

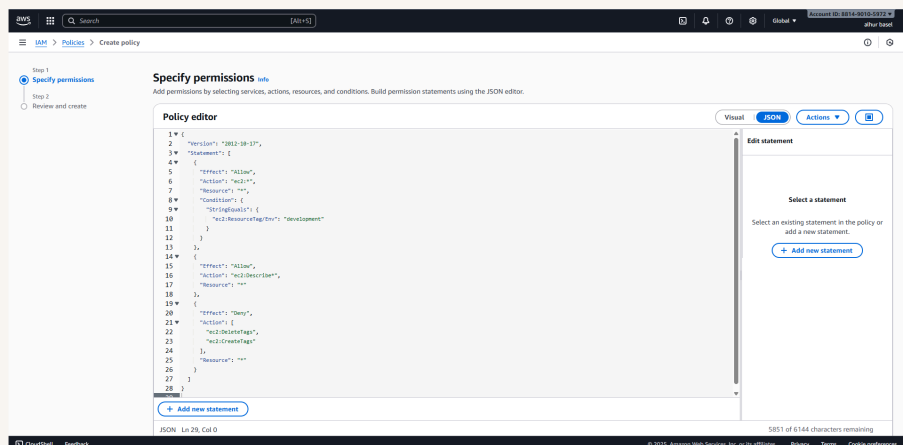


[nextwork.org](https://nextwork.org)

# Cloud Security with AWS IAM



al7r





**al7r**

NextWork Student

[nextwork.org](https://nextwork.org)

# Introducing Today's Project!

In this project, I will demonstrate how to create and manage AWS Identity and Access Management (IAM) users, groups, and policies, and how to launch an EC2 instance. I'm doing this project to learn how to control authentication and authorization in AWS, manage permissions securely, and gain hands-on experience with core AWS services that are essential for cloud computing, security, and DevOps.

## Tools and concepts

I learnt EC2 for launching virtual servers, IAM for creating users, groups, and policies, tags to organize resources, account aliases for easier logins, and how to control permissions to safely separate development and production environments.

## Project reflection

This project took me approximately 1 hour. The most challenging part was setting up the IAM policy correctly to restrict access to development while denying production. It was most rewarding to test the intern's access and see that the policy worked exactly as intended.

# Tags

Tags are labels attached to AWS resources. They help organize, filter, and identify resources, track costs, and apply policies based on environment or purpose.

The tag I've used on my EC2 instances is called Env. The value I've assigned for my instances is production for the first instance and for the second instance.

Launch an instance

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

**Name and tags**

Key	Value	Resource types
Name	nextwork-dev-al7r	Select resource types
Env	development	Select resource types

**Application and OS Images (Amazon Machine Image)**

Search our full catalog including 1000s of application and OS images

Recent Quick Start

Amazon Linux Ubuntu Windows Red Hat SUSE Linux Debian

**Summary**

- Number of instances: 1
- Software Image (AMI): Amazon Linux 2023 AMI 2023.8.2
- Virtual server type (Instance type): t3.micro
- Firewall (security group): New security group
- Storage (volumes): 1 volume(s) - 8 GB

Launch instance



al7r

NextWork Student

[nextwork.org](https://nextwork.org)

# IAM Policies

IAM Policies are rules that define who can access AWS resources and what actions they can perform. They control permissions for users, groups, or roles to ensure resources are used securely and appropriately.

## The policy I set up

I set up the policy using JSON to define precise permissions for the access to the development EC2 instance.

