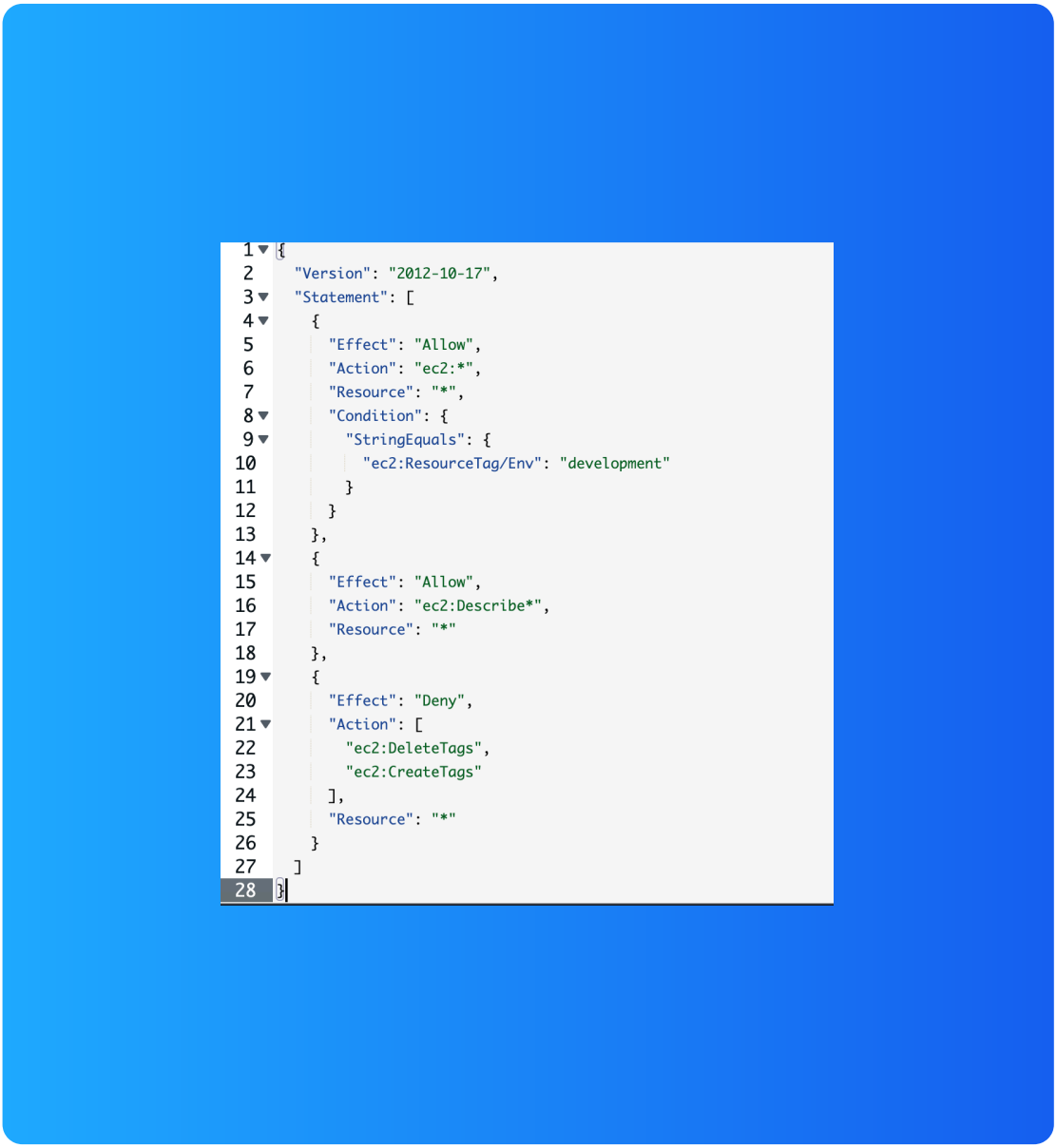
NextWork.org

# Cloud Security with AWS IAM

Osman Kpaka



O
**Osman Kpaka**
NextWork Student


# Introducing today's project!

## What is AWS IAM?

AM (Identity and Access Management) is a service in AWS that controls access to resources. It allows you to create and manage users, groups, roles, and permissions, ensuring secure and fine-grained access control for your AWS environment.

## How I'm using AWS IAM in this project

I created and applied a policy that permitted EC2-related actions on instances with the "Development" environment tag, allowing the ability to stop development instances while restricting the ability to stop production instances that are tagged.

## One thing I didn't expect...

One thing that surprised me about this project was importance of using an IAM administrator account instead of the root user for configuration. This simple step adds a layer of security, keeping the root account more isolated and reducing risk.

