

26-02-2025

TASK ON IAM

1) Create one IAM user and assign ec2, s3 full access role:

The image shows two screenshots from the AWS IAM console. The top screenshot is the 'Create user' page, and the bottom screenshot is the 's3-ec2-User' details page.

Top Screenshot: Create user

User details

Field	Value
User name	s3-ec2-User
Console password type	None
Require password reset	No

Permissions summary

Name	Type	Used as
AmazonEC2FullAccess	AWS managed	Permissions policy
AmazonS3FullAccess	AWS managed	Permissions policy

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](#)

Bottom Screenshot: s3-ec2-User

Summary

Field	Value
ARN	arn:aws:iam::971422718404:user/s3-ec2-User
Console access	Disabled
Access key 1	Create access key
Created	February 27, 2025, 12:11 (UTC+05:30)
Last console sign-in	-

Permissions policies (2/2)

Permissions are defined by policies attached to the user directly or through groups.

Filter by Type: All types

Policy name	Type	Attached via
<input checked="" type="checkbox"/> AmazonEC2FullAccess	AWS managed	Directly
<input checked="" type="checkbox"/> AmazonS3FullAccess	AWS managed	Directly

2) Create one Group in IAM and Assign Read access for ec2:

The screenshot shows the AWS IAM console interface for a newly created group named 'New-group1'. The left sidebar contains the 'Identity and Access Management (IAM)' menu with options like Dashboard, Access management, Access reports, and Access Analyzer. The main content area is titled 'New-group1' and includes a 'Summary' section with details: User group name (New-group1), Creation time (February 27, 2025, 12:21 (UTC+05:30)), and ARN (arn:aws:iam::971422718404:group/New-group1). Below the summary, the 'Permissions' tab is active, showing 'Permissions policies (1)' with a table listing the attached policy 'AmazonEC2ReadOnlyAccess' of type 'AWS managed'.

Policy name	Type	Attached entities
AmazonEC2ReadOnlyAccess	AWS managed	1

3) Create a new user with name DevOps and add to the group created in task2:

The first screenshot shows the 'Set permissions' step in the AWS IAM console. It includes a progress bar with three steps: 'Specify user details', 'Set permissions' (current), and 'Review and create'. Under 'Permissions options', the 'Add user to group' option is selected. Below, the 'User groups (1/1)' table shows 'New-group1' is selected. The 'Set permissions boundary - optional' section is also visible.

Group name	Users	Attached policies	Created
New-group1	0	AmazonEC2ReadOnlyAccess	2025-02-27 (3 minutes ago)

The second screenshot shows the 'Users (2)' page in the AWS IAM console. It lists two users: 'Devops' and 's3-ec2-User'. The 'Devops' user is associated with the 'New-group1' group.

User name	Path	Group	Last activity	MFA	Password age	Console last sign-in
Devops	/	1	-	-	-	-
s3-ec2-User	/	0	-	-	-	-

The screenshot shows the AWS IAM console interface. On the left is a navigation sidebar with sections for 'Identity and Access Management (IAM)', 'Access management', and 'Access reports'. The main content area is titled 'New-group1' and includes a 'Summary' section with details like 'User group name', 'Creation time', and 'ARN'. Below this, there are tabs for 'Users', 'Permissions', and 'Access Advisor'. The 'Users' tab is active, showing 'Users in this group (1)' with a table containing one user named 'Devops'.

