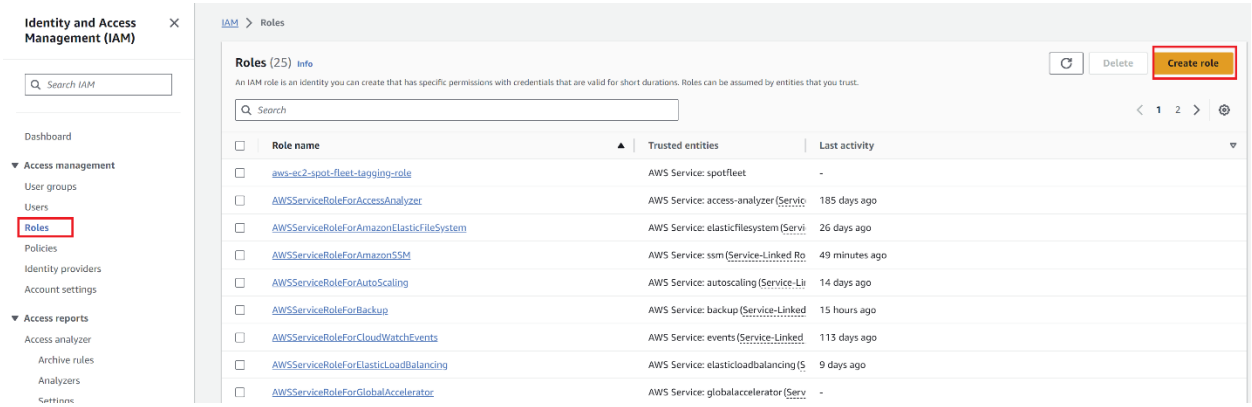


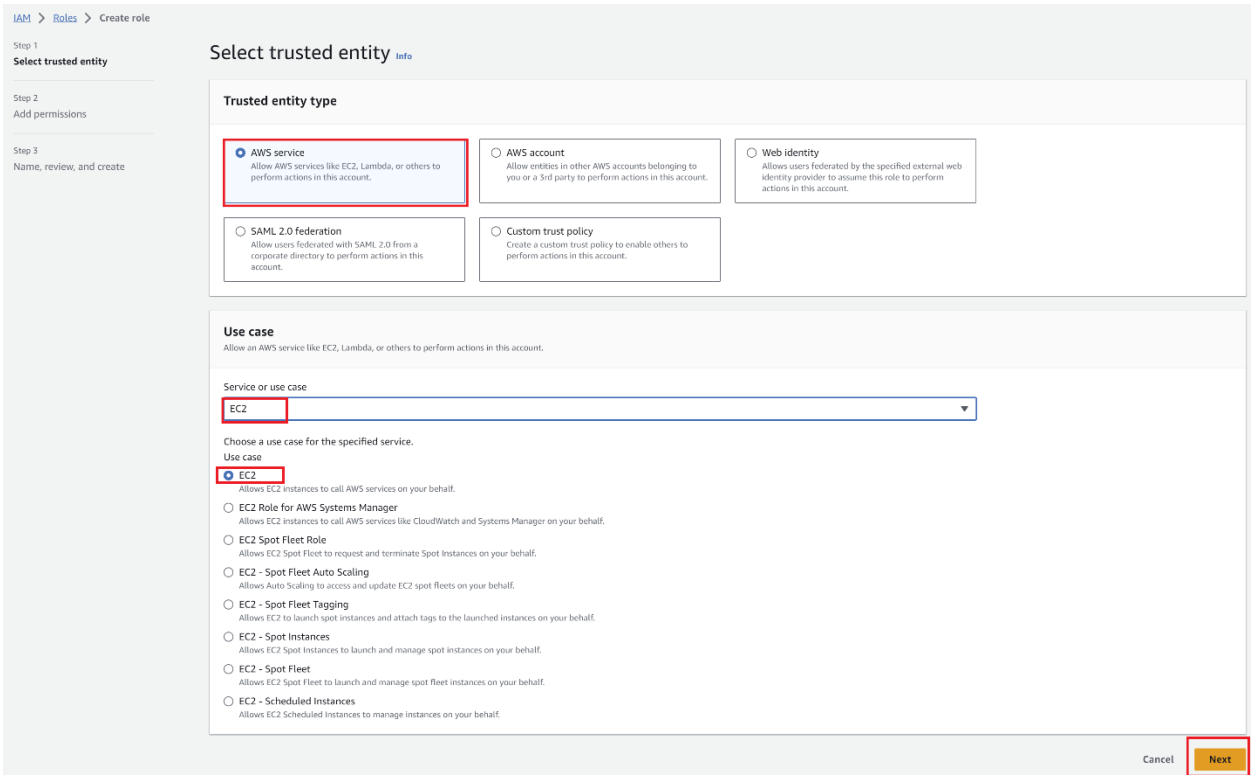
IAM Roles for Services

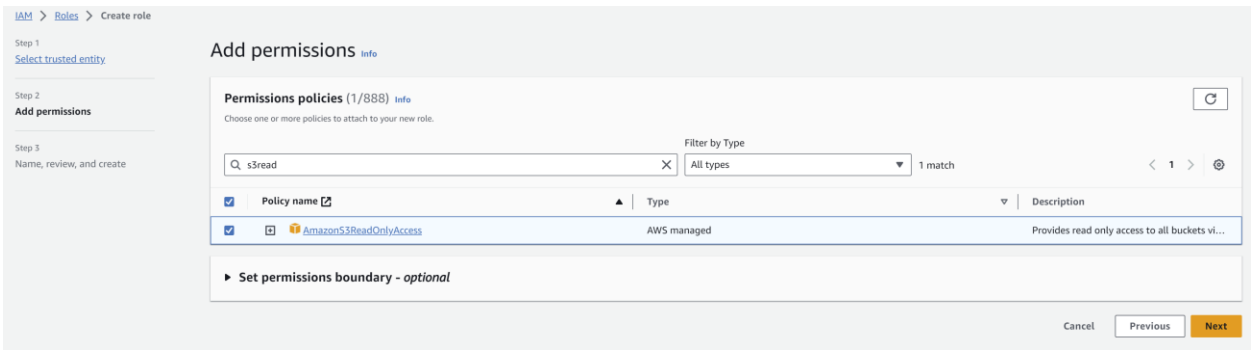
- Some AWS service will need to perform actions on your behalf
- To do so, we will assign **permissions** to AWS services with **IAM Roles**
- Common roles:
  - EC2 Instance Roles
  - Lambda Function Roles
  - Roles for CloudFormation

