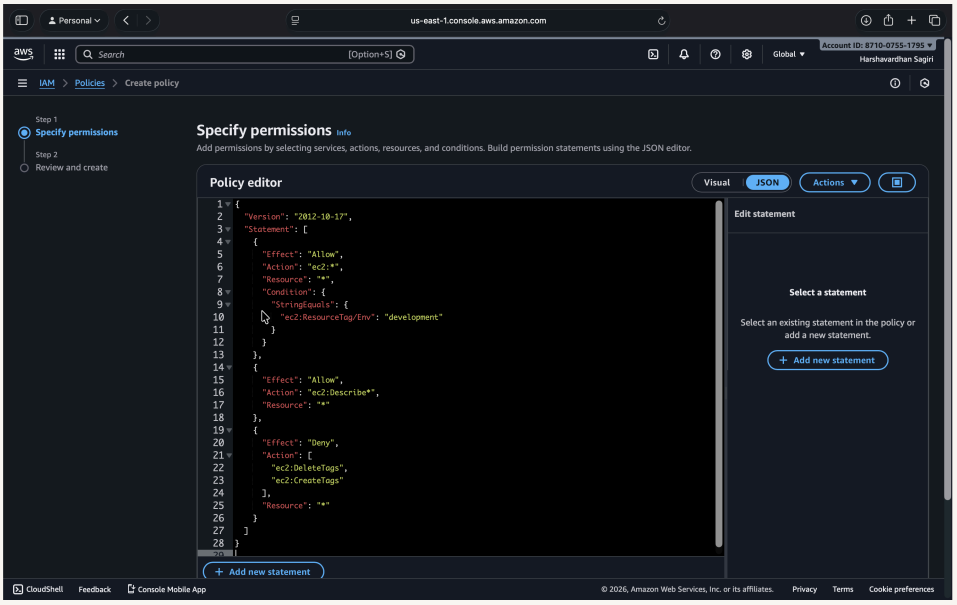
nextwork.org

Cloud Security with AWS IAM

Harshavardhan Sagiri



Harshavardhan Sagiri
NextWork Student

Introducing Today's Project!

Project overview

In this project, we will demonstrate how to use AWS IAM to control access and permission settings in our AWS account. We’re doing this project to learn about cloud security from the absolute foundations. Every company thinks about access permissions, and there are even entire jobs called “IAM Engineers” focused on the skills we’re about to build today.

Tools and concepts

In this project, we used Amazon EC2 and AWS Identity and Access Management (IAM). The key concepts we learned include IAM users, IAM policies, user groups, and account aliases. We also learned how to control access to AWS resources by assigning permissions through policies and groups, how to restrict access based on environment tags such as development and production, and how to safely test permissions using the IAM Policy Simulator. Overall, this project helped us understand how to manage and secure user access in AWS effectively.

Project reflection

This project took me approximately 2 hours to finish it. The most challenging part was deleting the production user in alias account as i ran into to a small error and it was most rewarding to use policy simulator when it came into play i was really impressed by that, because it just saves us the time.

Harshavardhan Sagiri
NextWork Student

Tags

What I did in this step

In this step, we will launch two EC2 instances because we need to boost NextWork's computing power. We're expecting more users and traffic into our website over the summer break!

Understanding tags

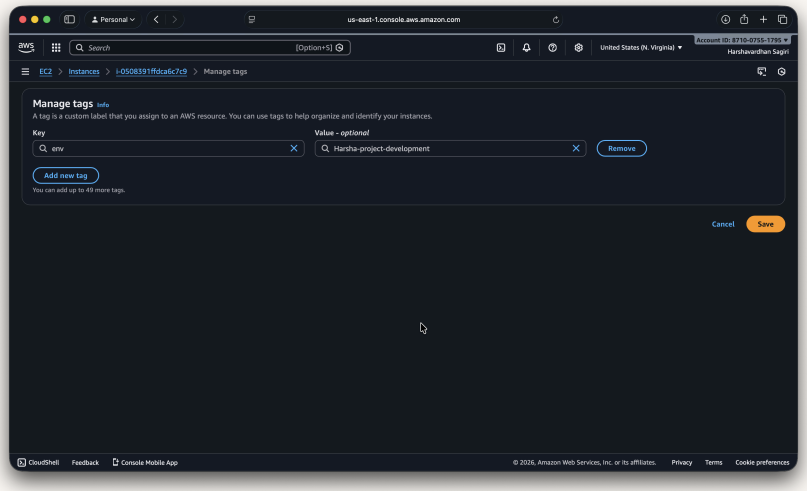
Tags are key-value labels that we attach to AWS resources like EC2 instances. They help us identify, organize, and manage resources more easily, especially when we have many instances running. For example, we can tag an EC2 instance with values like Environment: development.

