

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

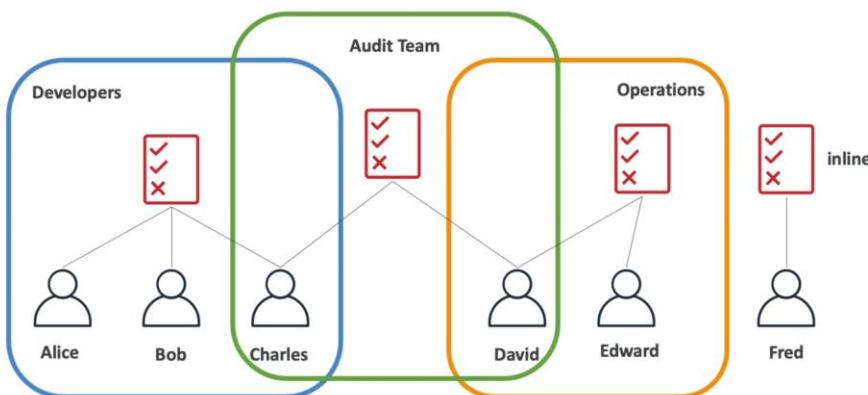


IAM: Permissions

- 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

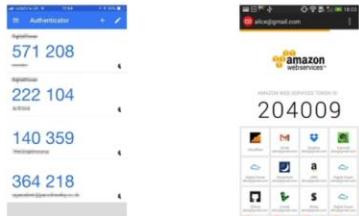
- Users have access to your account and can possibly change configurations or delete resources in your AWS account
- You want to protect your Root Accounts and IAM users
- MFA = password you know + security device you own



- 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

The screenshot shows the AWS IAM Account Settings page. The 'Account settings' section is open, specifically the 'Password policy' tab. The 'Edit' button is highlighted with a red box. The page displays the following information:

- Default password policy:** This AWS account uses the following default password policy:
 - Minimum length: 8 characters
 - Strength: Includes a minimum of three of the following mix of character types:
 - Uppercase
 - Lowercase
 - Numbers
 - Non-alphanumeric characters
- Other requirements:**
 - Never expire password
 - Must not be identical to your AWS account name or email address

Edit password policy Info

Password policy

IAM default

Apply default password requirements.

Custom

Apply customized password requirements.

Password minimum length.

Enforce a minimum length of characters.

8

characters

Needs to be between 6 and 128.

Password strength

- Require at least one uppercase letter from the Latin alphabet (A-Z)
- Require at least one lowercase letter from the Latin alphabet (a-z)
- Require at least one number
- Require at least one non-alphanumeric character (! @ # \$ % ^ & * () _ + - = [] { } | ')

Other requirements

- Turn on password expiration
- Password expiration requires administrator reset
- Allow users to change their own password
- Prevent password reuse

Cancel

Save changes

LAB: MFA Setting

The screenshot shows the AWS IAM Account Settings page. On the left, there's a navigation sidebar with various options like Dashboard, Access management, Account settings (which is currently selected), and others. The main content area displays the 'Account settings' page. It features a 'Password policy' section where it says 'This AWS account uses the following default password policy:'. Below this, it lists 'Password minimum length' as '8 characters' and 'Include a minimum of three of the following mix of character types:' followed by a list: '• Uppercase', '• Lowercase', '• Numbers', and '• Non-alphanumeric characters'. To the right of this, there's a 'Other requirements' section with two items: 'Never expire password' and 'Must not be identical to your AWS account name or email address'. At the bottom of the main content area, there's a 'Security Token Service (STS)' section with a note about temporary credentials. The right side of the screen shows a dark sidebar with account-related links: Account ID, Account, Organization, Service Quotas, Billing Dashboard, Security credentials (which is highlighted with a red box), and Settings. A 'Sign out' button is also visible in this sidebar.

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	User ARN
kaniz	arn:aws:iam::999838272208:user/kaniz
AWS account ID	Canonical user ID
999838272208	e302347e2defb13d0244f7d6bb54d6390f4a0894410f7b0a7fee87d617939273

AWS IAM credentials **AWS CodeCommit credentials** **Amazon Keyspaces credentials**

Console sign-in

Console sign-in link: <https://rajvbd.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)