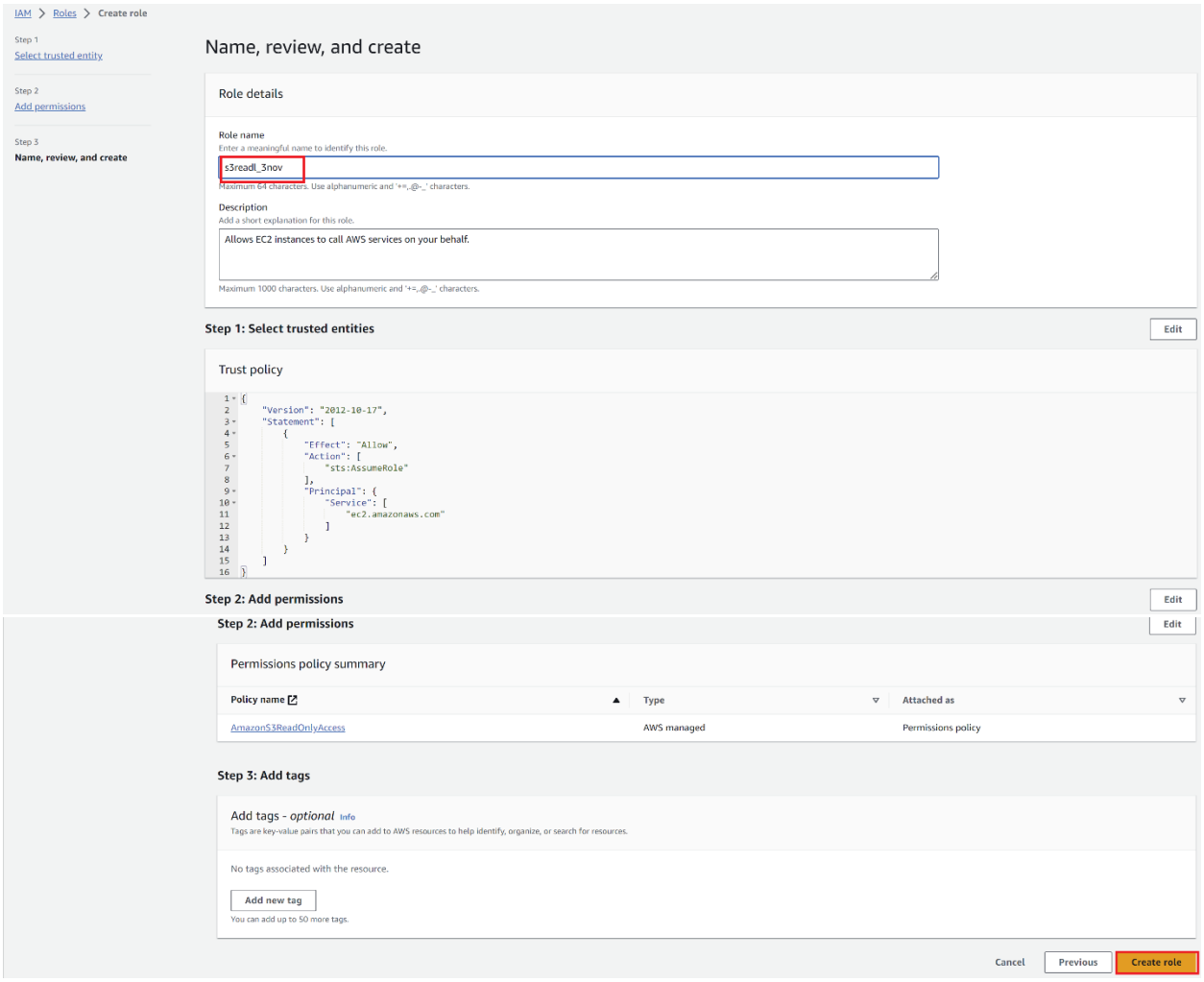


LAB: create an IAM Role




IAM > Roles > Create role
Step 1
Select trusted entity
Step 2
Add permissions
Step 3
Name, review, and create
Select trusted entity Info
Trusted entity type
AWS service
Allow AWS services like EC2, Lambda, or others to perform actions in this account.
AWS account
Allow entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.
Web identity
Allows users federated by the specified external web identity provider to assume this role to perform actions in this account.
SAML 2.0 federation
Allow users federated with SAML 2.0 from a corporate directory to perform actions in this account.
Custom trust policy
Create a custom trust policy to enable others to perform actions in this account.
Use case
Allow an AWS service like EC2, Lambda, or others to perform actions in this account.
Service or use case
EC2
Choose a use case for the specified service.
Use case
EC2
Allows EC2 instances to call AWS services on your behalf.
EC2 Role for AWS Systems Manager
Allows EC2 instances to call AWS services like CloudWatch and Systems Manager on your behalf.
EC2 Spot Fleet Role
Allows EC2 Spot Fleet to request and terminate Spot Instances on your behalf.
EC2 - Spot Fleet Auto Scaling
Allows Auto Scaling to access and update EC2 spot fleets on your behalf.
EC2 - Spot Fleet Tagging
Allows EC2 to launch spot instances and attach tags to the launched instances on your behalf.
EC2 - Spot Instances
Allows EC2 Spot Instances to launch and manage spot instances on your behalf.
EC2 - Spot Fleet
Allows EC2 Spot Fleet to launch and manage spot fleet instances on your behalf.
EC2 - Scheduled Instances
Allows EC2 Scheduled Instances to manage instances on your behalf.
Cancel
Next

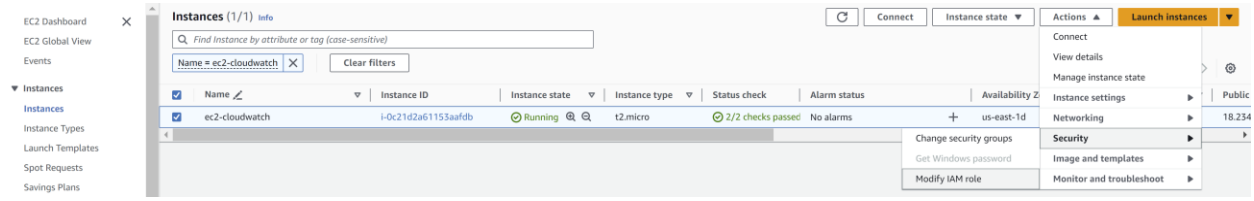

IAM > Roles > Create role
Step 1
Select trusted entity
Step 2
Add permissions
Step 3
Name, review, and create
Add permissions Info
Permissions policies (1/888) Info
Choose one or more policies to attach to your new role.
Filter by Type
s3read
All types
1 match
1
Policy name
Type
Description
AmazonS3ReadOnlyAccess
AWS managed
Provides read only access to all buckets vi...
Set permissions boundary - optional
Cancel
Previous
Next

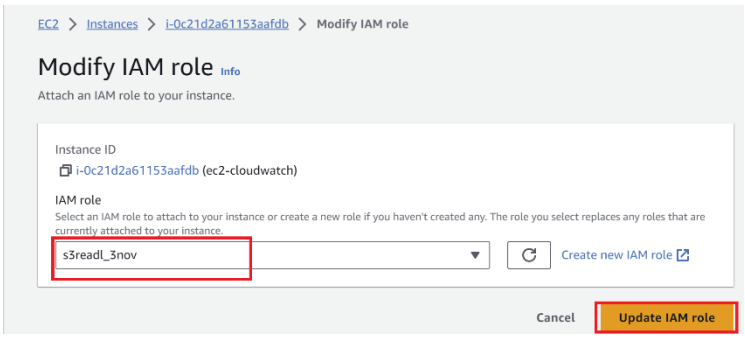


Now create a ec2 instance and run any aws cli command

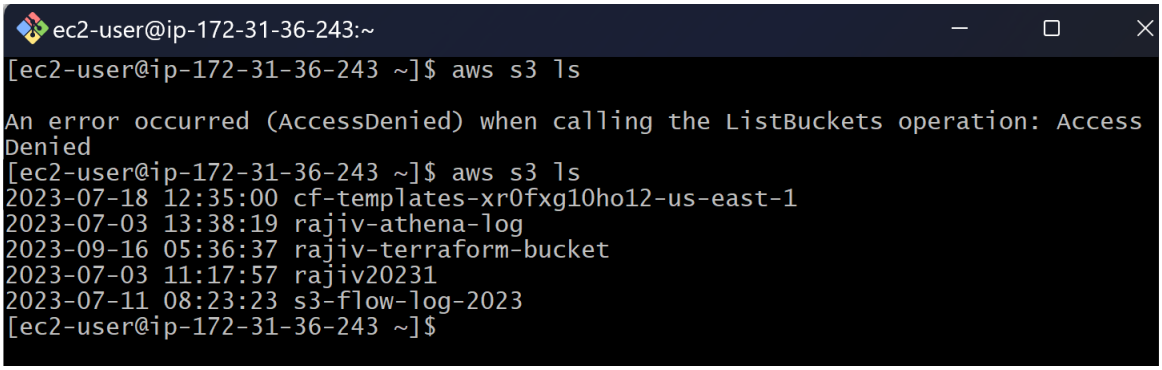
#aws s3 ls

Now attach the role and the run the same aws cli command

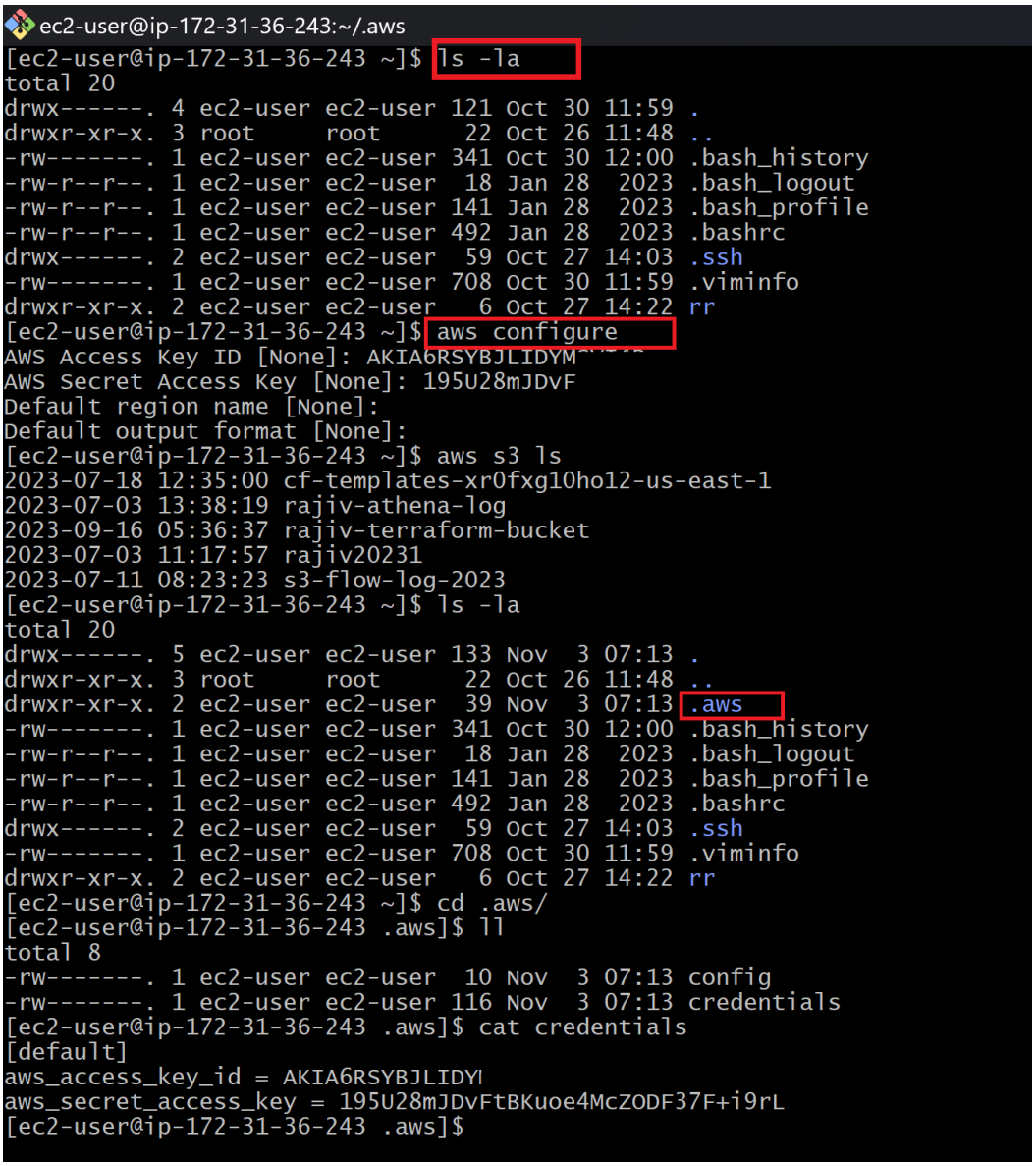




It looks like blow image



We can also configure it in this ec2 instance by running aws configure command but its not secure because your credentials will save it in ec2 instance