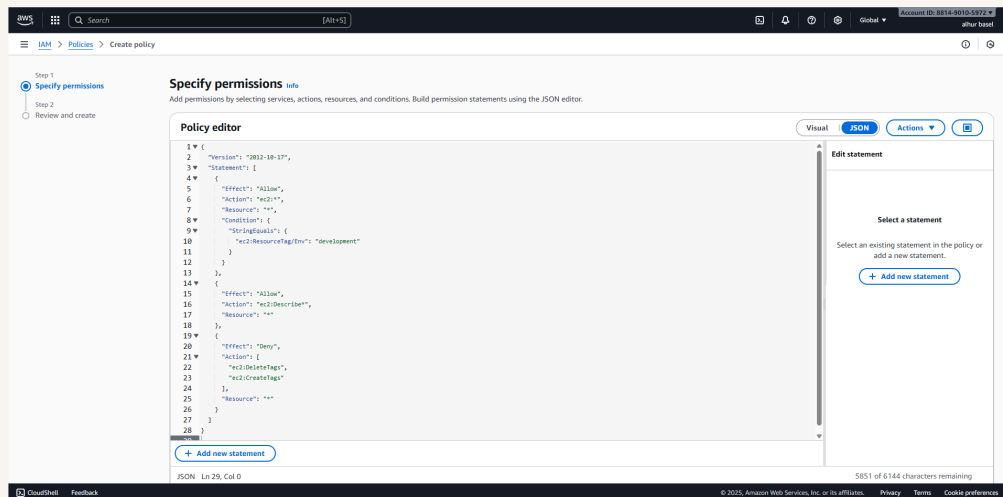
The effect of my policy is to allow the intern to manage only development EC2 instances while denying any actions that could affect production instances or tags, keeping resources secure.

## When creating a JSON policy, you have to define its Effect, Action and Resource.

Effect specifies whether the policy allows or denies an action. Action lists the operations the policy controls, like starting or stopping instances. Resource defines which AWS resources the policy applies to, such as specific EC2 instances or all resources.



# My JSON Policy

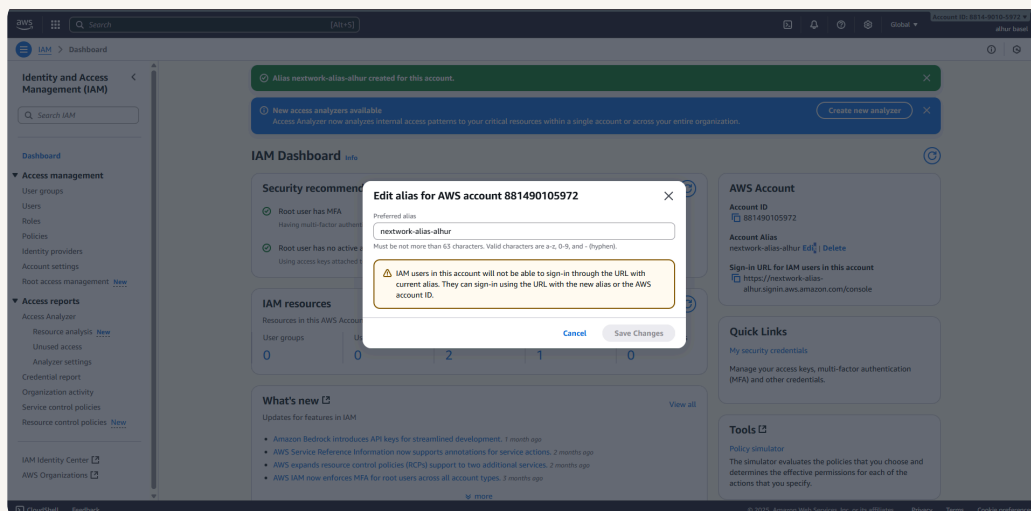




# Account Alias

An account alias is a friendly name for your AWS account that replaces the long numeric account ID in the login URL, making it easier to remember and share with team members.

Creating an account alias took me only a minute. Now, my new AWS console sign-in URL is <https://nextwork-alias-alhur.signin.aws.amazon.com/console>





**al7r**

NextWork Student

[nextwork.org](https://nextwork.org)

# IAM Users and User Groups

## Users

IAM users are individual accounts created in AWS for people or applications, giving them a secure way to log in and access resources with permissions defined by policies or groups.