Update console password

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



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

The screenshot shows the AWS IAM Security Credentials page. On the left, there's a sidebar with 'Search IAM' and sections for 'Dashboard', 'Access management' (User groups, Roles, Policies, Identity providers, Account settings), and 'Access reports'. The main content area is titled 'My security credentials (root user)' with an 'Info' link. It says 'The root user has access to all AWS resources in this account, and we recommend following best practices.' Below this is a 'Account details' section with fields for 'Account name' (rajuvsiddiqui), 'Email address' (rajuv1399@gmail.com), 'AWS account ID', and 'Canonical user ID'. There's also a 'Edit account name, email, and password' button. At the bottom of this section is an 'Actions' dropdown and a refresh icon. Below the main content is an 'AWS CloudShell' terminal window titled 'us-east-1'. The terminal shows a command-line session:

```
[cloudshell-user@ip-10-6-91-241 ~]$ aws --version
aws-cli/2.11.2 Python/3.11.3 Linux/4.14.255-314-253.539.amzn2.x86_64 exec-env/CloudShell exe/x86_64.amzn.2 prompt/off
[cloudshell-user@ip-10-6-91-241 ~]$ aws s3 ls
2023-05-04 06:31:08 rajuvsiddiqui
[cloudshell-user@ip-10-6-91-241 ~]$
```

At the bottom of the terminal window are links for 'CloudShell', 'Feedback', and 'Language'. The footer of the page includes copyright information: '© 2023, Amazon Web Services, Inc. or its affiliates.', 'Privacy', 'Terms', and 'Cookie preferences'.

LAB: install and configure AWS cli

create a access key for a user

The screenshot shows the AWS IAM console. On the left, the navigation pane is visible with sections like Identity and Access Management (IAM), Dashboard, Access management (User groups, Users, Roles), Access reports (Access analyzer, Archive rules, Analyzers, Settings, Credential report, Organization activity, Service control policies (SCPs)), and Related consoles (IAM Identity Center, AWS Organizations). The main area is titled 'Users (4) Info' and shows a table with four users. The user 'sid1' is selected and highlighted with a red box. The 'Security credentials' tab is selected, showing the 'Console sign-in' section with a 'Create access key' button (also highlighted with a red box) and the 'Multi-factor authentication (MFA)' section.

This is the first step of a wizard titled 'Access key best practices & alternatives'. It includes a 'Step 1' section with 'Access key best practices & alternatives' and a 'Step 2 - optional' section with 'Set description tag'. The main content area is titled 'Use case' and contains several radio button options: 'Command Line Interface (CLI)' (selected and highlighted with a red box), 'Local code', 'Application running on an AWS compute service', 'Third-party service', 'Application running outside AWS', and 'Other'. Below these is a warning box titled 'Alternatives recommended' with two items: 'Use AWS CloudShell, a browser-based CLI, to run commands.' and 'Use the AWS CLI V2 and enable authentication through a user in IAM Identity Center.'. At the bottom is a 'Confirmation' section with a checked checkbox 'I understand the above recommendation and want to proceed to create an access key.' and a 'Next' button (highlighted with a red box).

IAM > Users > sid1 > Create access key

Step 1
Access key best practices & alternatives

Step 2 - optional
Set description tag

Step 3
Retrieve access keys

Set description tag - optional Info

The description for this access key will be attached to this user as a tag and shown alongside the access key.

Description tag value
Describe the purpose of this access key and where it will be used. A good description will help you rotate this access key confidently later.

sid-1-acc

Maximum 256 characters. Allowed characters are letters, numbers, spaces representable in UTF-8, and: _ . : / = + - @

Cancel Previous **Create access key**

Access key created
This is the only time that the secret access key can be viewed or downloaded. You cannot recover it later. However, you can create a new access key any time.

IAM > Users > sid1 > Create access key

Step 1
Access key best practices & alternatives

Step 2 - optional
Set description tag

Step 3
Retrieve access keys

Retrieve access keys Info

Access key
If you lose or forget your secret access key, you cannot retrieve it. Instead, create a new access key and make the old key inactive.

Access key	Secret access key
AKIA6RSYBJLIP3RC2D5C	***** Show

Access key best practices

- Never store your access key in plain text, in a code repository, or in code.
- Disable or delete access key when no longer needed.
- Enable least-privilege permissions.
- Rotate access keys regularly.

For more details about managing access keys, see the [best practices for managing AWS access keys](#).