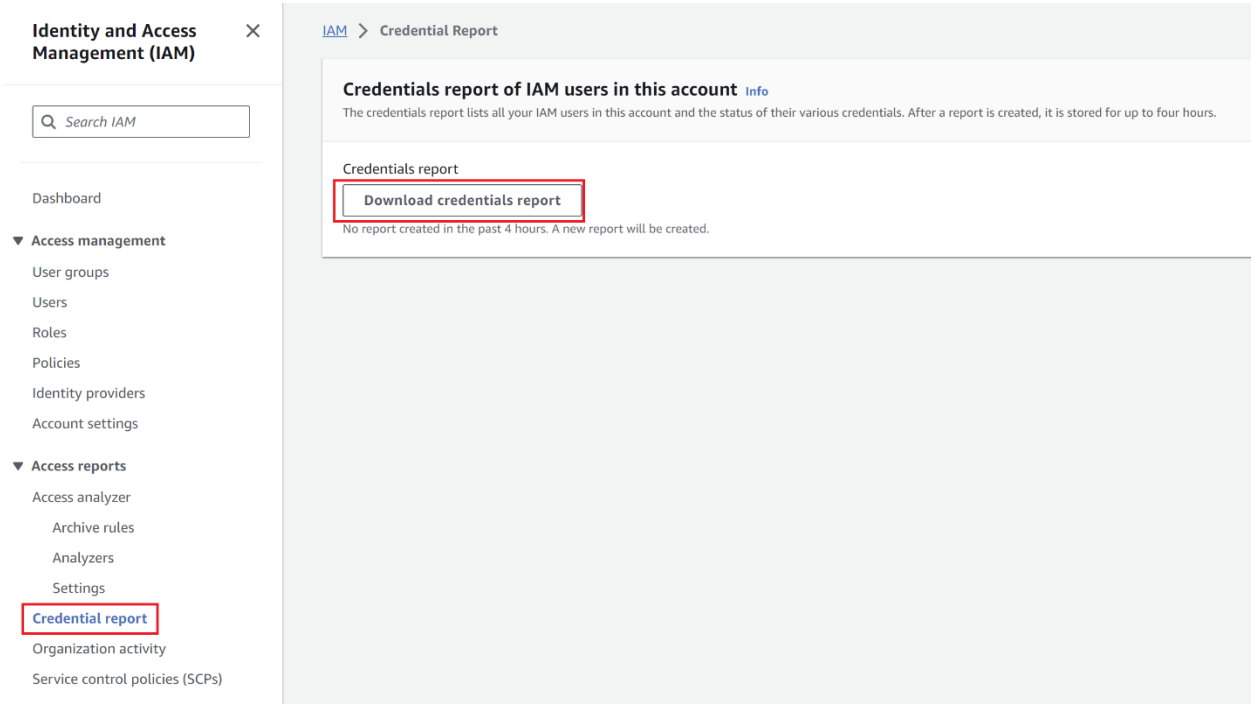


IAM Security Tools

- IAM Credentials Report (account-level)
  - a report that lists all your account's users and the status of their various credentials

- IAM Access Advisor (user-level)
  - Access advisor shows the service permissions granted to a user and when those services were last accessed.
  - You can use this information to revise your policies.

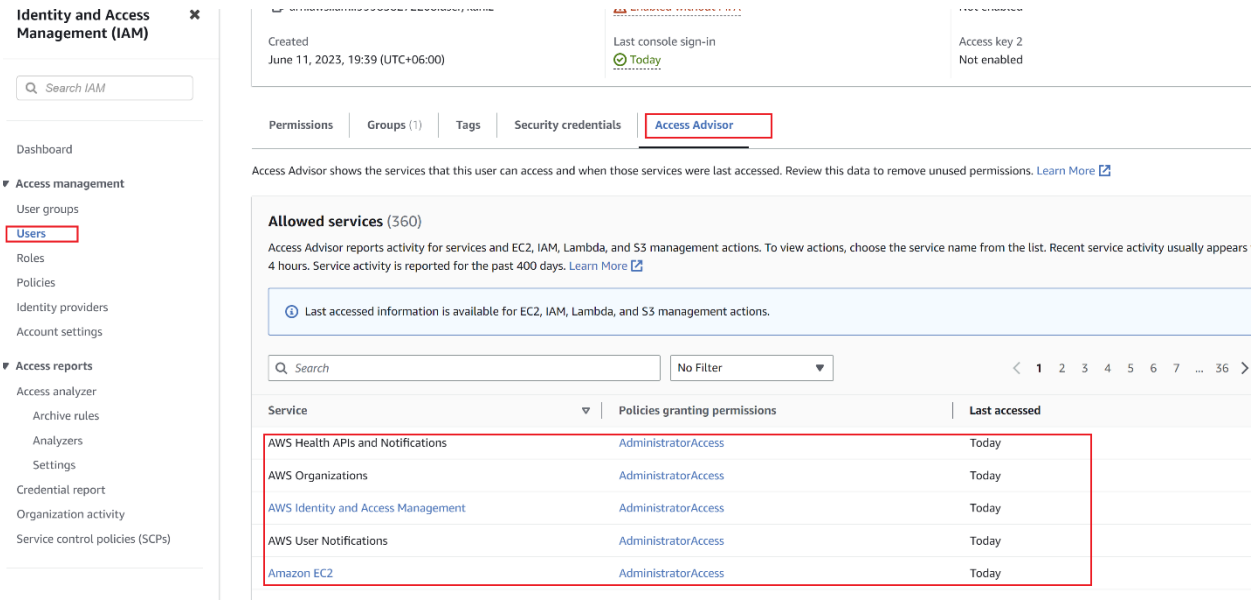
Credential's report



Now download the file and open and xls file and import that file and its look like below

A	B	C	D	E	F	G
user	arn	user_creation_time	password_enabled	password_last_used	password_last_changed	password_next_rotation
<root_account>	arn:aws:iam::123456789:root	4/18/2023 19:40	not_supported	2023-11-03T04:18:49+00:00	not_supported	not_supported
rajiv	arn:aws:iam::123456789:user/rajiv	5/1/2023 21:59	true	2023-05-01T16:02:22+00:00	2023-05-01T16:01:51+00:00	N/A
rajiv1	arn:aws:iam::123456789:user/rajiv1	6/20/2023 12:37	true	no_information	2023-06-20T06:37:06+00:00	N/A
rajiv2	arn:aws:iam::123456789:user/rajiv2	11/3/2023 10:26	true	2023-11-03T04:28:48+00:00	2023-11-03T04:26:46+00:00	N/A
sid1	arn:aws:iam::123456789:user/sid1	11/3/2023 11:15	true	no_information	2023-11-03T05:15:56+00:00	N/A

IAM Access Advisor



IAM Guidelines & Best Practices

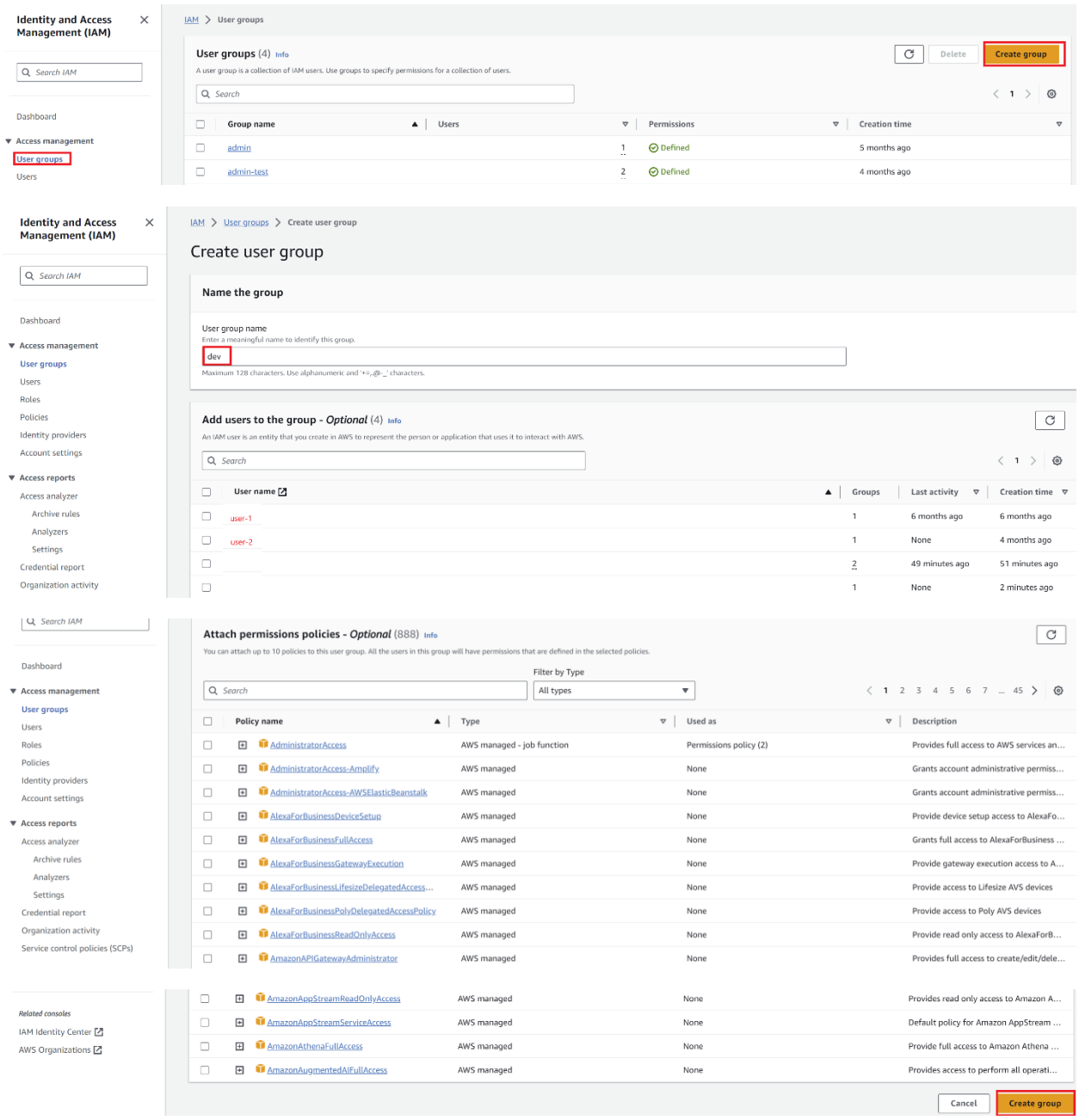
- Don't use the root account except for AWS account setup
- One physical user = One AWS user
- Assign users to groups and assign permissions to groups
- Create a strong password policy
- Use and enforce the use of Multi Factor Authentication (MFA)
- Create and use Roles for giving permissions to AWS services
- Use Access Keys for Programmatic Access (CLI / SDK)
- Audit permissions of your account using IAM Credentials Report & IAM Access Advisor

