nextwork.org

Cloud Security with AWS IAM

NI Nikhila Reddy

NI
Nikhila Reddy
NextWork Student

Introducing Today's Project!

In this project, I will demonstrate how to use AWS IAM to control access and permission in our AWS account. I'm doing this to learn cloud security from foundation—access control is essential, and IAM engineers focus entirely on mastering these skills

Tools and concepts

Services I used were Amazon EC2 and AWS IAM. Key concepts I learned include IAM users, groups, policies, and account aliases. I also learned how to use the Policy Simulator, write JSON policies, launch and tag instances, and log in as another user.

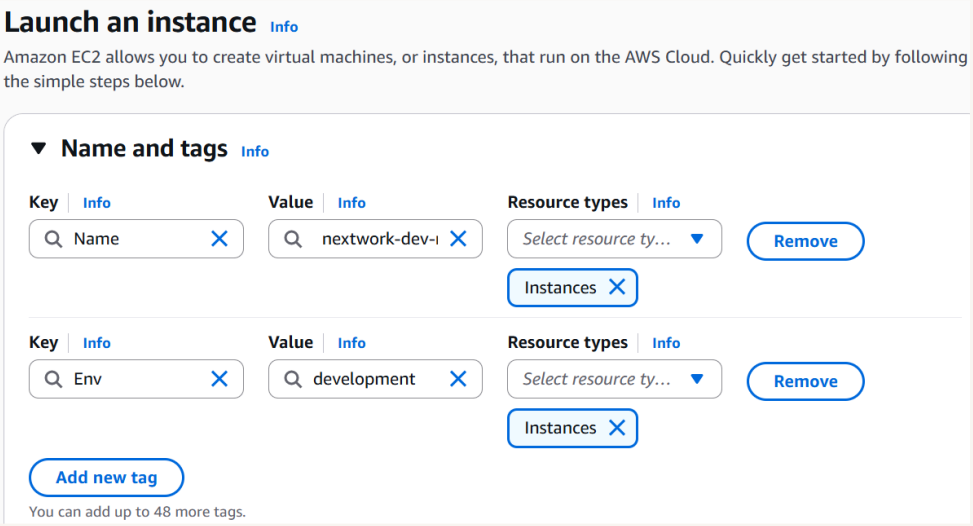
Project reflection

This project took me approximately 1.5 hours today including demo time. The most challenging part was understanding the JSON IAM policy. It was most rewarding to see “permission denied” when the intern tried to delete the production instance.

Tags

Tags are organizational tools that lets us label our resources. They are helpful for grouping resources, cost allocation and applying policies for all resources with the same tag.

The tag I've used on my EC2 instances is called Env stands for environment. The value I've assigned for my instances are production and development.



IAM Policies

IAM Policies are likes rules that determine who can do what in our AWS account. I am using polices today to control who has access to our production environment instance.

The policy I set up

For this project, I've set up a policy using JSON.

I've created a policy that allows the policy holder (i.e. the intern) to perform any action on instances tagged "development". They can also view details of any instance, but they're denied permission to create or delete tags on any instance.

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

The Effect, Action, and Resource attributes of a JSON policy mean whether the policy allows or denies actions (Effect); what actions the policy holder can or can't perform (Action); and which specific AWS resources the policy applies to (Resource).

My JSON Policy

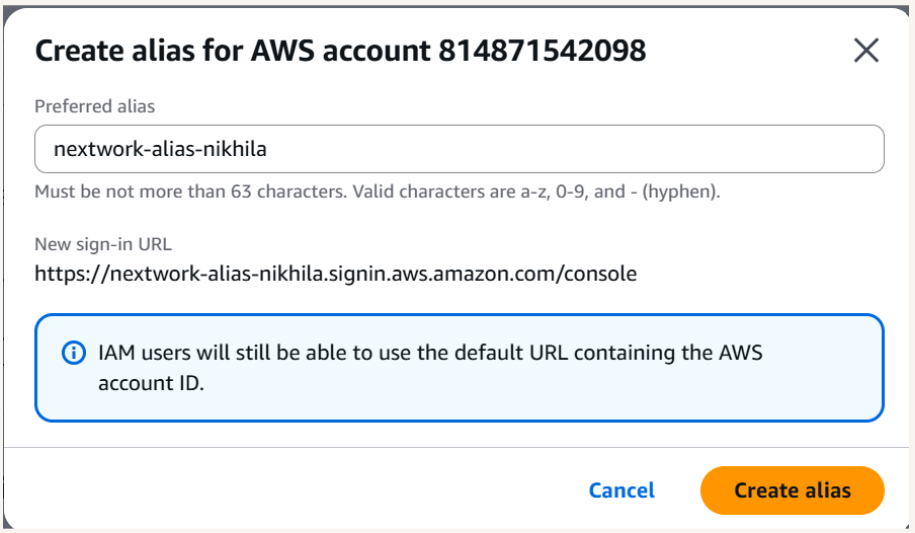
Policy editor

```
{
  "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": "*"
    }
  ]
```

Account Alias

An account alias is simply a nickname for the AWS account instead of a long account ID, the account alia can be referred.

Creating an account alias took me 30 seconds - a simple configuration in the IAM dashboard. Now, my new AWS console sign-in URL uses the alias instead of my account ID.



IAM Users and User Groups

Users

IAM users are people or entities that can have access/can login to the AWS account.