Download .csv file **Done**

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

```
rajiv@ubuntu-light:~$ aws configure
AWS Access Key ID [None]: AKIA6RSYBJLIP
AWS Secret Access Key [None]: gdEuDrVi/+Yd71ndzNqRDQv
Default region name [None]:
Default output format [None]:
rajiv@ubuntu-light:~$ aws s3 ls
2023-07-18 12:35:00 cf-templates-xr0fxg10ho12-us-east-1
2023-07-03 13:38:19 rajiv-athena-log
2023-09-16 05:36:37 rajiv-terraform-bucket
2023-07-03 11:17:57 rajiv20231
2023-07-11 08:23:23 s3-flow-log-2023
rajiv@ubuntu-light:~$
```

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

Identity and Access Management (IAM)

Console sign-in link
https://ravivbd.sigin.aws.amazon.com/console

Console password
Updated 1 hour ago (2023-11-03 11:15 GMT+6)

Last console sign-in
Never

Multi-factor authentication (MFA) (0)

No MFA devices. Assign an MFA device to improve the security of your AWS environment

Access keys (1)

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

Device type	Identifier	Certifications	Created on
No MFA devices. Assign an MFA device to improve the security of your AWS environment			
Assign MFA device			

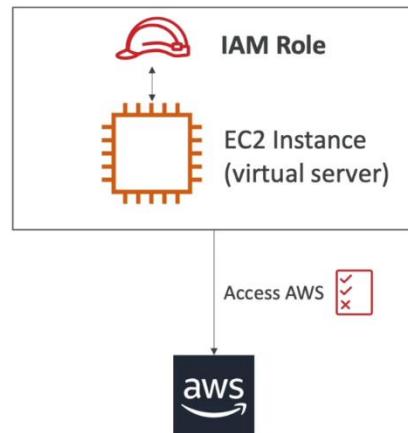
Create access key

Actions ▾

- Edit description
- Deactivate
- Activate
- Delete

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

Identity and Access Management (IAM)

Roles (25) Info

An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assumed by entities that you trust.

[Create role](#)

Role name	Trusted entities	Last activity
aws-ec2-spot-fleet-tagging-role	AWS Service: spotfleet	-
AWSServiceRoleForAccessAnalyzer	AWS Service: access-analyzer (Service-Linked)	185 days ago
AWSServiceRoleForAmazonElasticFileSystem	AWS Service: elasticfilesystem (Service-Linked)	26 days ago
AWSServiceRoleForAmazonSSM	AWS Service: ssm (Service-Linked)	49 minutes ago
AWSServiceRoleForAutoScaling	AWS Service: autoscaling (Service-Linked)	14 days ago
AWSServiceRoleForBackup	AWS Service: backup (Service-Linked)	15 hours ago
AWSServiceRoleForCloudWatchEvents	AWS Service: events (Service-Linked)	113 days ago
AWSServiceRoleForElasticLoadBalancing	AWS Service: elasticloadbalancing (Service-Linked)	9 days ago
AWSServiceRoleForGlobalAccelerator	AWS Service: globalaccelerator (Service-Linked)	-

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.

Use case
Allow an AWS service like EC2, Lambda, or others to perform actions in this account.

Service or use case

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.

Policy name	Type	Description
<input checked="" type="checkbox"/> s3read	AWS managed	Provides read only access to all buckets vi...
<input checked="" type="checkbox"/>  AmazonS3ReadOnlyAccess	AWS managed	Provides read only access to all buckets vi...

Set permissions boundary - optional

Cancel **Previous** **Next**

IAM > Roles > Create role

Step 1
Select trusted entity

Step 2
Add permissions

Step 3
Name, review, and create

Name, review, and create

Role details

Role name
Enter a meaningful name to identify this role.
s3readl_3nov

Maximum 64 characters. Use alphanumeric and '+-,@_,-' characters.

Description
Add a short explanation for this role.
Allows EC2 instances to call AWS services on your behalf.

Maximum 1000 characters. Use alphanumeric and '+-,@_,-' characters.

Step 1: Select trusted entities Edit

Trust policy

```

1 ~ {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Effect": "Allow",
6       "Action": "sts:AssumeRole"
7       "Principal": {
8         "Service": [
9           "ec2.amazonaws.com"
10          ]
11        }
12      }
13    }
14  ]
15 }
```

Step 2: Add permissions Edit

Step 2: Add permissions

Permissions policy summary

Policy name	Type	Attached as
AmazonS3ReadOnlyAccess	AWS managed	Permissions policy

Step 3: Add tags

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 role

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

EC2 Dashboard Instances (1/1) Info

EC2 Global View

Events

Instances

Instances

Instance Types

Launch Templates

Spot Requests

Savings Plans

Instances (1/1) Info

Find Instance by attribute or tag (case-sensitive)

Name = ec2-cloudwatch Clear filters

Name	Instance ID	Instance state	Instance type	Status check	Availability Z
ec2-cloudwatch	i-0c21d2a61153aafdb	Running	t2.micro	2/2 checks passed No alarms	+ us-east-1d

Actions Launch instances

Connect

View details

Manage instance state

Instance settings

Networking

Security Public

Change security groups

Get Windows password

Modify IAM role

EC2 > Instances > i-0c21d2a61153aafdb > Modify IAM role

Modify IAM role Info

Attach an IAM role to your instance.