My tag configuration

The tag I've used on my EC2 instances is called Environment. The value I've assigned to both instances is development. This helps clearly identify that these EC2 instances are part of the development environment and separates them from resources used for testing or production.

Harshavardhan Sagiri
NextWork Student

nextwork.org



Harshavardhan Sagiri
NextWork Student

IAM Policies

What I did in this step

In this step, we will use IAM policies to control the access level of a new NextWork intern because they should have access to the development environment (i.e. the development instance) but NOT the production environment.

Understanding IAM policies

IAM Policies are like rules that determine who can do what in our AWS account. We're using policies today to control who has access to our production environment instance.

The policy I set up

For this project, I set up the IAM policy using JSON. Using JSON gave me more control over the permissions and allowed me to clearly define which actions are allowed on the development EC2 instance while restricting access to the production environment.

Policy effect

I've created a policy that allows full EC2 access only to instances that are tagged with Env = development. It also allows the user to view EC2 resources using describe actions across the account.

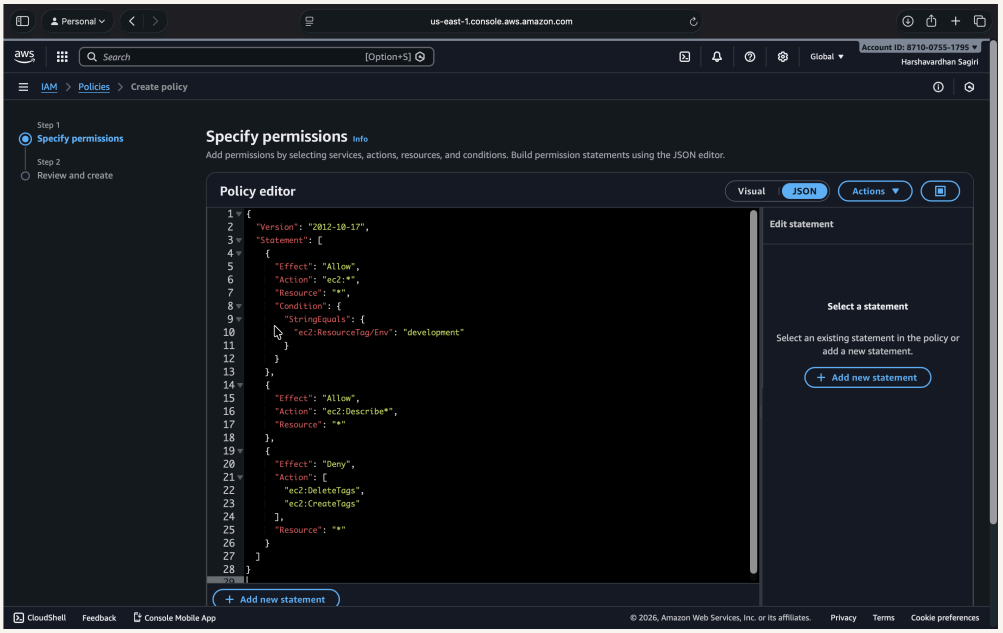
At the same time, the policy explicitly denies the ability to create or delete EC2 tags, preventing the user from changing resource tags to gain additional access. Overall, this policy ensures the user can manage development EC2 instances while protecting production resources from unauthorized changes.

Understanding Effect, Action, and Resource

The Effect defines whether a permission is allowed or denied. The Action specifies what actions can or cannot be performed, such as starting or stopping an EC2 instance. The Resource defines which AWS resources the policy applies to, such as specific EC2 instances or all instances in an account. Together, these elements control exactly who can do what on which resources in AWS.

Harshavardhan Sagiri
NextWork Student

My JSON Policy



Harshavardhan Sagiri
NextWork Student

Account Alias

What I did in this step

In this step, we will set up an Account Alias, which is like a nickname for our AWS account's console login. This is because an account alias makes it simpler for our users to log in.