IAM Section – Summary

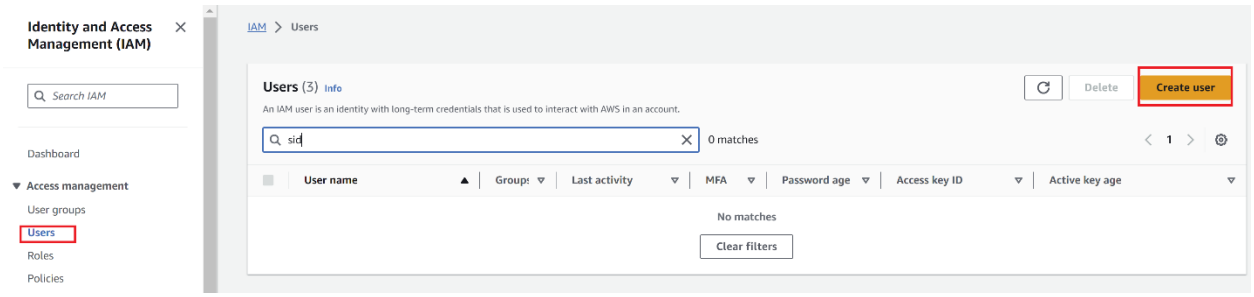
- Users: mapped to a physical user, has a password for AWS Console
- Groups: contains users only
- Policies: JSON document that outlines permissions for users or groups
- Roles: for EC2 instances or AWS services
- Security: MFA + Password Policy
- AWS CLI: manage your AWS services using the command-line
- AWS SDK: manage your AWS services using a programming language
- Access Keys: access AWS using the CLI or SDK
- Audit: IAM Credential Reports & IAM Access Advisor

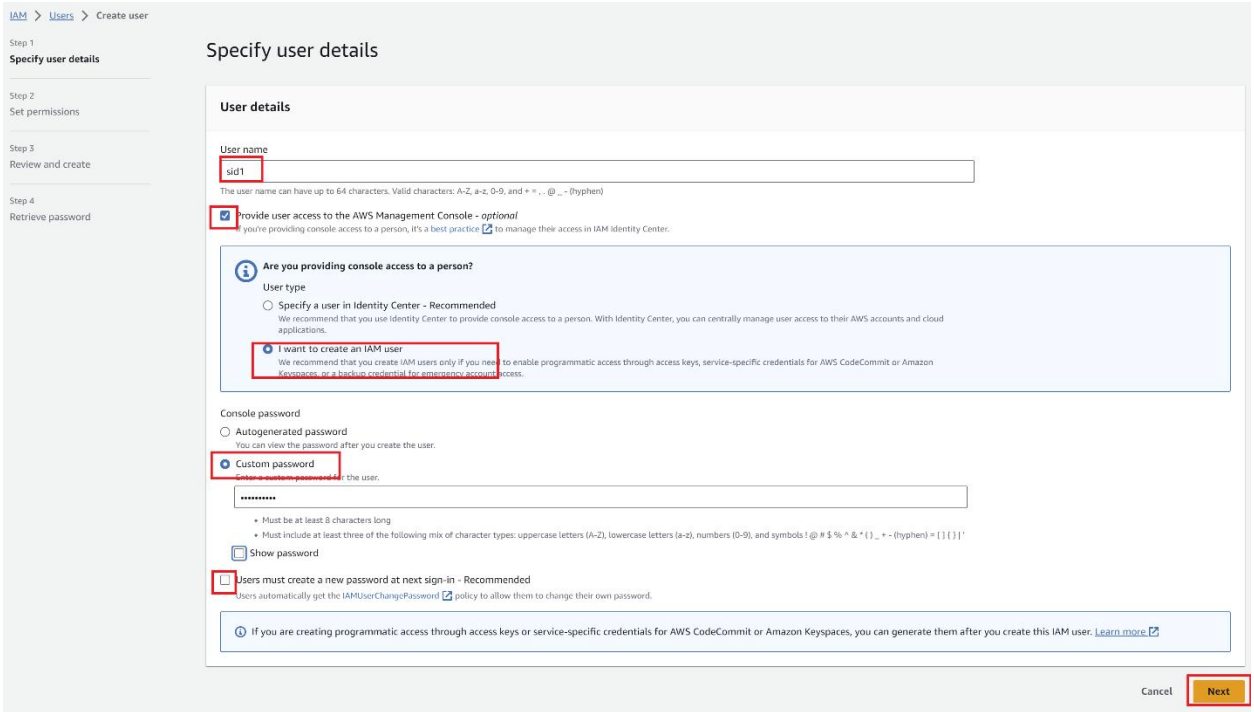
LAB: IAM -create user, group, policies

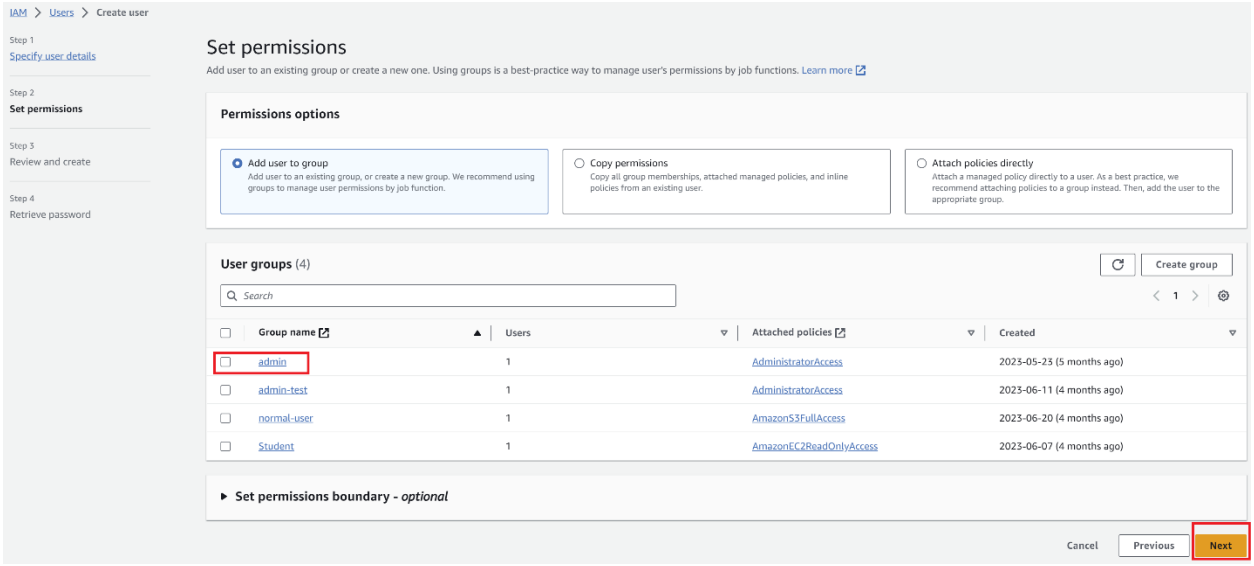
Create a group:

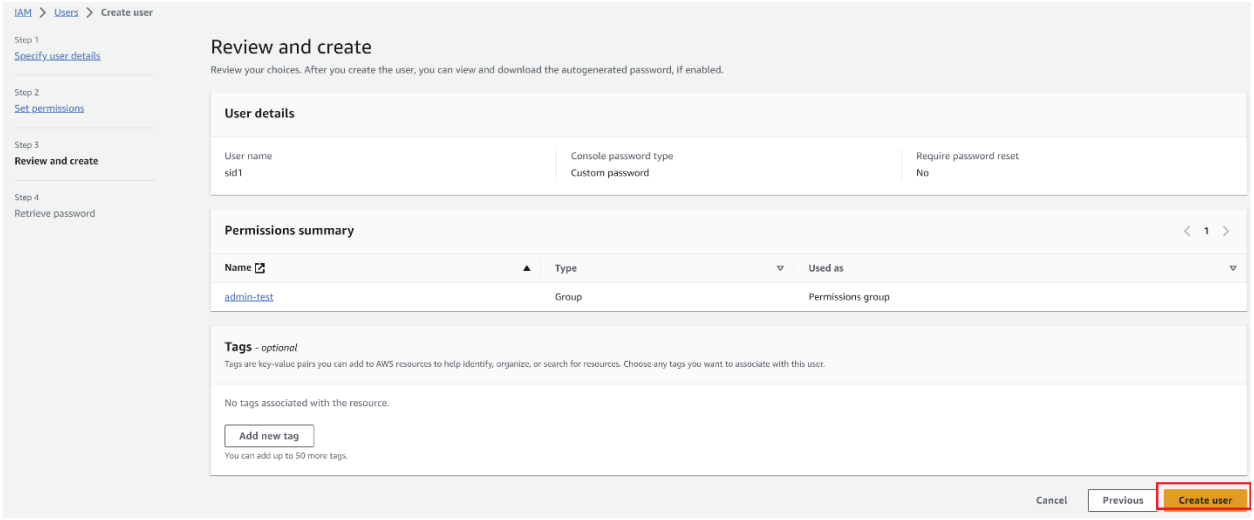


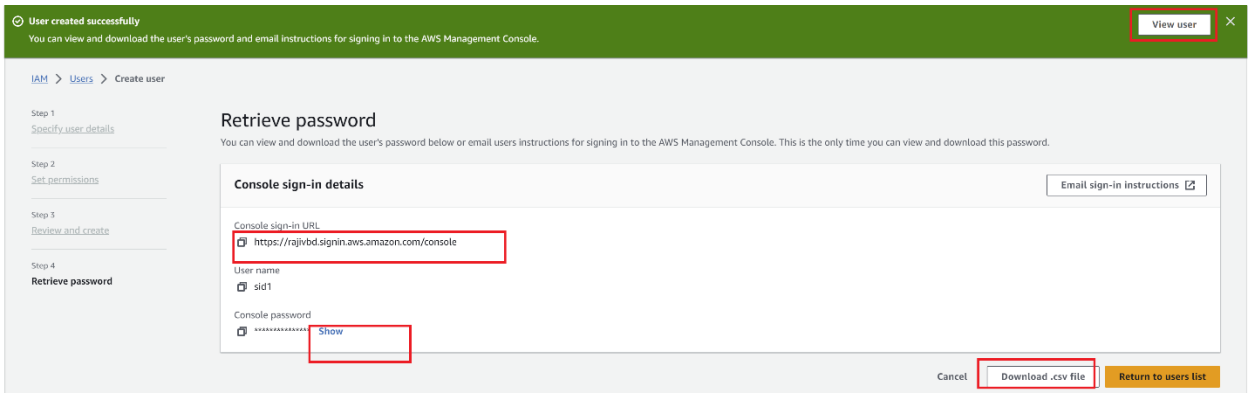
Create a user:



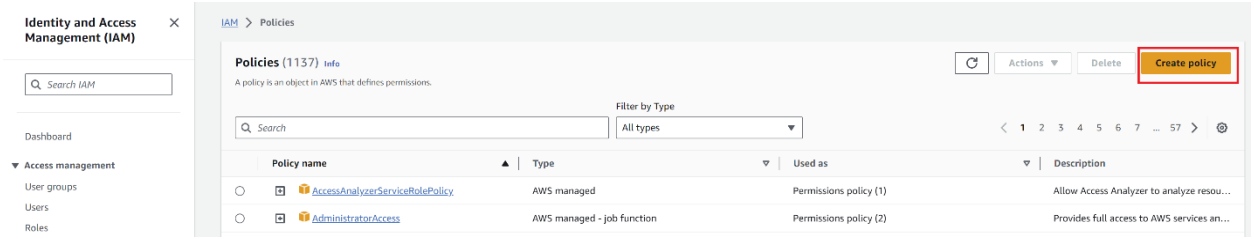

IAM > Users > Create user
Step 1
Specify user details
Step 2
Set permissions
Step 3
Review and create
Step 4
Retrieve password
Specify user details
User details
User name
sid1
The user name can have up to 64 characters. Valid characters: A-Z, a-z, 0-9, and + = , . @ _ - (hyphen)
Provide user access to 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.
Are you providing console access to a person?
User type
Specify a user in Identity Center - Recommended
We recommend that you use Identity Center to provide console access to a person. With Identity Center, you can centrally manage user access to their AWS accounts and cloud applications.
I want to create an IAM user
We recommend that you create IAM users only if you need to enable programmatic access through access keys, service-specific credentials for AWS CodeCommit or Amazon Keyspaces, or a backup credential for emergency account access.
Console password
Autogenerated password
You can view the password after you create the user.
Custom password
Enter a custom password for the user.
Must be at least 8 characters long
Must include at least three of the following mix of character types: uppercase letters (A-Z), lowercase letters (a-z), numbers (0-9), and symbols ! @ # $ % ^ & * ( ) _ + - (hyphen) = [ ] { } '
Show password
Users must create a new password at next sign-in - Recommended
Users automatically get the IAMUserChangePassword policy to allow them to change their own password.
If you are creating programmatic access through access keys or service-specific credentials for AWS CodeCommit or Amazon Keyspaces, you can generate them after you create this IAM user. Learn more
Cancel
Next

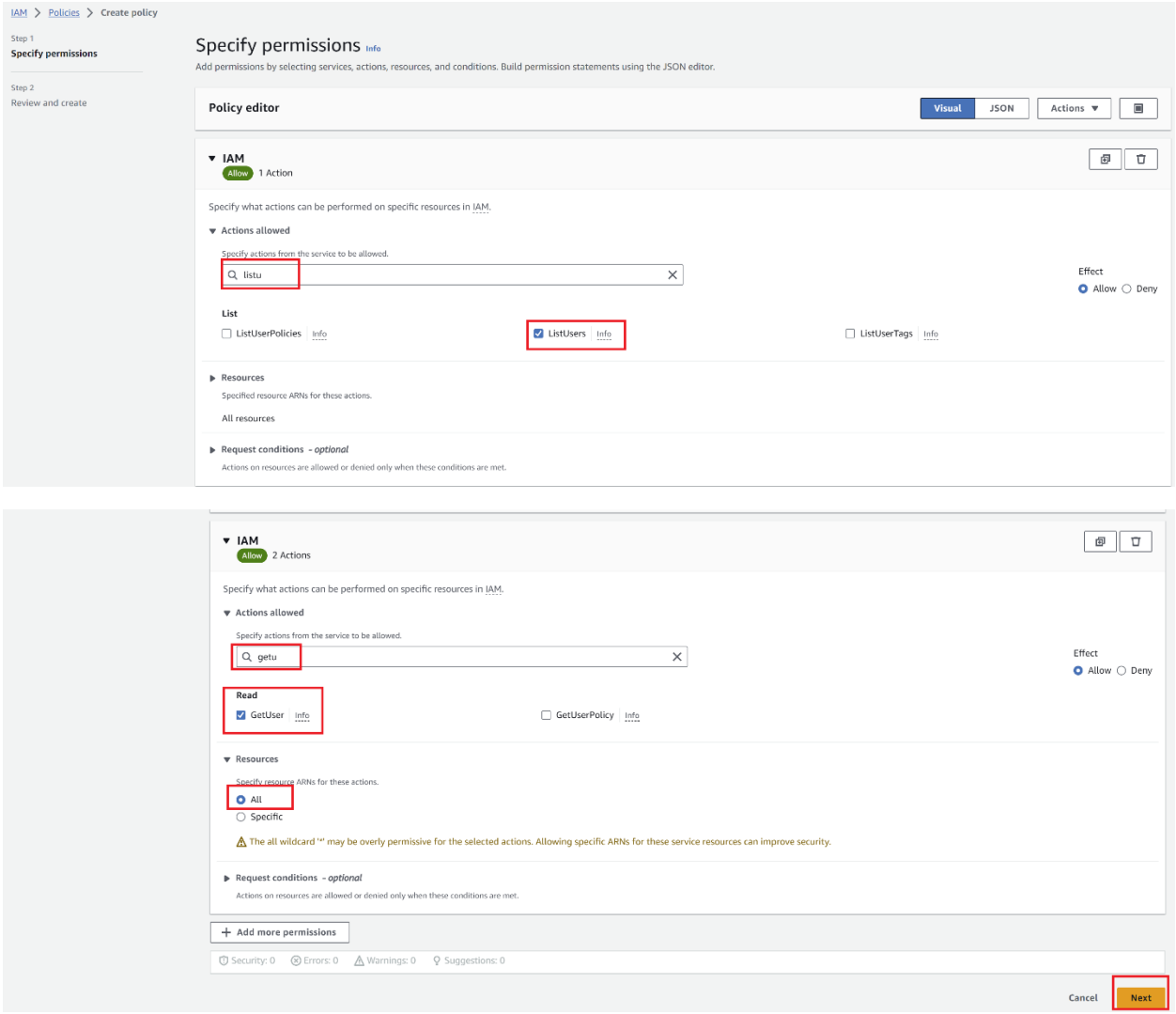

IAM > Users > Create user
Step 1
Specify user details
Step 2
Set permissions
Step 3
Review and create
Step 4
Retrieve password
Set permissions
Add user to an existing group or create a new one. Using groups is a best-practice way to manage user's permissions by job functions. Learn more
Permissions options
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.
User groups (4)
Create group
Search
1
Group name
Users
Attached policies
Created
admin
1
AdministratorAccess
2023-05-23 (5 months ago)
admin-test
1
AdministratorAccess
2023-06-11 (4 months ago)
normal-user
1
AmazonS3FullAccess
2023-06-20 (4 months ago)
Student
1
AmazonEC2ReadOnlyAccess
2023-06-07 (4 months ago)
Set permissions boundary - optional
Cancel
Previous
Next