Instance ID
i-0c21d2a61153aafdb (ec2-cloudwatch)

IAM role
Select an IAM role to attach to your instance or create a new role if you haven't created any. The role you select replaces any roles that are currently attached to your instance.
s3readl_3nov

Create new IAM role

Cancel Update IAM role

It looks like blow image

```
ec2-user@ip-172-31-36-243:~  
[ec2-user@ip-172-31-36-243 ~]$ aws s3 ls  
An error occurred (AccessDenied) when calling the ListBuckets operation: Access Denied  
[ec2-user@ip-172-31-36-243 ~]$ aws s3 ls  
2023-07-18 12:35:00 cf-templates-xr0fxg10ho12-us-east-1  
2023-07-03 13:38:19 rajiv-athena-log  
2023-09-16 05:36:37 rajiv-terraform-bucket  
2023-07-03 11:17:57 rajiv20231  
2023-07-11 08:23:23 s3-flow-log-2023  
[ec2-user@ip-172-31-36-243 ~]$
```

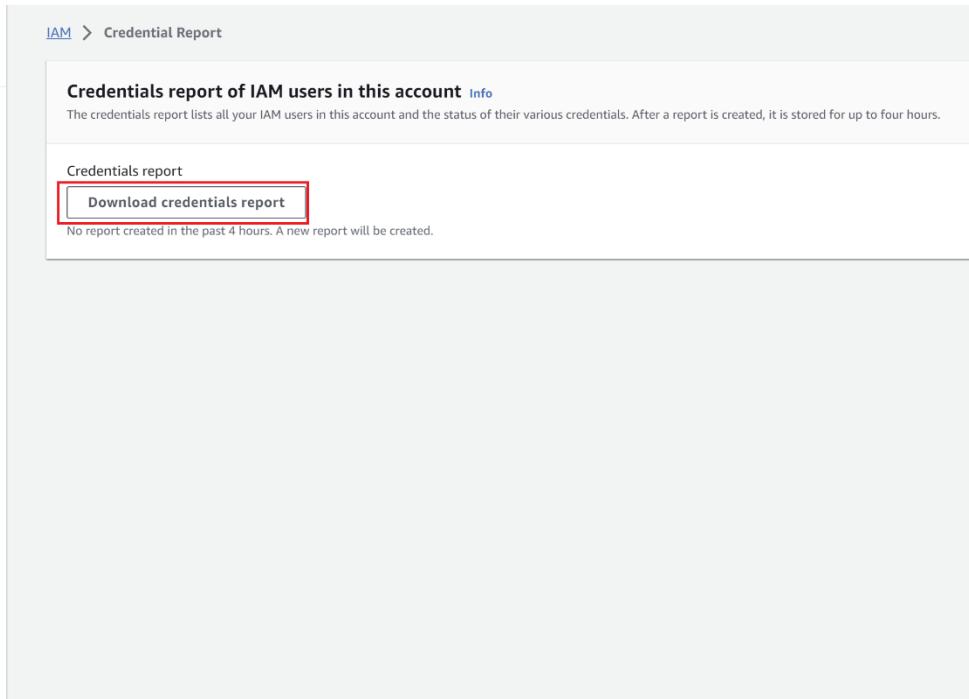
We can also configure it in this ec2 instance by running aws configure command but its not secure because your credentials will save it in ec2 instance

```
ec2-user@ip-172-31-36-243:~/aws  
[ec2-user@ip-172-31-36-243 ~]$ ls -la  
total 20  
drwx----- 4 ec2-user ec2-user 121 Oct 30 11:59 .  
drwxr-xr-x 3 root root 22 Oct 26 11:48 ..  
-rw----- 1 ec2-user ec2-user 341 Oct 30 12:00 .bash_history  
-rw-r--r-- 1 ec2-user ec2-user 18 Jan 28 2023 .bash_logout  
-rw-r--r-- 1 ec2-user ec2-user 141 Jan 28 2023 .bash_profile  
-rw-r--r-- 1 ec2-user ec2-user 492 Jan 28 2023 .bashrc  
drwx----- 2 ec2-user ec2-user 59 Oct 27 14:03 .ssh  
-rw----- 1 ec2-user ec2-user 708 Oct 30 11:59 .viminfo  
drwxr-xr-x 2 ec2-user ec2-user 6 Oct 27 14:22 rr  
[ec2-user@ip-172-31-36-243 ~]$ aws configure  
AWS Access Key ID [None]: AKIA6RSYBJLIDYI  
AWS Secret Access Key [None]: 195U28mJDvF  
Default region name [None]:  
Default output format [None]:  
[ec2-user@ip-172-31-36-243 ~]$ aws s3 ls  
2023-07-18 12:35:00 cf-templates-xr0fxg10ho12-us-east-1  
2023-07-03 13:38:19 rajiv-athena-log  
2023-09-16 05:36:37 rajiv-terraform-bucket  
2023-07-03 11:17:57 rajiv20231  
2023-07-11 08:23:23 s3-flow-log-2023  
[ec2-user@ip-172-31-36-243 ~]$ ls -la  
total 20  
drwx----- 5 ec2-user ec2-user 133 Nov 3 07:13 .  
drwxr-xr-x 3 root root 22 Oct 26 11:48 ..  
drwxr-xr-x 2 ec2-user ec2-user 39 Nov 3 07:13 .aws  
-rw----- 1 ec2-user ec2-user 341 Oct 30 12:00 .bash_history  
-rw-r--r-- 1 ec2-user ec2-user 18 Jan 28 2023 .bash_logout  
-rw-r--r-- 1 ec2-user ec2-user 141 Jan 28 2023 .bash_profile  
-rw-r--r-- 1 ec2-user ec2-user 492 Jan 28 2023 .bashrc  
drwx----- 2 ec2-user ec2-user 59 Oct 27 14:03 .ssh  
-rw----- 1 ec2-user ec2-user 708 Oct 30 11:59 .viminfo  
drwxr-xr-x 2 ec2-user ec2-user 6 Oct 27 14:22 rr  
[ec2-user@ip-172-31-36-243 ~]$ cd .aws/  
[ec2-user@ip-172-31-36-243 .aws]$ ll  
total 8  
-rw----- 1 ec2-user ec2-user 10 Nov 3 07:13 config  
-rw----- 1 ec2-user ec2-user 116 Nov 3 07:13 credentials  
[ec2-user@ip-172-31-36-243 .aws]$ cat credentials  
[default]  
aws_access_key_id = AKIA6RSYBJLIDYI  
aws_secret_access_key = 195U28mJDvFtBKuoe4McZODF37F+i9rL  
[ec2-user@ip-172-31-36-243 .aws]$
```

