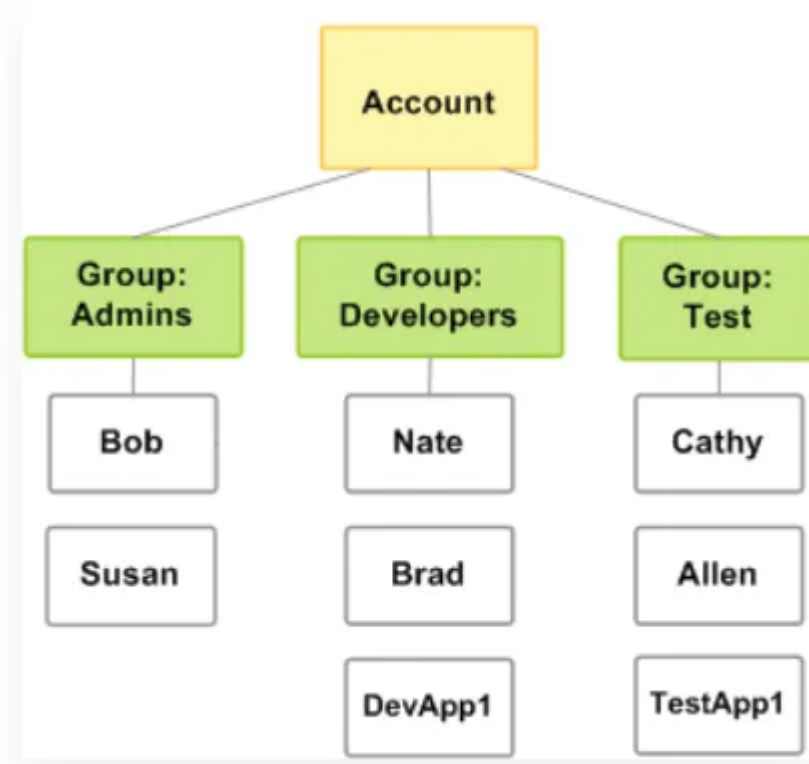


AWS Identity and Access Management (IAM), is a web service that helps the user securely control access to AWS resources. IAM allows you to manage, create users, policies, permissions, user groups, roles.

IAM user: It is an entity that you create in AWS to represent a person or application that interacts with AWS resources.

IAM account root user: When you first create an AWS account , you begin with single sign in identity. This is AWS account root user and it has complete access to all the AWS services and resources in the account. You cannot control the permissions of the AWS root user credentials It is a best practice that you should not use root user for daily interactions.

IAM user group: An IAM user group is a collection of IAM users. User groups let you specify permissions for multiple users, which can make it easier to manage the permissions for those users. for e.g You have a user group called Testers and the type of permissions testers typically need. All the users in the user group have all the permissions of Tester group.



IAM Roles: IAM roles are used to delegate access to AWS resources. Roles provide temporary access.

Lets discuss about Authentication and Authorization.