Understanding account aliases

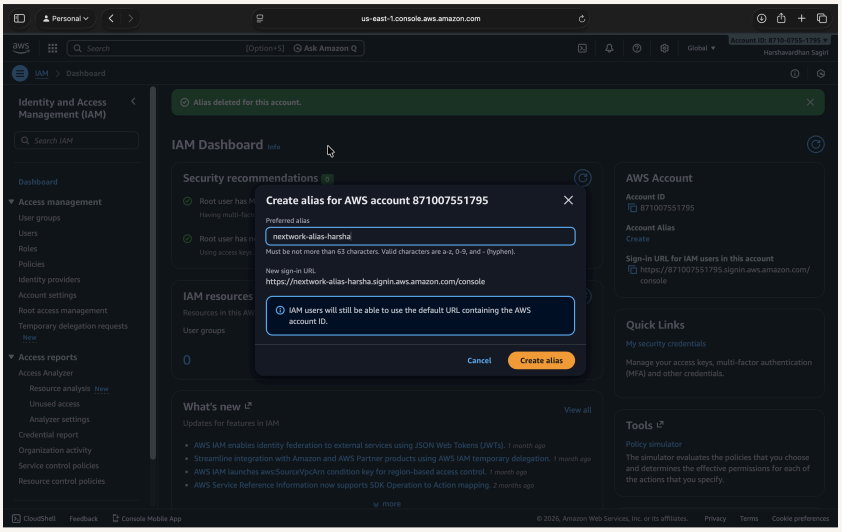
An account alias is just a nickname for our AWS account, Instead of having a long Account ID, we can now reference our account alias instead

Setting up my account alias

It took me just 30 seconds to setup as it is a simple configuration in the IAM dahsboard. now our new console sign-in URL is https://nextwork-alias harsha.signin.aws.amazon.com/console now instead of my account ID the new URL that i have created as alias will be using instead of my account ID

Harshavardhan Sagiri
NextWork Student

nextwork.org



Harshavardhan Sagiri
NextWork Student

IAM Users and User Groups

What I did in this step

In this step, I will create an IAM user and user group for the intern so I can control their permissions securely without sharing the main AWS account.

Understanding user groups

IAM user groups are like folders that collect IAM users that so that you can apply permisssion settings at the group level.

Attaching policies to user groups

I attached the policy that we created to this user group which means any user created inside this user group will automatically get the permissions attached to next to our NextworkDevEnvironment Polic. IAM Policy

Understanding IAM users

IAM users are people/entities who can login into our AWS Account

Harshavardhan Sagiri
NextWork Student

Logging in as an IAM User

Sharing sign-in details

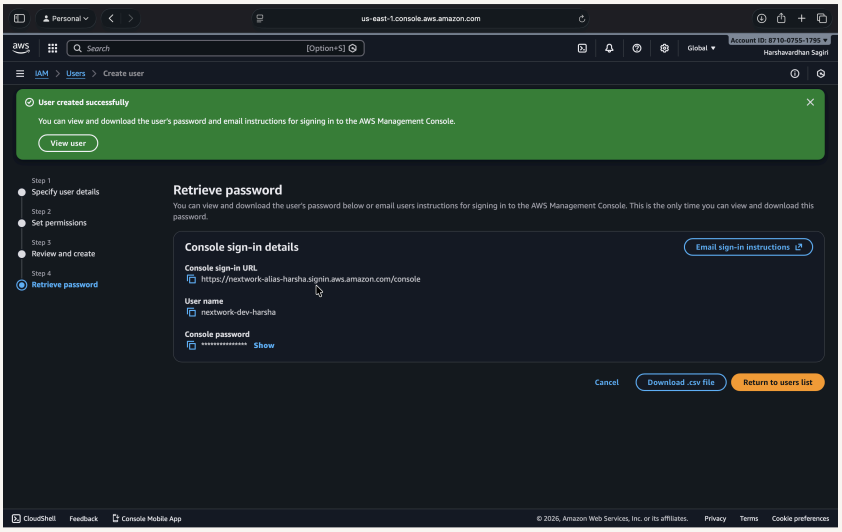
The first way is to email sign-in instructions to the user, while the second way is download a csv file with the sign-in details inside

Observations from the IAM user dashboard

Once I logged in as my IAM user, we noticed that our user is already denied the access to the items or panels of main AWS dashboard because we only set up permissions to EC2 instance. sop our intern wouldn't have access to anything else and even see anything else

Harshavardhan Sagiri
NextWork Student

nextwork.org



Harshavardhan Sagiri
NextWork Student

Testing IAM Policies

What I did in this step

In this step, I will login to our AWS account as a intern and test access to the production and development instances because we want to make sure our intern doesn't have the ability to do anything that can effect our production environment

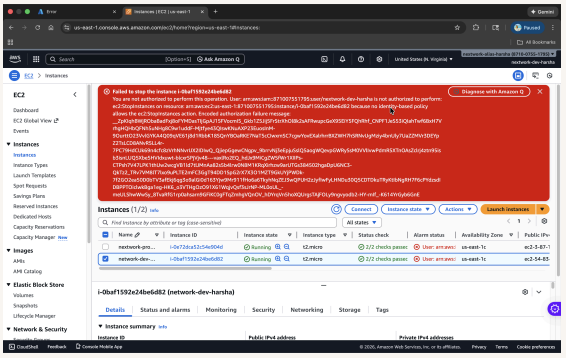
Testing policy actions

tested my JSON IAM policy by attempting to stop both EC2 instances. The stop action succeeded on the development instance but was denied on the production instance, which confirmed that the policy correctly restricts actions based on the instance’s environment tag.