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



The screenshot shows the AWS IAM console. On the left, there's a navigation sidebar with 'Identity and Access Management (IAM)' at the top, followed by a search bar and several menu items under 'Access management' and 'Access reports'. A red box highlights the 'Credential report' link under 'Access reports'. The main content area is titled 'Credentials report of IAM users in this account' and includes a note that no report has been created in the past 4 hours. A prominent red box highlights the 'Download credentials report' button.

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

The screenshot shows the IAM Access Advisor interface. On the left, a sidebar lists navigation options like Dashboard, User groups, Users (which is selected), Roles, Policies, Identity providers, Account settings, Access reports, Access analyzer, Archive rules, Analyzers, Settings, Credential report, Organization activity, and Service control policies (SCPs). The main area displays user information: Created (June 11, 2023, 19:39 (UTC+06:00)), Last console sign-in (Today), and Access key 2 (Not enabled). Below this, tabs include Permissions, Groups (1), Tags, Security credentials, and Access Advisor (which is selected). A note states: "Access Advisor shows the services that this user can access and when those services were last accessed. Review this data to remove unused permissions. Learn More". The "Allowed services (360)" section notes: "Access Advisor reports activity for services and EC2, IAM, Lambda, and S3 management actions. To view actions, choose the service name from the list. Recent service activity usually appears 4 hours. Service activity is reported for the past 400 days. Learn More". A message indicates: "Last accessed information is available for EC2, IAM, Lambda, and S3 management actions." A table lists services and their last accessed status:

Service	Policies granting permissions	Last accessed
AWS Health APIs and Notifications	AdministratorAccess	Today
AWS Organizations	AdministratorAccess	Today
AWS Identity and Access Management	AdministratorAccess	Today
AWS User Notifications	AdministratorAccess	Today
Amazon EC2	AdministratorAccess	Today

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:

Identity and Access Management (IAM)

User groups

Create user group

Attach permissions policies - Optional (888) Info

Create a user:

The first screenshot shows the 'User groups' page with two existing groups: 'admin' and 'admin-test'. A red box highlights the 'Create group' button.