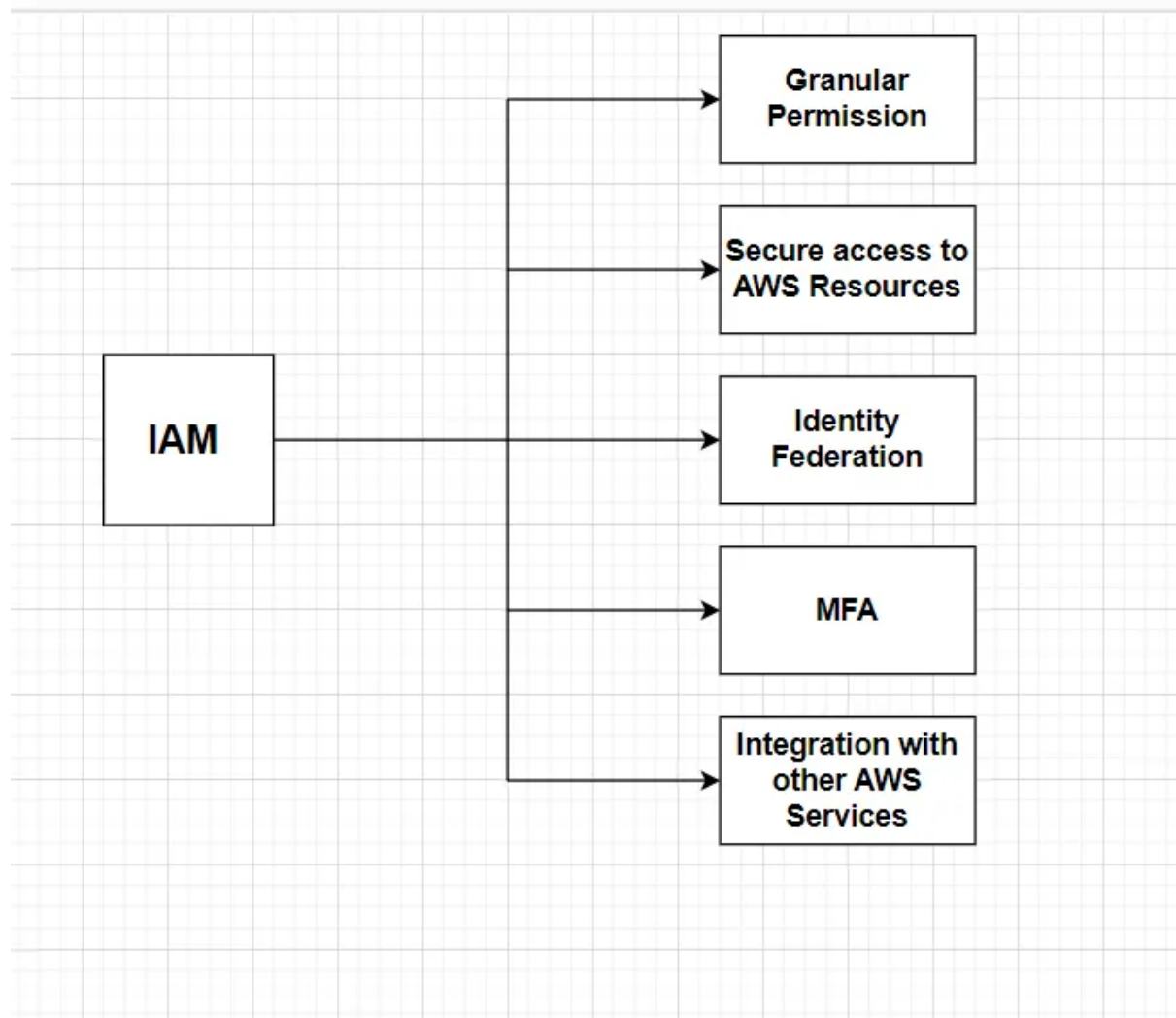
Authentication

Use IAM to configure authentication, which is the first step because it controls who can access AWS resources. IAM is used for user authentication, and applications and other AWS services also use it for access.

Authorization

IAM is used to configure authorization based on the user. Authorization determines which resources users can access and what they can do to or with those resources. Authorization is defined through the use of policies. A policy is an object in AWS that, when associated with an identity or resource, defines their permissions.

Following are the features of IAM:



Lets do some hands-on on IAM

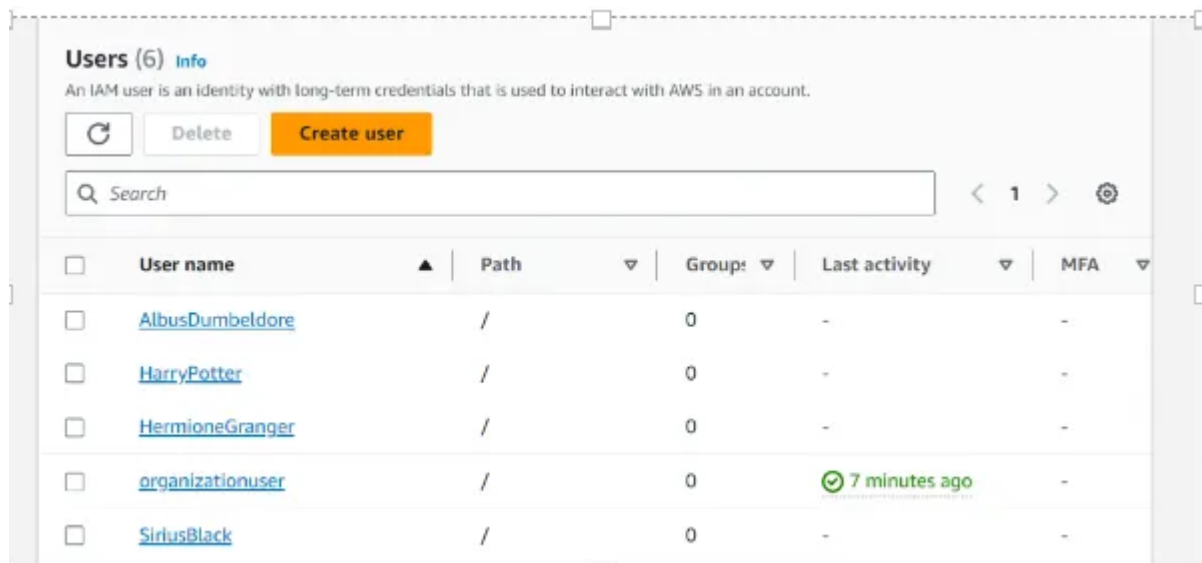
Here we will perform three task :

1. Create IAM User
2. Create IAM User Groups and add users
3. Create EC2 Policy

Task 1. Create IAM User

In the Add User page, fill in the User Details section as follows:

- i) User name: Enter HarryPotter (or the desired name for the user)
- ii) Check the Provide user access to the AWS Management Console — optional checkbox
- iii) Select Custom password under Console Password and Enter the desired password for the user
- iv) Uncheck the Users must create a new password at the next sign-in (recommended) checkbox.
- v) Click on the Next button.
- vi) In the Set permissions section, keep things as default. Click on the Next button.
- vii) Scroll down and Under Tags, Click on the Add new tag button:
 - Key: Enter *Dev-Team*
 - Value: Enter *Developers*
- viii) Create User