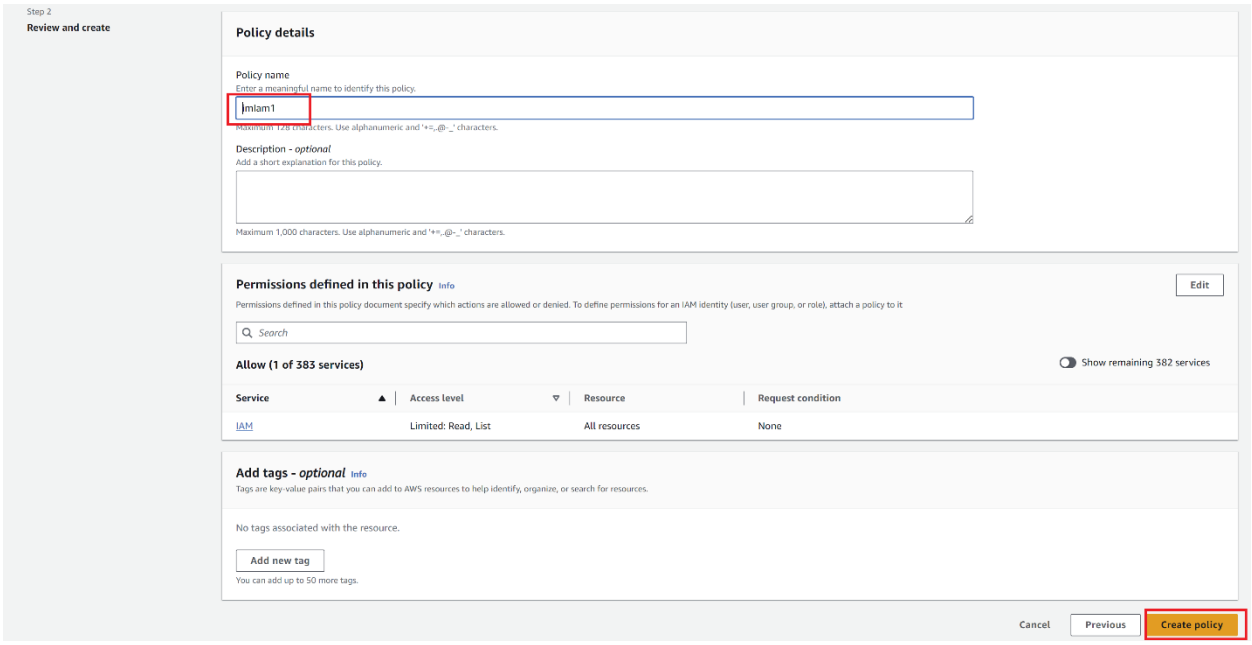


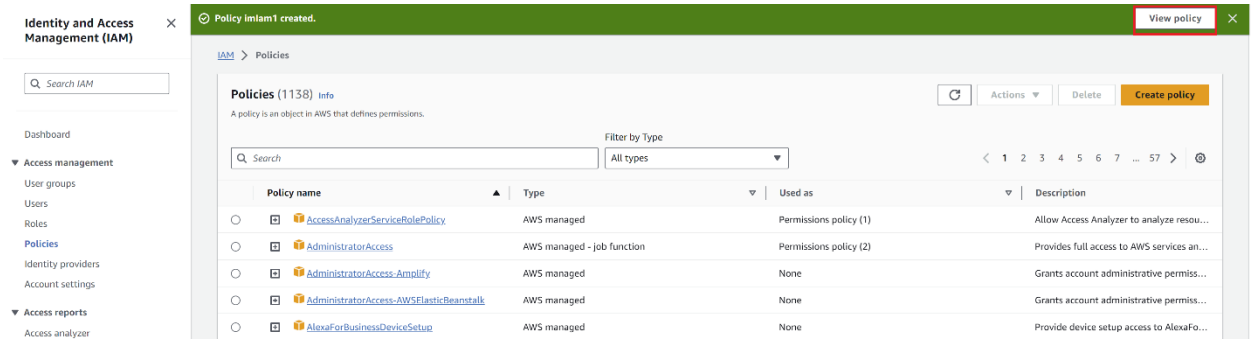
Create a policies:

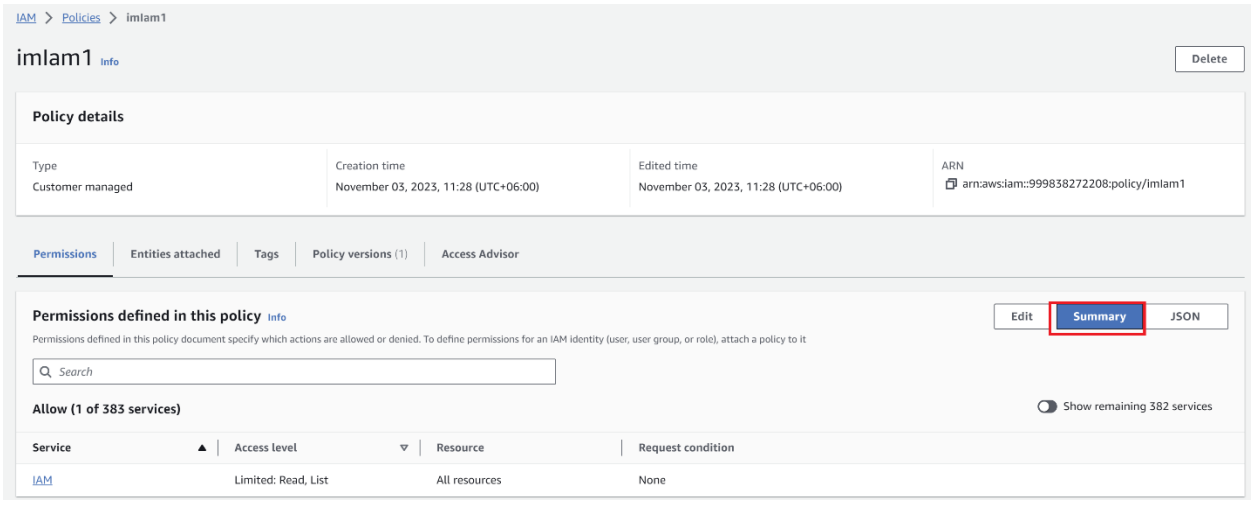


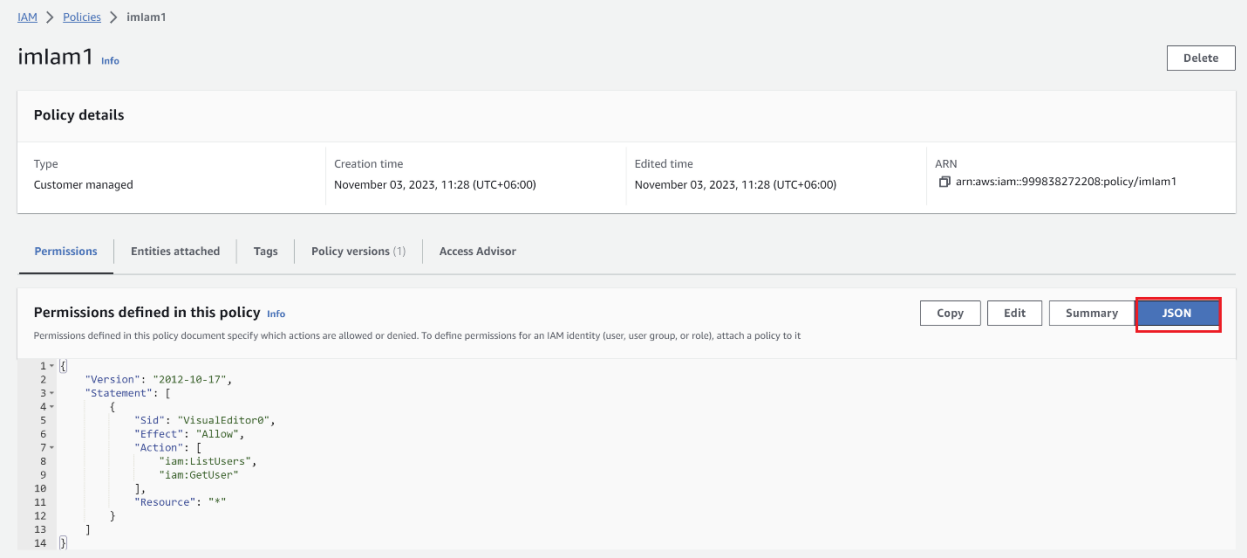

IAM > Policies > Create policy
Step 1
Specify permissions
Step 2
Review and create
Specify permissions Info
Add permissions by selecting services, actions, resources, and conditions. Build permission statements using the JSON editor.
Policy editor
Visual
JSON
Actions
IAM
Allow 1 Action
Specify what actions can be performed on specific resources in IAM.
Actions allowed
Specify actions from the service to be allowed.
listu
Effect
Allow
Deny
List
ListUserPolicies Info
ListUsers Info
ListUserTags Info
Resources
Specified resource ARNs for these actions.
All resources
Request conditions - optional
Actions on resources are allowed or denied only when these conditions are met.

IAM
Allow 2 Actions
Specify what actions can be performed on specific resources in IAM.
Actions allowed
Specify actions from the service to be allowed.
getu
Effect
Allow
Deny
Read
GetUser Info
GetUserPolicy Info
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

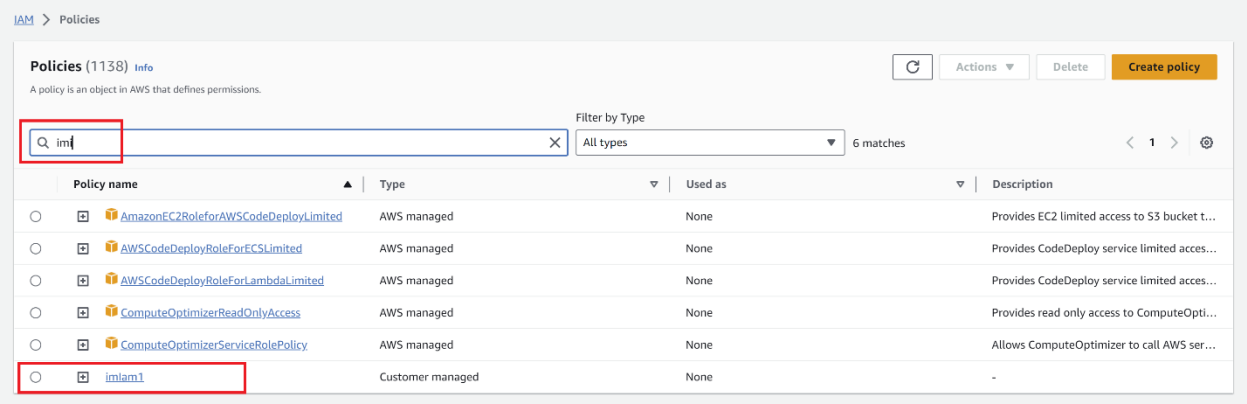

Step 2
Review and create
Policy details
Policy name
Enter a meaningful name to identify this policy.
imlam1
Maximum 128 characters. Use alphanumeric and '+=,.@-_' characters.
Description - optional
Add a short explanation for this policy.
Maximum 1,000 characters. Use alphanumeric and '+=,.@-_' characters.
Permissions defined in this policy Info
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 383 services)
Show remaining 382 services
Service
Access level
Resource
Request condition
IAM
Limited: Read, List
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


Identity and Access Management (IAM)
Search IAM
Dashboard
Access management
User groups
Users
Roles
Policies
Identity providers
Account settings
Access reports
Access analyzer
Policy imlam1 created.
View policy
IAM > Policies
Policies (1138) Info
A policy is an object in AWS that defines permissions.
Actions
Delete
Create policy
Filter by Type
Search
All types
1 2 3 4 5 6 7 ... 57
Policy name
Type
Used as
Description
AccessAnalyzerServiceRolePolicy
AWS managed
Permissions policy (1)
Allow Access Analyzer to analyze resou...
AdministratorAccess
AWS managed - job function
Permissions policy (2)
Provides full access to AWS services an...
AdministratorAccess-Amplify
AWS managed
None
Grants account administrative permiss...
AdministratorAccess-AWSElasticBeanstalk
AWS managed
None
Grants account administrative permiss...
AlexaForBusinessDeviceSetup
AWS managed
None
Provide device setup access to AlexaFo...