The second screenshot shows the 'Create user group' wizard. In the 'Name the group' step, 'dev' is entered into the 'User group name' field. In the 'Add users to the group - Optional' step, 'user-1' and 'user-2' are selected from a list of users. A red box highlights the 'Create group' button.

The third screenshot shows the 'Attach permissions policies - Optional' step. It lists 888 available policies, including various AWS managed policies like 'AdministratorAccess', 'AmazonAppStreamReadOnlyAccess', and 'AmazonAthenaFullAccess'. A red box highlights the 'Create group' button at the bottom right.

Identity and Access Management (IAM)

Users

Create user

The screenshot shows the 'Create user' wizard. The 'User name' field contains 'sic'. A red box highlights the 'Create user' button.

IAM > Users > Create user

Step 1
Specify user details

Step 2
Set permissions

Step 3
Review and create

Step 4
Retrieve password

User details

User name 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 Kinesis Data Analytics, or temporary security 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)

<input type="checkbox"/>	Group name Edit	Users	Attached policies Edit	Created
<input type="checkbox"/>	admin Edit	1	AdministratorAccess	2023-05-23 (5 months ago)
<input type="checkbox"/>	admin-test Edit	1	AdministratorAccess	2023-06-11 (4 months ago)
<input type="checkbox"/>	normal-user Edit	1	AmazonS3FullAccess	2023-06-20 (4 months ago)
<input type="checkbox"/>	Student Edit	1	AmazonEC2ReadOnlyAccess	2023-06-07 (4 months ago)

Set permissions boundary - optional

Cancel **Previous** **Next**

IAM > Users > Create user

Step 1
Specify user details

Step 2
Set permissions

Step 3
Review and create

Step 4
Retrieve password

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 sid1	Console password type Custom password	Require password reset No

Permissions summary		
Name	Type	Used as
admin-test	Group	Permissions group

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**

⌚ User created successfully
You can view and download the user's password and email instructions for signing in to the AWS Management Console.

View user

IAM > Users > Create user

Step 1
Specify user details

Step 2
Set permissions

Step 3
Review and create

Step 4
Retrieve password

Retrieve password

You can view and download the user's password below or email users instructions for signing in to the AWS Management Console. This is the only time you can view and download this password.

Console sign-in details		Email sign-in instructions
Console sign-in URL https://rajivbd.sigin.aws.amazon.com/console		Email sign-in instructions
User name sid1		
Console password XXXXXXXXXXXXXX	Show	

Cancel Download .CSV file Return to users list

Create a policies:

Identity and Access Management (IAM)

Search IAM

Dashboard

Access management

- User groups
- Users
- Roles

IAM > Policies

Policies (1137) Info

A policy is an object in AWS that defines permissions.

Create policy

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...

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

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.

Q_ listu

List

ListUserPolicies Info

ListUsers Info

ListUserTags Info

Resources

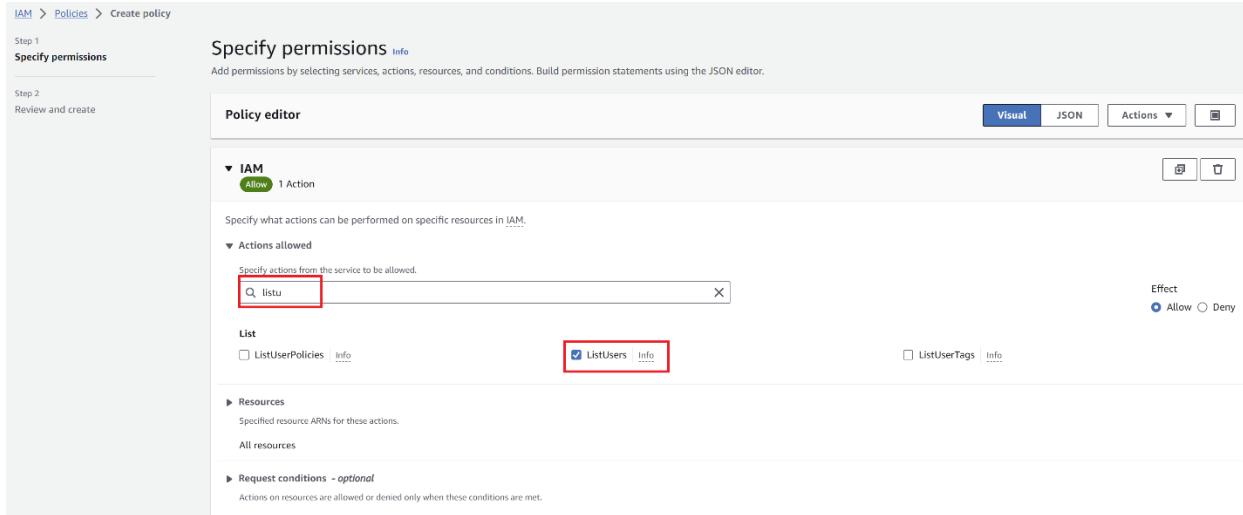
Specify resource ARNs for these actions.