## This project took me...

Around 1 hour

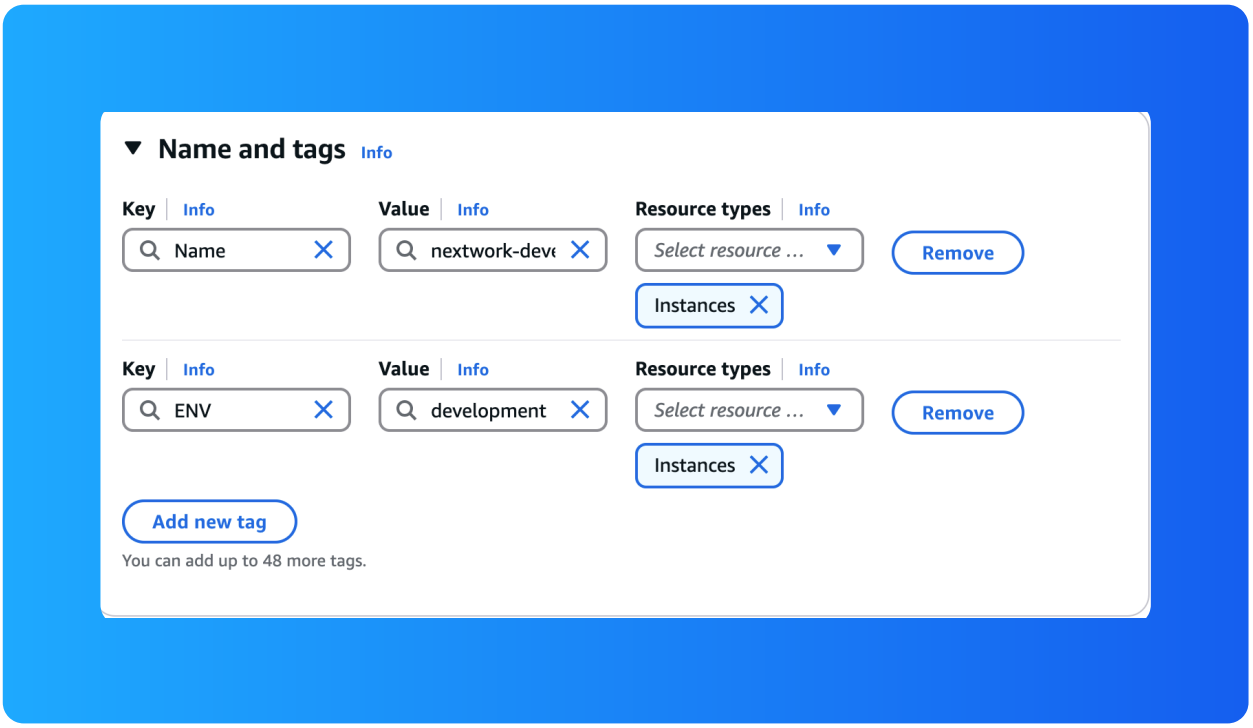
O
**Osman Kpaka**
NextWork Student



# Tags

Tags are key-value pairs used to label AWS resources for identification, organization, and management. They help with cost allocation, resource tracking, automation, and security by enabling easier search, filtering, and reporting.

The tag values I used are "Production" and "Development," representing the two distinct environments we are using for building and releasing the network application. These tags help differentiate between the stages of development and deployment



O
**Osman Kpaka**
NextWork Student


# IAM Policies

IAM policies define permissions for AWS users, groups, and roles, specifying what actions they can or cannot perform on AWS resources. They control access to services, resources, and actions, ensuring secure and fine-grained permissions management.

## The policy I set up

I've set up a policy using JSON for instances tagged with "ENV=development" while denying the ability to create or delete tags for all instances.

The policy affect should allow modifications for Development tagged Enviornments

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

The "Effect" can be "Allow" or "Deny," with Deny taking priority. "Action" lists allowed or denied actions (e.g., "ec2:" for all EC2 actions). "Resource" defines which resources the policy applies to, with "" indicating all resources