User Groups

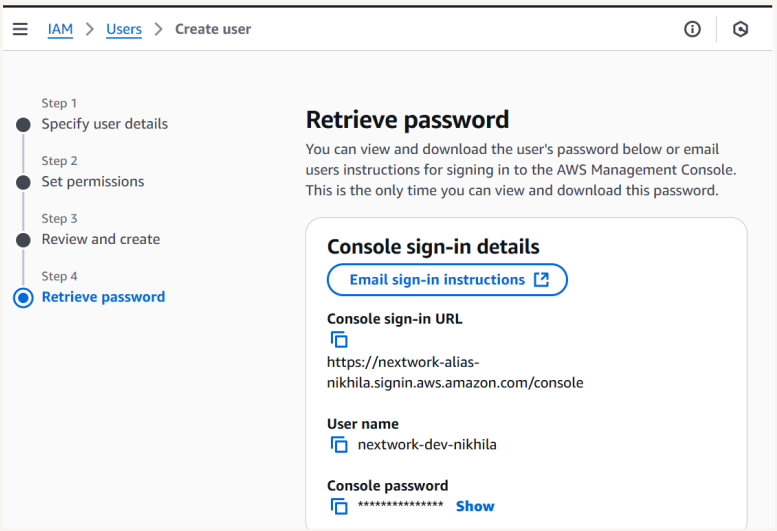
IAM user groups are like folders that collect IAM users so that the permissions settings can be applied at the group level.

I attached the policy I created to this user group, which means any user created inside this group will automatically get yhe permissions attached to the NextWorkDevEnvironmentPolicy IAM policy.

Logging in as an IAM User

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

Once I logged in as my IAM user, I saw that access to many AWS console panels was denied. That's because I only gave permissions for the development EC2 instance, so the intern isn't allowed to view or interact with anything else in the account.

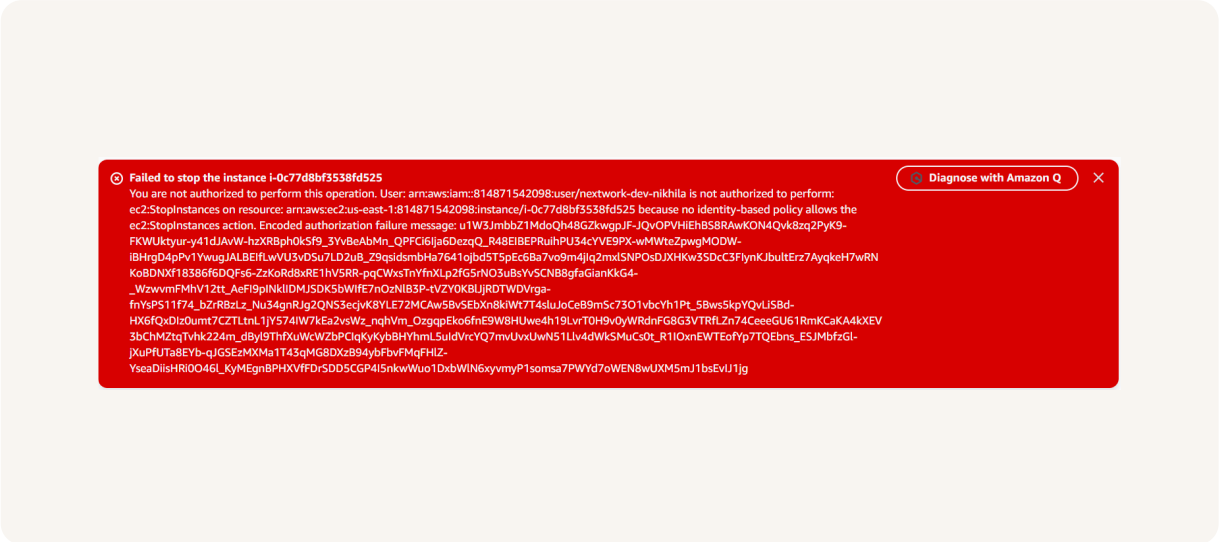


Testing IAM Policies

I tested my JSON IAM policy by attempting to stop both the development and production instances.

Stopping the production instance

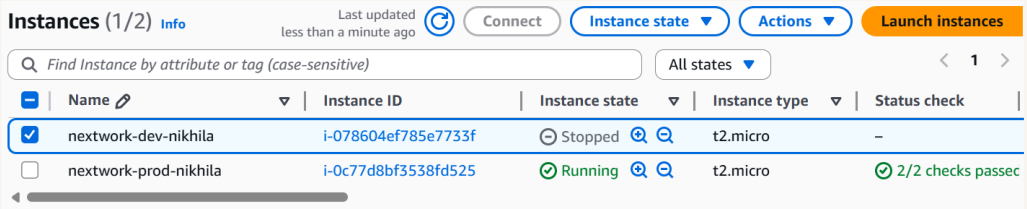
When I tried to stop the production instance I met with an error! This happened because the instance is tagged 'production', which isn't covered by the permission policy—interns only have access to perform actions on instances tagged 'development'.



Testing IAM Policies

Stopping the development instance

Next, when I tried to stop the development instance I successfully saw the instance state change to Stopping and then stopped. This was because the permission policy allows the intern (i.e the users in the nextwork-dev-group) to stop instances.



The place to learn & showcase your skills

Check out nextwork.org for more projects