All resources

Request conditions - optional

Actions on resources are allowed or denied only when these conditions are met.

Effect
 Allow Deny



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.

Q_getu

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.

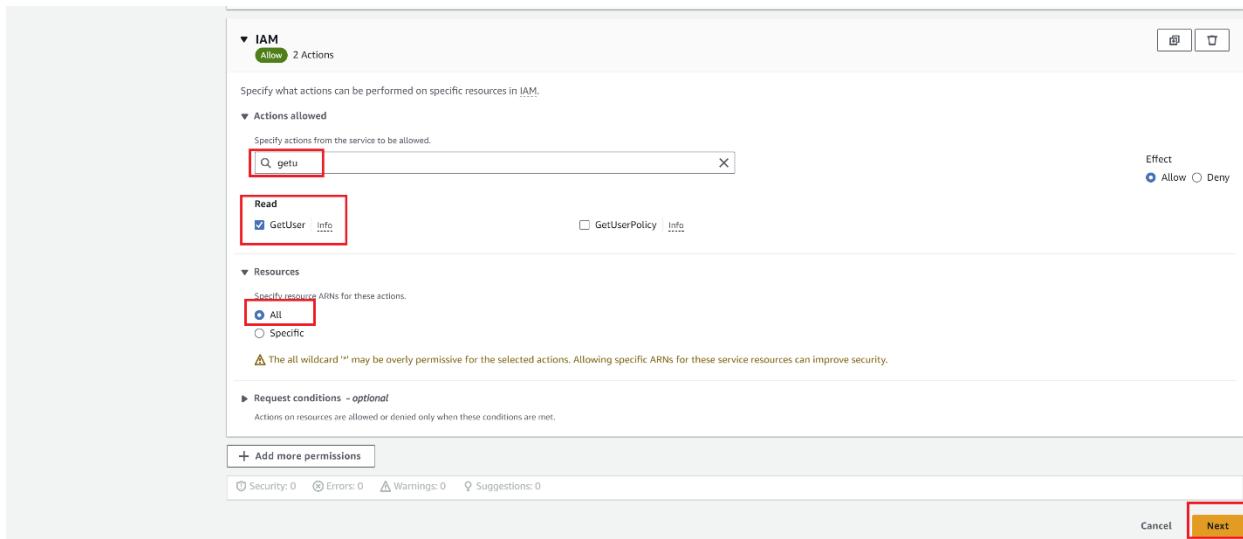
Request conditions - optional

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.
 Maximum 128 characters. Use alphanumeric and '+'-' characters.

Description - optional
Add a short explanation for this policy.

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.

Show remaining 382 services

Allow (1 of 383 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.

You can add up to 50 more tags.

Identity and Access Management (IAM)

Policy imiam1 created.

Policies (1138) Info
A policy is an object in AWS that defines permissions.

Filter by Type

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 permis...
AdministratorAccess-AWSFlexibleBeastalk	AWS managed	None	Grants account administrative permiss...
AlexaForBusinessDeviceSetup	AWS managed	None	Provide device setup access to AlexaFo...

IAM > Policies > imiam1

imiam1 Info

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 <input type="button" value="arn:aws:iam::999838272208:policy/imiam1"/>
--------------------------	---	---	---

Permissions **Entities attached** **Tags** **Policy versions (1)** **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.

Show remaining 382 services

Allow (1 of 383 services)

Service	Access level	Resource	Request condition
IAM	Limited: Read, List	All resources	None

IAM > Policies > imlam1

imlam1 Info

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

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.

```

1 "Version": "2012-10-17",
2 "Statement": [
3     {
4         "Sid": "VisualEditor0",
5         "Effect": "Allow",
6         "Action": [
7             "iam>ListUsers",
8             "iam:GetUser"
9         ],
10        "Resource": "*"
11    }
12 ]
13
14

```

Copy **Edit** **Summary** **JSON** JSON

Search policies:

IAM > Policies

Policies (1138) Info

A policy is an object in AWS that defines permissions.