O
Osman Kpaka
NextWork Student



# My JSON Policy

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "ec2:*",
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "ec2:ResourceTag/Env": "development"
        }
      }
    },
    {
      "Effect": "Allow",
      "Action": "ec2:Describe*",
      "Resource": "*"
    },
    {
      "Effect": "Deny",
      "Action": [
        "ec2:DeleteTags",
        "ec2:CreateTags"
      ],
      "Resource": "*"
    }
  ]
}
```

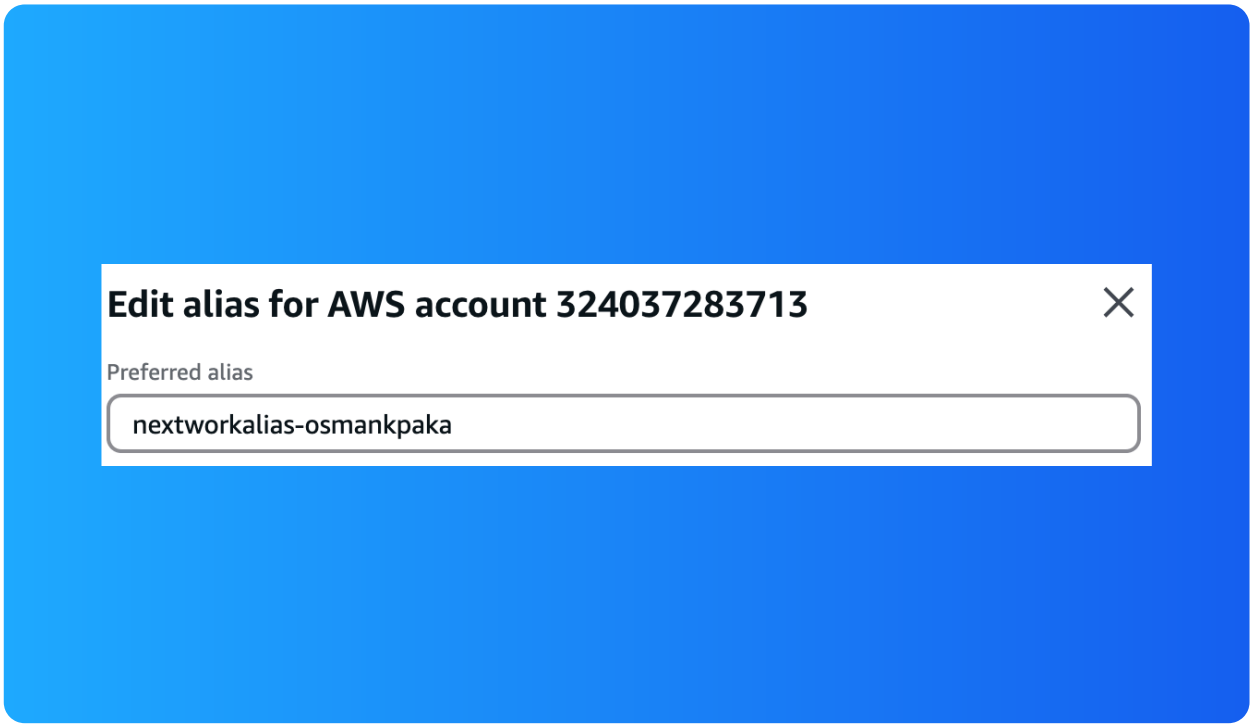
Osman Kpaka
NextWork Student



# Account Alias

An Account Alias is a friendly name for your AWS account that you can use instead of your account ID

It took a few minutes



O
**Osman Kpaka**
NextWork Student


# IAM Users and User Groups

## Users

An IAM user is an entity within AWS that represents an individual or application with specific permissions. It has unique credentials (username/password or access keys) to interact with AWS resources based on assigned policies.

## User Groups

IAM user groups are collections of IAM users that allow you to manage permissions for multiple users at once