4) Write a bash script to create a IAM user with VPC full access:

```
#!/bin/bash

# Variables
IAM_USER="VPC-Admin-User"
POLICY_ARN="arn:aws:iam::aws:policy/AmazonVPCFullAccess"

# Check if AWS CLI is installed
if ! command -v aws &> /dev/null; then
    echo "AWS CLI not found. Please install AWS CLI first."
    exit 1
fi

# Create IAM User
echo "Creating IAM user: $IAM_USER..."
aws iam create-user --user-name $IAM_USER

# Attach VPC Full Access Policy
echo "Attaching VPC Full Access policy to $IAM_USER..."
aws iam attach-user-policy --user-name $IAM_USER --policy-arn $POLICY_ARN

# Generate and Save Access Keys
echo "Creating access keys for $IAM_USER..."
aws iam create-access-key --user-name $IAM_USER > ${IAM_USER}_credentials.json

echo "IAM user '$IAM_USER' created successfully with VPC Full Access!"
echo "Access keys saved in ${IAM_USER}_credentials.json"
```

```
Amazon Linux 2
AL2 End of Life is 2026-06-30.
A newer version of Amazon Linux is available!
Amazon Linux 2023, GA and supported until 2028-03-15.
https://aws.amazon.com/linux/amazon-linux-2023/

[ec2-user@ip-192-168-0-26 ~]$ sudo -i
[root@ip-192-168-0-26 ~]# vi iam_vpc.bash
[root@ip-192-168-0-26 ~]# chmod 777 iam_vpc.bash
[root@ip-192-168-0-26 ~]# ./iam_vpc.bash
Creating IAM user: VPC-Admin-User...
Unable to locate credentials. You can configure credentials by running "aws configure".
Attaching VPC Full Access policy to VPC-Admin-User...
Unable to locate credentials. You can configure credentials by running "aws configure".
Creating access keys for VPC-Admin-User...
Unable to locate credentials. You can configure credentials by running "aws configure".
IAM user 'VPC-Admin-User' created successfully with VPC Full Access!
Access keys saved in VPC-Admin-User_credentials.json
[root@ip-192-168-0-26 ~]# cat iam_vpc.bash
#!/bin/bash
```

```
[root@ip-192-168-0-26 ~]# aws configure
AWS Access Key ID [None]: AKIA6ELKOLHCFDRJTV7W
AWS Secret Access Key [None]: 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
Default region name [None]: ap-south-1
Default output format [None]: json
[root@ip-192-168-0-26 ~]# ./iam_vpc.bash
Creating IAM user: VPC-Admin-User...
{
  "User": {
    "UserName": "VPC-Admin-User",
    "Path": "/",
    "CreateDate": "2025-02-27T08:05:45Z",
    "UserId": "AIDA6ELKOLHCFY7DTXTOL",
    "Arn": "arn:aws:iam::971422718404:user/VPC-Admin-User"
  }
}
Attaching VPC Full Access policy to VPC-Admin-User...
Creating access keys for VPC-Admin-User...
IAM user 'VPC-Admin-User' created successfully with VPC Full Access!
Access keys saved in VPC-Admin-User credentials.json
```

The screenshot shows the AWS IAM console interface for the user 'VPC-Admin-User'. The left sidebar contains navigation links for Identity and Access Management (IAM), Access management, and Access reports. The main content area displays the user's details and permissions.

Summary:

- ARN: `arn:aws:iam::971422718404:user/VPC-Admin-User`
- Console access: Disabled
- Created: February 27, 2025, 13:35 (UTC+05:30)
- Last console sign-in: -
- Access key 1: AKIA6ELKOLHCFJLCDWWY - Active (Never used. Created today.)
- Access key 2: [Create access key]

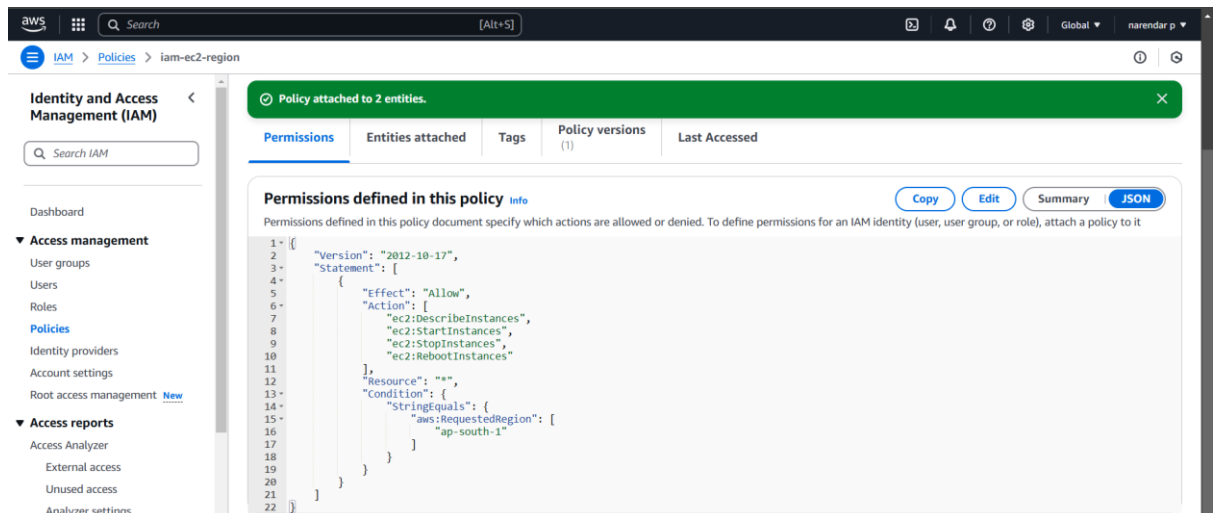
Permissions policies (1):

Permissions are defined by policies attached to the user directly or through groups.