Stopping the production instance

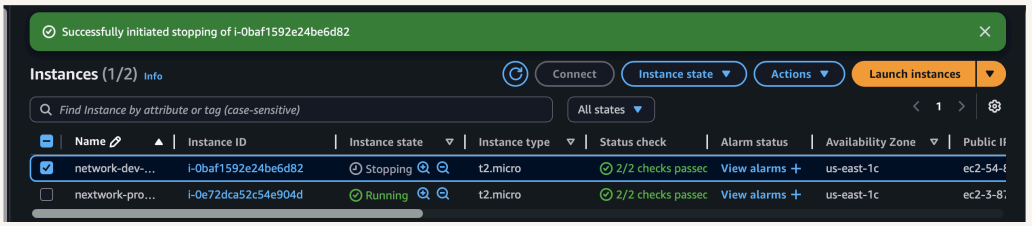
When I tried to stop the production instance, the action was denied and I received an authorization error. This happened because the IAM policy attached to my user group only allows EC2 actions on instances tagged with Env = development and does not permit changes to production resources. This confirms that the policy is correctly restricting access to the production environment.

Harshavardhan Sagiri
NextWork Student



Stopping the development instance

When I tried to stop the development instance, the action was successful. This was because the EC2 instance was tagged with Env = development, and my IAM policy explicitly allows stop actions on instances with this tag. This confirms that the policy is working as intended for the development environment.



Harshavardhan Sagiri
NextWork Student

IAM Policy Simulator

To extend our project, we're going to test our permission policies in a safer and more controlled way using a tool called the IAM Policy Simulator. We're doing this because having to stop instances and log into AWS accounts as other users is a bit disruptive. Let's find a more efficient way.

Understanding the IAM Policy Simulator

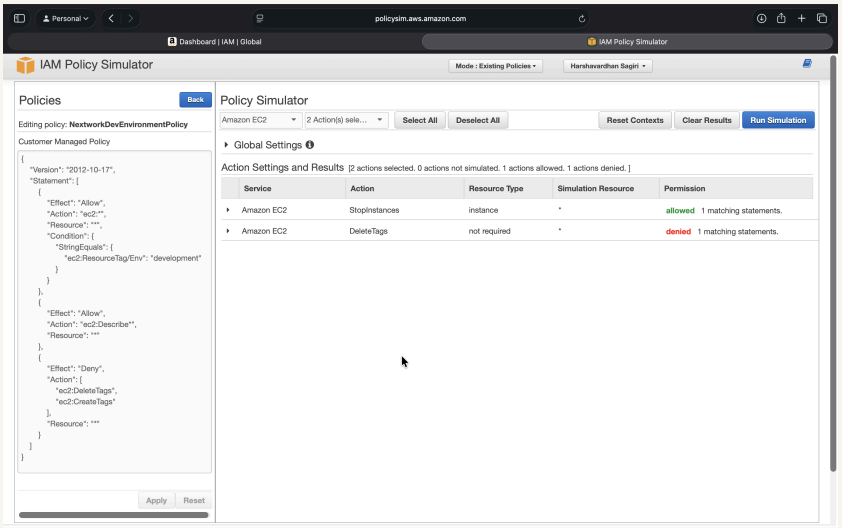
The IAM Policy Simulator is a tool that lets us simulate actions and test permission settings by defining a specific user, group, or role and the actions we want to test. It is useful for saving time when testing permissions because it allows us to validate policies without logging in as another user or actually stopping or modifying real AWS resources.

How I used the simulator

The IAM Policy Simulator is a tool that allows us to safely test and validate IAM permission policies without affecting real AWS resources. It lets us simulate actions by selecting a specific user, user group, or role and defining the actions we want to test. This helps us understand whether a policy allows or denies certain actions before applying it in a real environment. Using the simulator saves time and reduces risk because we do not need to log in as different users or actually stop, start, or modify EC2 instances. Overall, it provides a safer, more controlled, and more efficient way to verify IAM permissions.

Harshavardhan Sagiri
NextWork Student

nextwork.org



The place to learn & showcase your skills

Check out nextwork.org for more projects