Attaching a policy to an IAM user group applies the policy to all users in that group, granting or restricting their permissions based on the policy's rules.

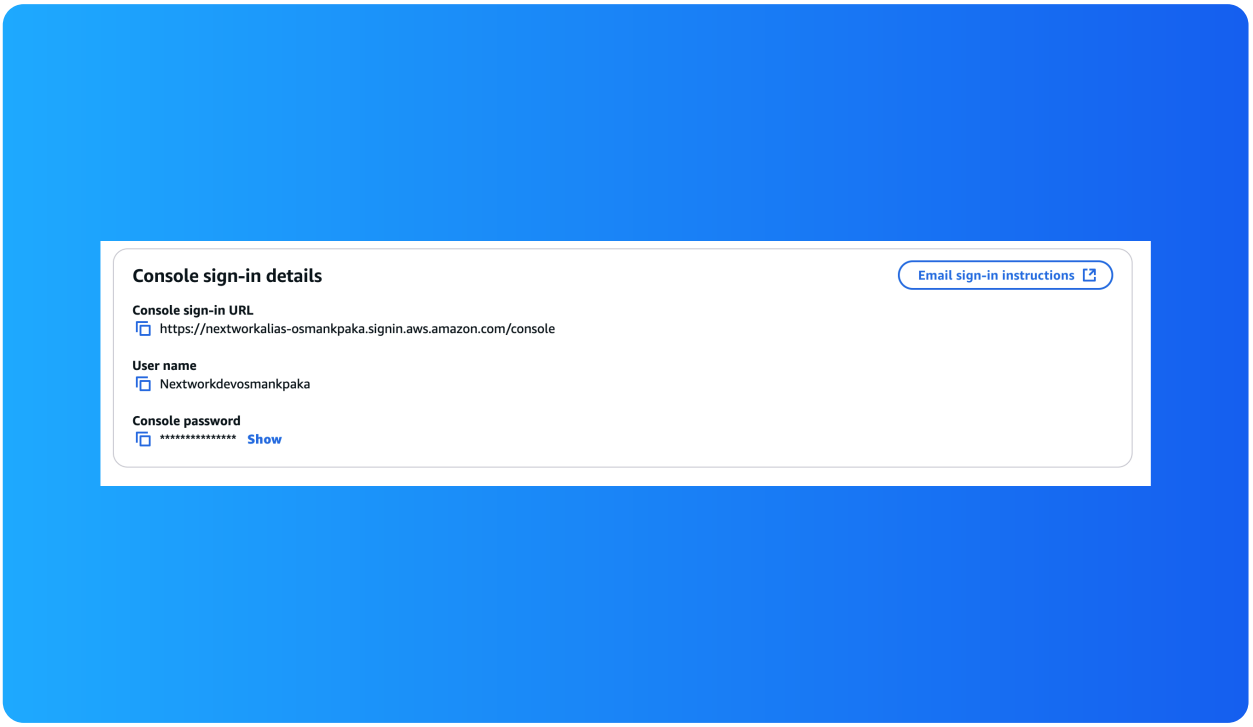
O
**Osman Kpaka**
NextWork Student



# Logging in as an IAM User

You can share a new user's sign-in details by either sending them an email invitation to set up their account or by manually providing the username and password/access keys for direct login.

There are many Access Denied error codes that are happening on the user-interface



Osman Kpaka
NextWork Student



# Testing IAM Policies

I stopped Development instances and it worked.

## Stopping the production instance

It was denied this was because the policy in place from the JSON file that I created prevented modification from the production instance

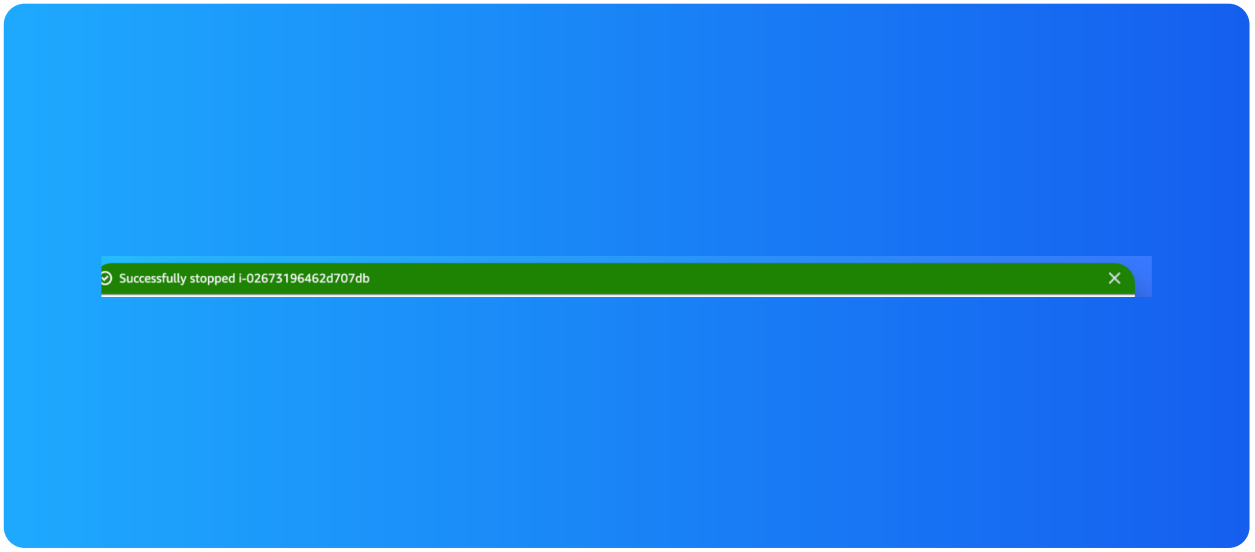
Osman Kpaka
NextWork Student



# Testing IAM Policies

## Stopping the development instance

Next, when I attempted to stop the development instance, the action was permitted because my policy allowed it.



# Everyone should be in a job they love.

Check out nextwork.org for more projects