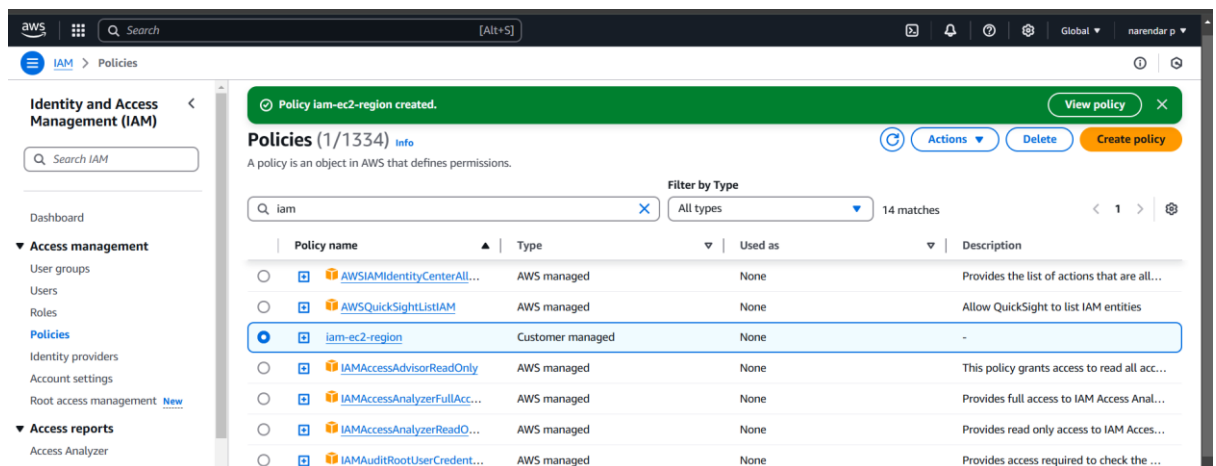
Policy name	Type	Attached via
AmazonVPCFullAccess	AWS managed	Directly

5) Create a IAM policy to access ec2 for a specific user in specific regions only:

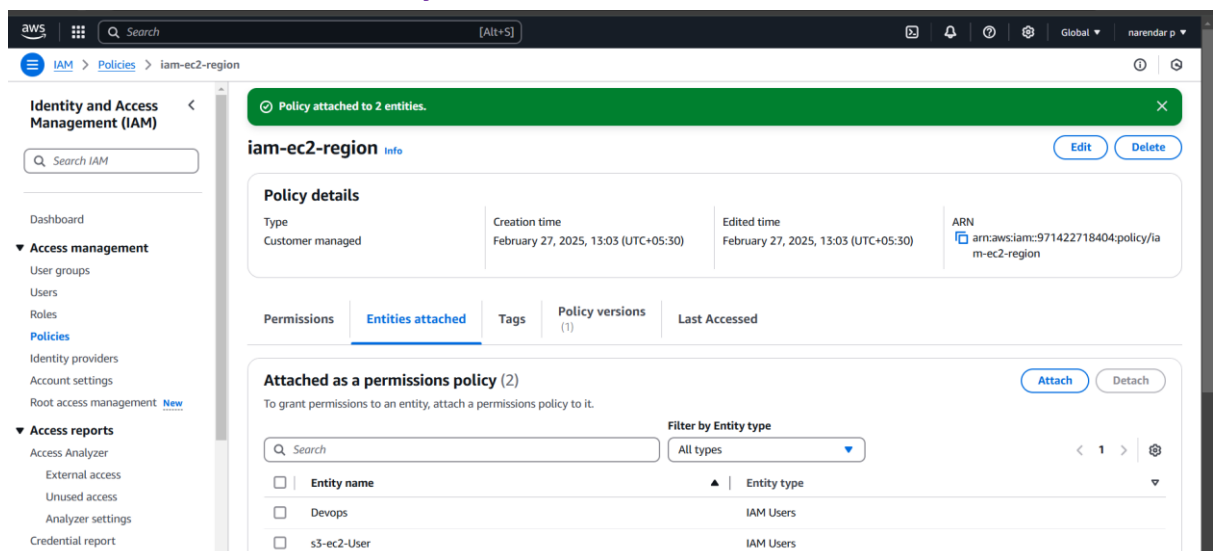
Json entry: To create IAM policy to access ec2 in specific region:



Created Policy:



Attached users to the Policy:



6) We have two accounts Account A and Account B, Account A user should access s3 bucket in Account B.

(Collaborate with team member and execute this. Mostly asked in every interview):

User details

User name: s3-ec2-User | Console password type: None | Require password reset: No

Permissions summary

Name	Type	Used as
AmazonEC2FullAccess	AWS managed	Permissions policy
AmazonS3FullAccess	AWS managed	Permissions policy

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](#)

→ Created a s3 bucket from my account (Account B)

General purpose buckets | Directory buckets

General purpose buckets (1/2) [Info](#) [All AWS Regions](#) [Refresh](#) [Copy ARN](#) [Empty](#) [Delete](#) [Create bucket](#)

Buckets are containers for data stored in S3.