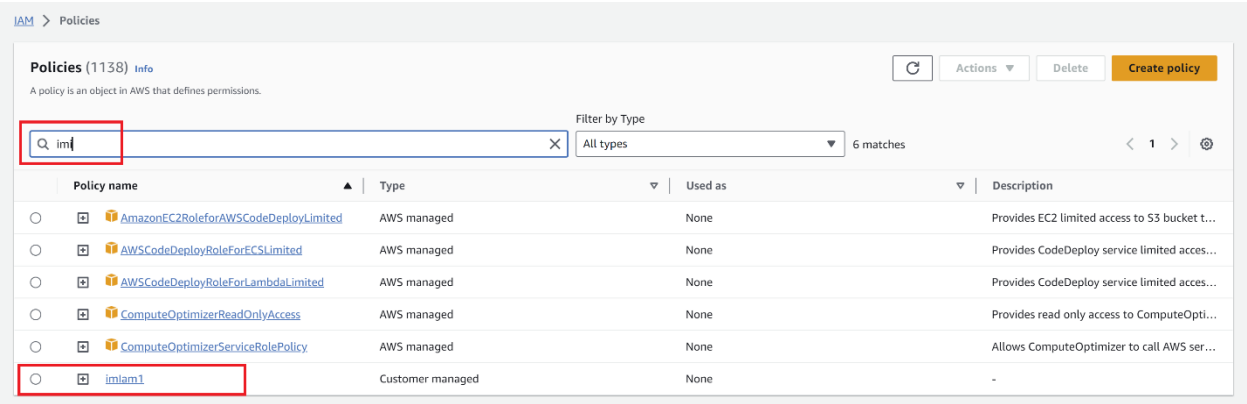

IAM > Policies > imlam1
imlam1 Info
Delete
Policy details
Type
Customer managed
Creation time
November 03, 2023, 11:28 (UTC+06:00)
Edited time
November 03, 2023, 11:28 (UTC+06:00)
ARN
arn:aws:iam::999838272208:policy/imlam1
Permissions
Entities attached
Tags
Policy versions (1)
Access Advisor
Permissions defined in this policy Info
Edit
Summary
JSON
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 383 services)
Show remaining 382 services
Service
Access level
Resource
Request condition
IAM
Limited: Read, List
All resources
None

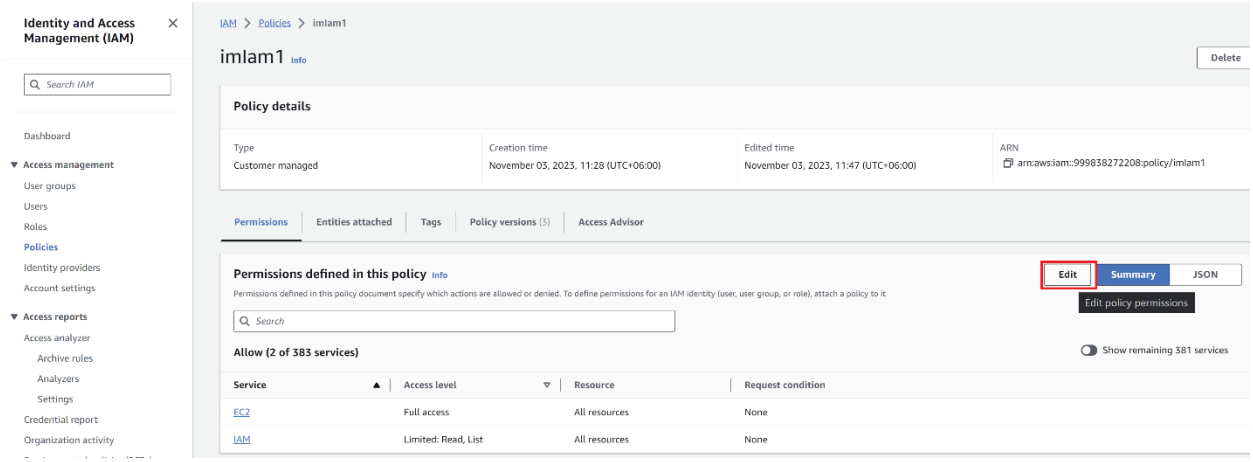


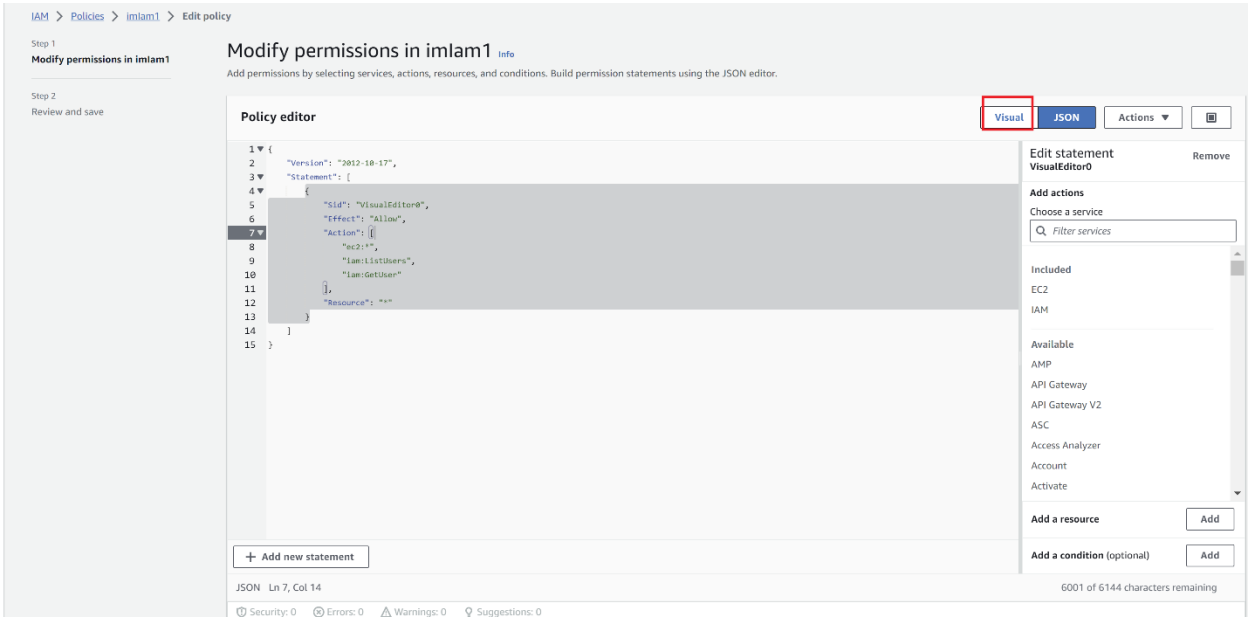
Search policies:

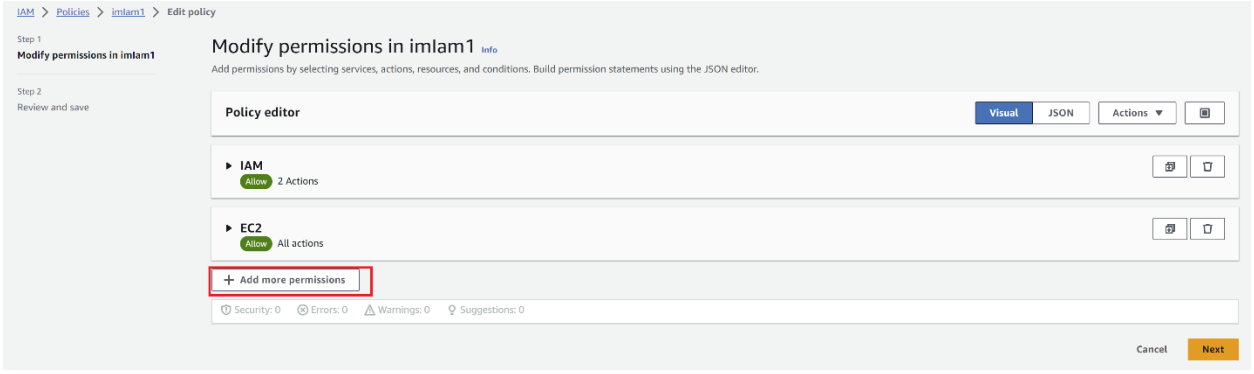


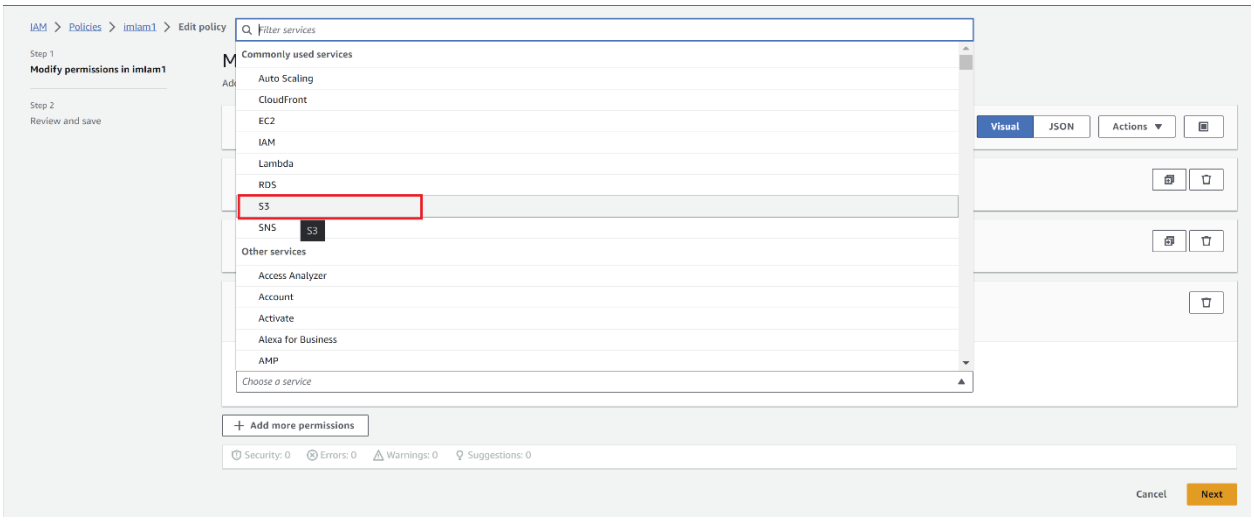
edit

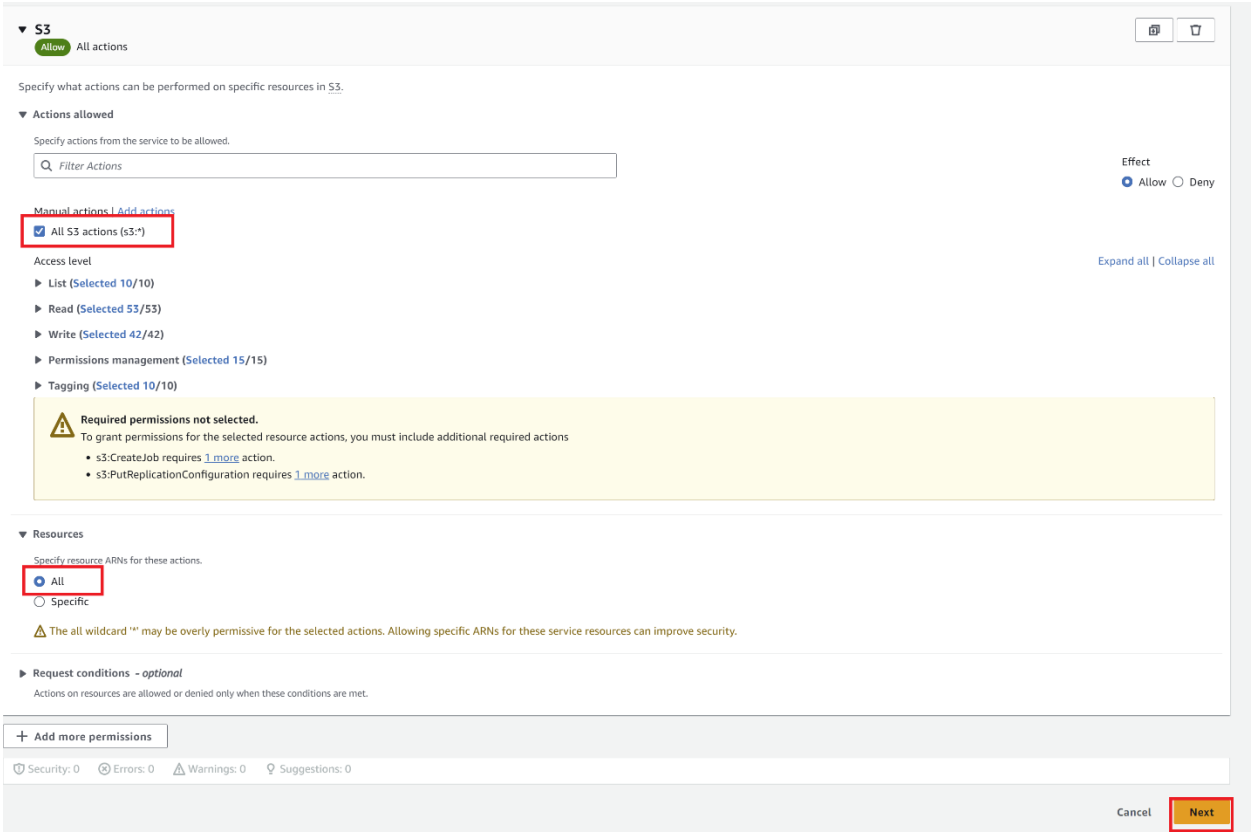


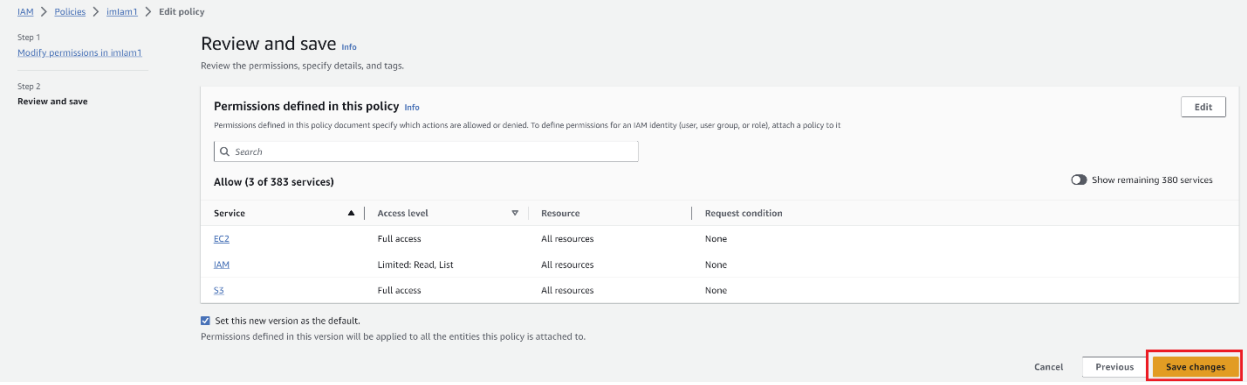

Identity and Access Management (IAM)
Search IAM
Dashboard
Access management
User groups
Users
Roles
Policies
Identity providers
Account settings
Access reports
Access analyzer
Archive rules
Analyzers
Settings
Credential report
Organization activity
IAM > Policies > imlam1
imlam1 Info
Delete
Policy details
Type
Customer managed
Creation time
November 03, 2023, 11:28 (UTC+06:00)
Edited time
November 03, 2023, 11:47 (UTC+06:00)
ARN
arn:aws:iam::999838272208:policy/imlam1
Permissions
Entities attached
Tags
Policy versions (3)
Access Advisor
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
Edit
Summary
JSON
Edit policy permissions
Search
Allow (2 of 383 services)
Show remaining 381 services
Service
Access level
Resource
Request condition
EC2
Full access
All resources
None
IAM
Limited: Read, List
All resources
None


IAM > Policies > imlam1 > Edit policy
Step 1
Modify permissions in imlam1
Step 2
Review and save
Modify permissions in imlam1 Info
Add permissions by selecting services, actions, resources, and conditions. Build permission statements using the JSON editor.
Policy editor
Visual
JSON
Actions
{
"Version": "2012-10-17",
"Statement": [
{
"Sid": "VisualEditor0",
"Effect": "Allow",
"Action": [
"ec2:*",
"iam:ListUsers",
"iam:GetUser"
],
"Resource": "*"
}
]
}
Edit statement
VisualEditor0
Remove
Add actions
Choose a service
Filter services
Included
EC2
IAM
Available
AMP
API Gateway
API Gateway V2
ASC
Access Analyzer
Account
Activate
Add a resource
Add
Add a condition (optional)
Add
Add new statement
JSON Ln 7, Col 14
6001 of 6144 characters remaining
Security: 0
Errors: 0
Warnings: 0
Suggestions: 0


IAM > Policies > imlam1 > Edit policy
Step 1
Modify permissions in imlam1
Step 2
Review and save
Modify permissions in imlam1 Info
Add permissions by selecting services, actions, resources, and conditions. Build permission statements using the JSON editor.
Policy editor
Visual
JSON
Actions
IAM
Allow
2 Actions
EC2
Allow
All actions
Add more permissions
Security: 0
Errors: 0
Warnings: 0
Suggestions: 0
Cancel
Next


IAM > Policies > imlam1 > Edit policy
Step 1
Modify permissions in imlam1
Step 2
Review and save
Filter services
Commonly used services
Auto Scaling
CloudFront
EC2
IAM
Lambda
RDS
S3
SNS
S3
Other services
Access Analyzer
Account
Activate
Alexa for Business
AMP
Choose a service
Visual
JSON
Actions
Add more permissions
Security: 0
Errors: 0
Warnings: 0
Suggestions: 0
Cancel
Next


S3
Allow
All actions
Specify what actions can be performed on specific resources in S3.
Actions allowed
Specify actions from the service to be allowed.
Filter Actions
Effect
Allow
Deny
Manual actions | Add actions
All S3 actions (s3:*)
Access level
Expand all | Collapse all
List (Selected 10/10)
Read (Selected 53/53)
Write (Selected 42/42)
Permissions management (Selected 15/15)
Tagging (Selected 10/10)
Required permissions not selected.
To grant permissions for the selected resource actions, you must include additional required actions
s3:CreateJob requires 1 more action.
s3:PutReplicationConfiguration requires 1 more action.
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



Delete the policies