Filter by Type Q. im All types 6 matches

Policy name	Type	Used as	Description
AmazonEC2RoleforAWSCodeDeployLimited	AWS managed	None	Provides EC2 limited access to S3 bucket t...
AWSCodeDeployRoleForECSLimited	AWS managed	None	Provides CodeDeploy service limited acces...
AWSCodeDeployRoleForLambdaLimited	AWS managed	None	Provides CodeDeploy service limited acces...
ComputeOptimizerReadOnlyAccess	AWS managed	None	Provides read only access to ComputeOpti...
ComputeOptimizerServiceRolePolicy	AWS managed	None	Allows ComputeOptimizer to call AWS ser...
imlam1	Customer managed	None	-

edit

IAM > Policies

Policies (1138) Info

A policy is an object in AWS that defines permissions.

Filter by Type Q. im All types 6 matches

Policy name	Type	Used as	Description
AmazonEC2RoleforAWSCodeDeployLimited	AWS managed	None	Provides EC2 limited access to S3 bucket t...
AWSCodeDeployRoleForECSLimited	AWS managed	None	Provides CodeDeploy service limited acces...
AWSCodeDeployRoleForLambdaLimited	AWS managed	None	Provides CodeDeploy service limited acces...
ComputeOptimizerReadOnlyAccess	AWS managed	None	Provides read only access to ComputeOpti...
ComputeOptimizerServiceRolePolicy	AWS managed	None	Allows ComputeOptimizer to call AWS ser...
imlam1	Customer managed	None	-

Identity and Access Management (IAM)

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::999836272208:policy/imlam1
--------------------------	---	---	--

Permissions [Entities attached](#) [Tags](#) [Policy versions \(5\)](#) [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 policy permissions](#) [Show remaining 581 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

Policy editor

```

1 ▼ {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Sid": "VisualEditor9",
6       "Effect": "Allow",
7       "Action": [
8         "ec2:DescribeInstances",
9         "iam:ListUsers",
10        "iam:GetUser"
11      ],
12      "Resource": "*"
13    }
14  ]
15 }

```

[+ Add new statement](#)

JSON Ln 7, Col 14

Security: 0 Errors: 0 Warnings: 0 Suggestions: 0 6001 of 6144 characters remaining

[Edit statement VisualEditor9 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](#)

IAM > Policies > imlam1 > Edit policy

Step 1
Modify permissions in imlam1

Step 2
Review and save

Policy editor

- ▶ IAM [Allow](#) 2 Actions
- ▶ EC2 [Allow](#) All actions

[+ Add more permissions](#)

Security: 0 Errors: 0 Warnings: 0 Suggestions: 0

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

IAM > Policies > **imiam1** > Edit policy

Step 1
Modify permissions in imiam1

Step 2
Review and save

S3 S3

Commonly used services: Auto Scaling, CloudFront, EC2, IAM, Lambda, RDS, S3, SNS.

Other services: Access Analyzer, Account, Activate, Alexa for Business, AMP.

Visual **JSON** **Actions** **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.

Effect: Allow Deny

Manual actions | [Add actions](#)

All 53 actions (s3:*)

Access level:

- ▶ 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

Warning: 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.

Cancel **Next**

IAM > Policies > imlam1 > Edit policy

Step 1
Modify permissions in imlam1

Step 2
Review and save

Review and save Info

Review the permissions, specify details, and tags.

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

Set this new version as the default.

Permissions defined in this version will be applied to all the entities this policy is attached to.

Service	Access level	Resource	Request condition
EC2	Full access	All resources	None
IAM	Limited: Read, List	All resources	None
S3	Full access	All resources	None

Show remaining 380 services

Cancel Previous Save changes

Delete the policies

Identity and Access Management (IAM)

Search IAM

Dashboard

Access management

- User groups
- Users
- Roles
- Policies**
- Identity providers
- Account settings

Access reports

Access analyzer

IAM > Policies

Polices (1/1138) Info

A policy is an object in AWS that defines permissions.

imi

Actions Delete Create policy

Policy name	Type	Used as	Description
AmazonEC2RoleforAWSCodeDeployLimited	AWS managed	None	Provides EC2 limited access to S3 buck...
AWSCodeDeployRoleForECSLimited	AWS managed	None	Provides CodeDeploy service limited a...
AWSCodeDeployRoleForLambdaLimited	AWS managed	None	Provides CodeDeploy service limited a...
ComputeOptimizerReadOnlyAccess	AWS managed	None	Provides read only access to Compute...
ComputeOptimizerServiceRolePolicy	AWS managed	None	Allows ComputeOptimizer to call AWS ...
imlam1	Customer managed	None	-

Delete imlam1?

Delete **imlam1** policy and all its versions permanently?

This action cannot be undone.

To confirm deletion, enter the policy name in the text input field.

Cancel Delete