## User Groups

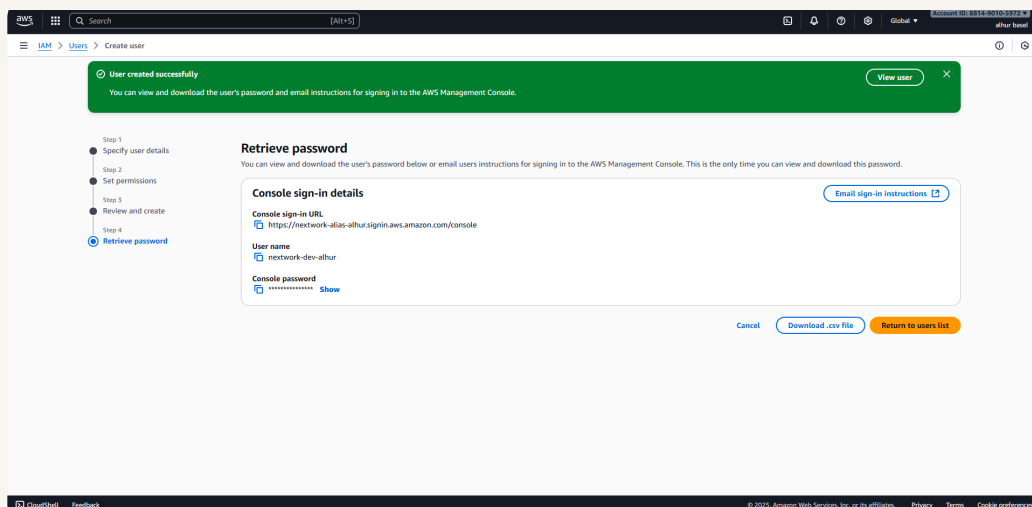
IAM user groups are collections of IAM users that let you manage permissions for multiple users at once by attaching policies to the group instead of each user individually.

Attaching a policy to my user group applies the permissions in that policy to all users in the group, ensuring they can access only the resources allowed, like the development EC2 instance, without managing each user individually.

# Logging in as an IAM User

You can share a new user's sign-in details by sending the login URL, username, and temporary password directly, or by having them set up their own password through an email invitation from AWS.

Once I logged in as my IAM user, I noticed that some dashboard panels showed Access Denied. This was because the user only has permissions assigned through the NextWorkDevEnvironmentPolicy and cannot access production resources.







## Stopping the production instance

The screenshot shows the AWS IAM console interface. At the top, there's a navigation bar with "AWS" logo, search bar, user profile, and language settings. Below it, the left sidebar contains navigation links like "EC2 Global View", "Instances", "Dashboard", etc. The main area displays an error message:

**Failed to load instance i-06d058e9c4dbde1966**

You are not authorized to perform this operation. User arn:aws:iam::81470592:user/network-dev-alias is not authorized to perform ec2:DescribeInstances or resource arn:aws:ec2::us-east-1:184104105927:instance-f-005e5dab51366 because no identity-based policy allows the ec2:DescribeInstances action. Encoded authorization failure message: L\_BwYHrTJQZv-KM9E... [The rest of the encoded message is truncated]

Below the error message, the "Instances (1/22) view" section is visible, showing a table with columns: Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Availability Zone, Public IPv4 DNS, Public IPv4 VPC Elastic IP, Private IP, and Tags.

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 VPC Elastic IP	Private IP	Tags
nesteqg-com-	i-06d058e9c4dbde1966	Running	t3.micro	All checks passed	No alarms present	mumbai-1	pq-137-175-27-234.m...	152.175-27-234...	-	flo



# Testing IAM Policies

## Stopping the development instance

Next, when I tried to stop the development instance, it stopped successfully. This was because my IAM user has permissions for instances tagged with Env = development.