	Name	AWS Region	IAM Access Analyzer	Creation date
<input type="radio"/>	my-vpc-flowlog-s3-0509	US East (N. Virginia) us-east-1	View analyzer for us-east-1	February 25, 2025, 17:55:19 (UTC+05:30)
<input checked="" type="radio"/>	revan-s3bucket-001	US East (N. Virginia) us-east-1	View analyzer for us-east-1	February 27, 2025, 14:37:30 (UTC+05:30)

→ Attached a json policy to the s3 bucket, adding account A id:

Bucket policy [Edit](#) [Delete](#)

The bucket policy, written in JSON, provides access to the objects stored in the bucket. Bucket policies don't apply to objects owned by other accounts. [Learn more](#)

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": "*",
      "Action": "s3:*",
      "Resource": [
        "arn:aws:s3:::revan-s3bucket-001",
        "arn:aws:s3:::revan-s3bucket-001/*"
      ],
      "Condition": {
        "StringEquals": {
          "aws:PrincipalAccount": "183631328120"
        }
      }
    }
  ]
}
```

[Copy](#)

→ Created a s3 bucket access policy from account A

S3accessfrom=AtoB [Info](#) [Edit](#) [Delete](#)

Policy details

Type Customer managed	Creation time February 27, 2025, 17:32 (UTC+05:30)	Edited time February 27, 2025, 17:49 (UTC+05:30)	ARN arn:aws:iam::183631328120:policy/S3accessfrom=AtoB
--------------------------	---	---	---

[Permissions](#)
[Entities attached](#)
[Tags](#)
[Policy versions \(2\)](#)
[Last Accessed](#)

Permissions defined in this policy [Info](#)
[Copy](#)
[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

```

1 {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Effect": "Allow",
6       "Action": "s3:GetObject",
7       "Resource": "arn:aws:s3:::revan-s3bucket-001/*"
8     }
9   ]
10 }
```

→ Attached an IAM user (techi) to this policy (from account A)

S3accessfrom=AtoB [Info](#) [Edit](#) [Delete](#)

Policy details

Type Customer managed	Creation time February 27, 2025, 17:32 (UTC+05:30)	Edited time February 27, 2025, 17:49 (UTC+05:30)	ARN arn:aws:iam::183631328120:policy/S3accessfrom=AtoB
--------------------------	---	---	---

[Permissions](#)
[Entities attached](#)
[Tags](#)
[Policy versions \(2\)](#)
[Last Accessed](#)

Attached as a permissions policy (1/1)
[Attach](#)
[Detach](#)

To grant permissions to an entity, attach a permissions policy to it.

Filter by Entity type

All types

<input checked="" type="checkbox"/> Entity name	Entity type
<input checked="" type="checkbox"/> techi	IAM Users

→ Configured Account A account using aws cli

```

The config profile (AccountA-profile) could not be found
[root@ip-172-31-81-82 ~]# aws configure --profile AccountA-profile
AWS Access Key ID [None]: AKIASVQKHWN4KGZ4ILVI
AWS Secret Access Key [None]: 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
Default region name [None]: us-east-1
Default output format [None]: json
[root@ip-172-31-81-82 ~]# aws sts get-caller-identity --profile AccountA-profile

{
  "Account": "183631328120",
  "UserId": "183631328120",
  "Arn": "arn:aws:iam::183631328120:root"
}
```

→ s3 bucket objects in account B

Objects (1/1) [Refresh](#) [Copy S3 URI](#) [Copy URL](#) [Download](#) [Open](#) [Delete](#) [Actions](#) [Create folder](#) [Upload](#)

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

< 1 > [Settings](#)

<input checked="" type="checkbox"/>	Name	Type	Last modified	Size	Storage class
<input checked="" type="checkbox"/>	challenge.bash	bash	February 27, 2025, 18:07:55 (UTC+05:30)	194.0 B	Standard

➔ Account A was able to access the s3 bucket objects of account B, using below command:

```
[root@ip-172-31-81-82 ~]#
[root@ip-172-31-81-82 ~]# aws s3 ls s3://revan-s3bucket-001 --profile AccountA-profile
[root@ip-172-31-81-82 ~]# aws s3 ls s3://revan-s3bucket-001 --profile AccountA-profile
2025-02-27 12:37:55      194 challenge.bash
[root@ip-172-31-81-82 ~]# aws s3 ls s3://bucket060918 --profile AccountA-profile
2025-02-27 12:57:26    221037 challenge-3.docx
[root@ip-172-31-81-82 ~]# exit
```