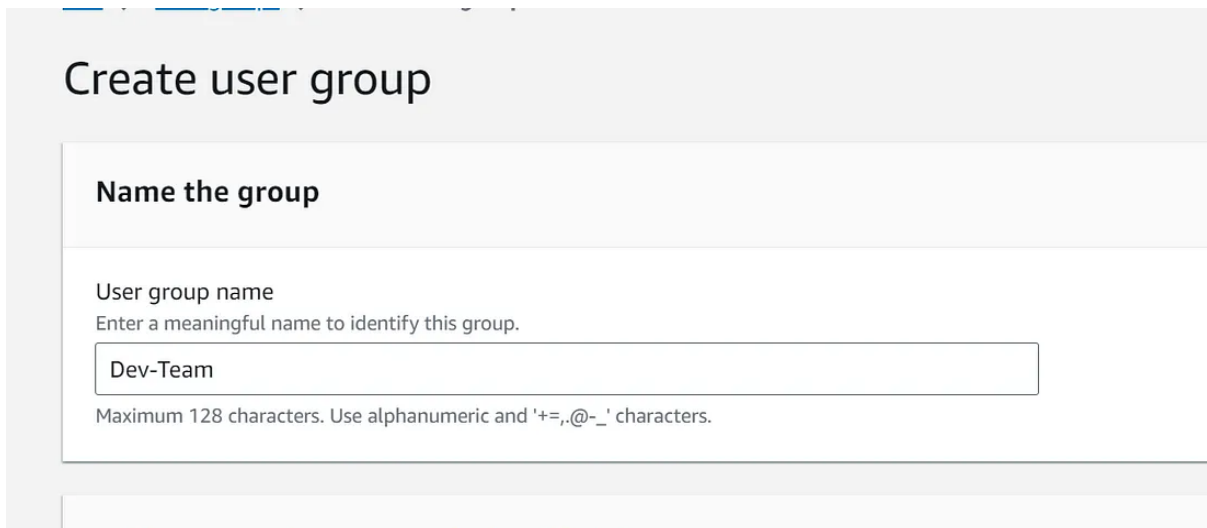
Repeat the steps to create 4 IAM users by the name *HarryPotter*, *AlbusDumbeldore*, *HermoineGranger*, *SiriusBlack* with the above details.

Task 2. Create IAM Groups and add IAM Users

In this task, we are going to create new IAM groups and will add the users to their respective groups. Moreover, we will be adding permissions to the group so that users within the group have access to the services allocated to them using the permission policies.

1. Select the User groups in the left panel and click on the Create group
2. Set Group Name:

i) User group name: Enter *Dev-Team*



Create user group

Name the group

User group name
Enter a meaningful name to identify this group.

Maximum 128 characters. Use alphanumeric and '+=, @-_' characters.

ii) Scroll down and select the users and Add Users to the group.

iii) Scroll down to the Attach permissions Policies section and search for AmazonEC2ReadOnlyAccess and AmazonS3ReadOnlyAccess policies. These policies provide read access for EC2 and S3 to the added users in the group.

Note: Do not add other policies than the ones mentioned above. You will get an error while creating a group

iv) Review all details and click on the Create group button.

Repeat the same steps to create an HR-Team group.

Task 3. Create EC2 Policy

1. Navigate to the Services menu at the top, then click on IAM in the Security, identity, & Compliance section.
 2. In the left menu, select Policies.
- i) Click on Create Policy button.

ii) Under Visual, Type EC2 in the search box and select EC2.

iii) In the Actions, specify the actions allowed in EC2. For this service, We'll choose List.

▼ Actions allowed

Specify actions from the service to be allowed.

Q Filter Actions

Effect

☒ Allow ☐ Deny

Manual actions | [Add actions](#)

☒ All EC2 actions (ec2:*)

Access level

Expand all | Collapse

▶ List (Selected 168/168)

▶ Read (Selected 31/31)

▶ Write (Selected 406/406)

▶ Permissions management (Selected 5/5)

▶ Tagging (Selected 2/2)

iv) Click on Resources, scroll down and choose All resources so that there is no need to specify the resource ARN.

Policy details

Policy name

Enter a meaningful name to identify this policy.

EC2Policy

Maximum 128 characters. Use alphanumeric and '+=, @-_' characters.

Description - optional

Add a short explanation for this policy.

EC2 Full Read and List access

Maximum 1,000 characters. Use alphanumeric and '+=, @-_' characters.

Allow (1 of 384 services)			<input type="checkbox"/> Show remaining 383 services
Service ▲	Access level ▼	Resource	
EC2	Full access	All resources	

v) Create the Policy

✓ Policy EC2Policy created.

[IAM](#) > Policies

Policies (1133) [Info](#)

A policy is an object in AWS that defines permissions.

Filter by Type

[1](#) [2](#) [3](#) [4](#) [5](#) [6](#)

Similarly we can create IAM policy for S3, DynamoDB

I hope you enjoyed reading and learning about AWS IAM.