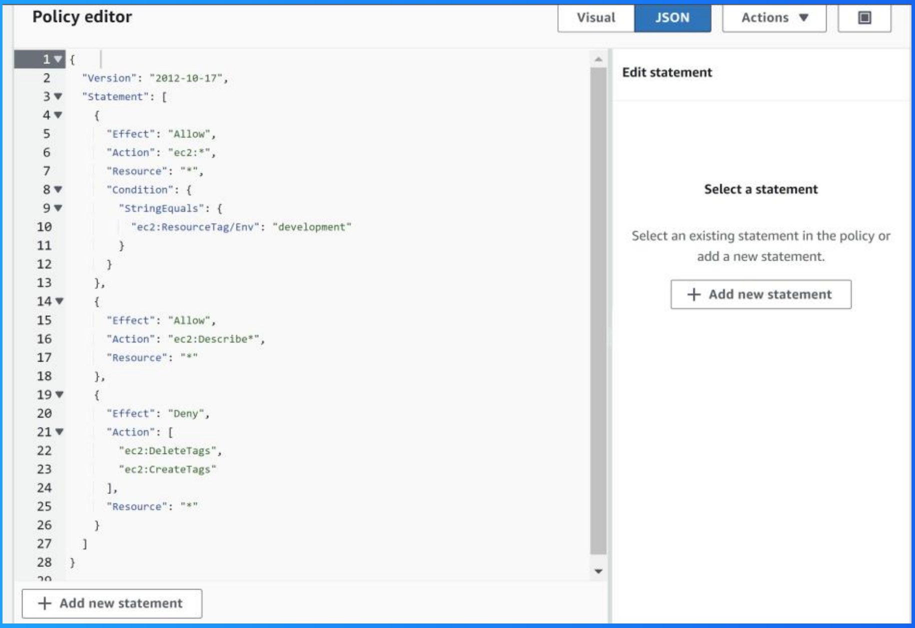
NextWork.org

Cloud Security with AWS IAM

Prajit Venkatachalam

https://www.linkedin.com/in/prajit-venkatachalam-435b2a150/



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

Prajit Venkatachalam
NextWork Student

NextWork.org

Introducing today's project!

What is AWS IAM?

AWS IAM (Identity and Access Management) is useful for security purpose which can be implemented by creating and attaching policies to users and user groups, defining permissions to allow or deny actions on resources within an AWS account

How I'm using AWS IAM in this project

In today's project, I used IAM to create and assign a policy to my users in theuser group, granting access to production and development EC2 instances within the AWS account based on the defined permissions.

One thing I didn't expect...

One thing I didn't expect in this project was encountering an error when trying to delete the production instance. Investigating the reason and root cause helped me learn more about IAM policies.

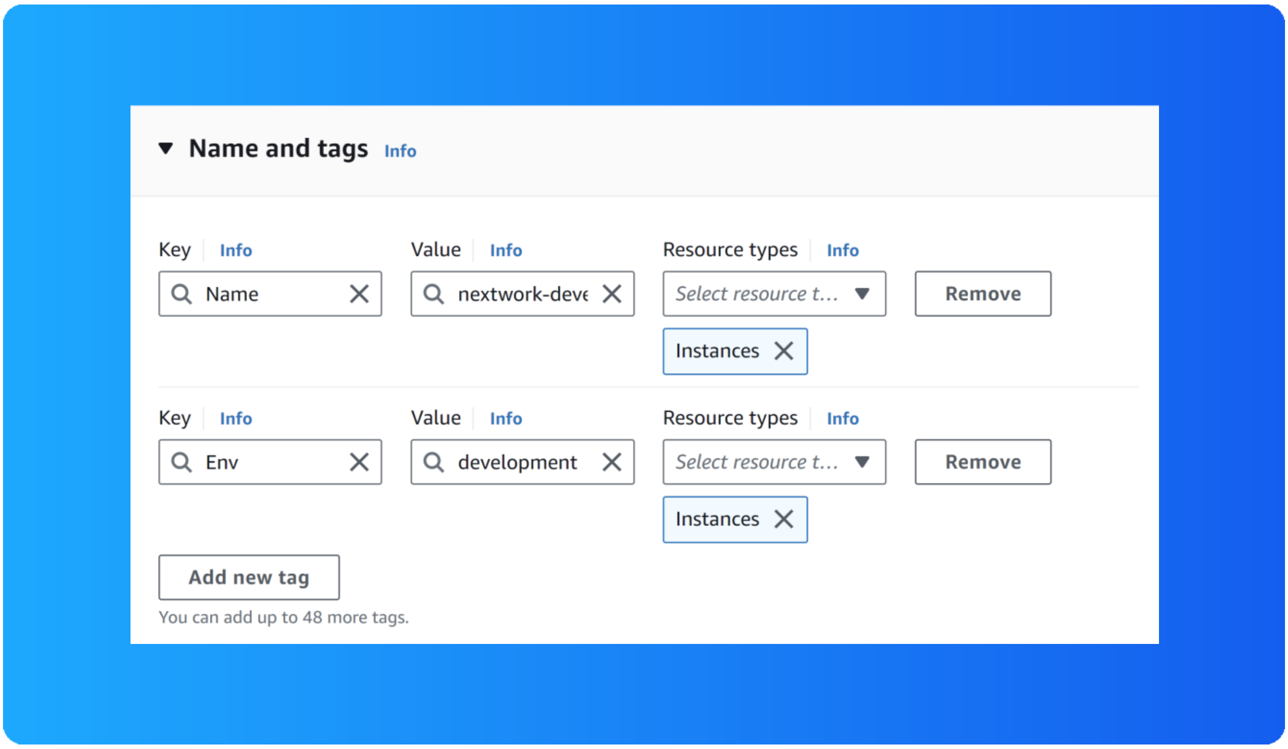
This project took me...

Overall, this project took me an hour and a half to complete, including writing the report.

Tags

I launched two EC2 instances to test the permission settings in AWS IAM. Tags are labels that help users organize and manage resources. They assist in grouping, bulk management, and applying security policies across the AWS environment.

The tags that I used for my two EC2 instances is called Env. The value that I assigned for my EC2 instances are production and development.



▼ Name and tags Info

Key: Name
Value: nextwork-deve
Resource types: Instances

Key: Env
Value: development
Resource types: Instances

You can add up to 48 more tags.

IAM Policies

IAM policies are rules that define permissions, allowing or denying users or resources the ability to perform specific actions on resources within my AWS account.

The policy I set up

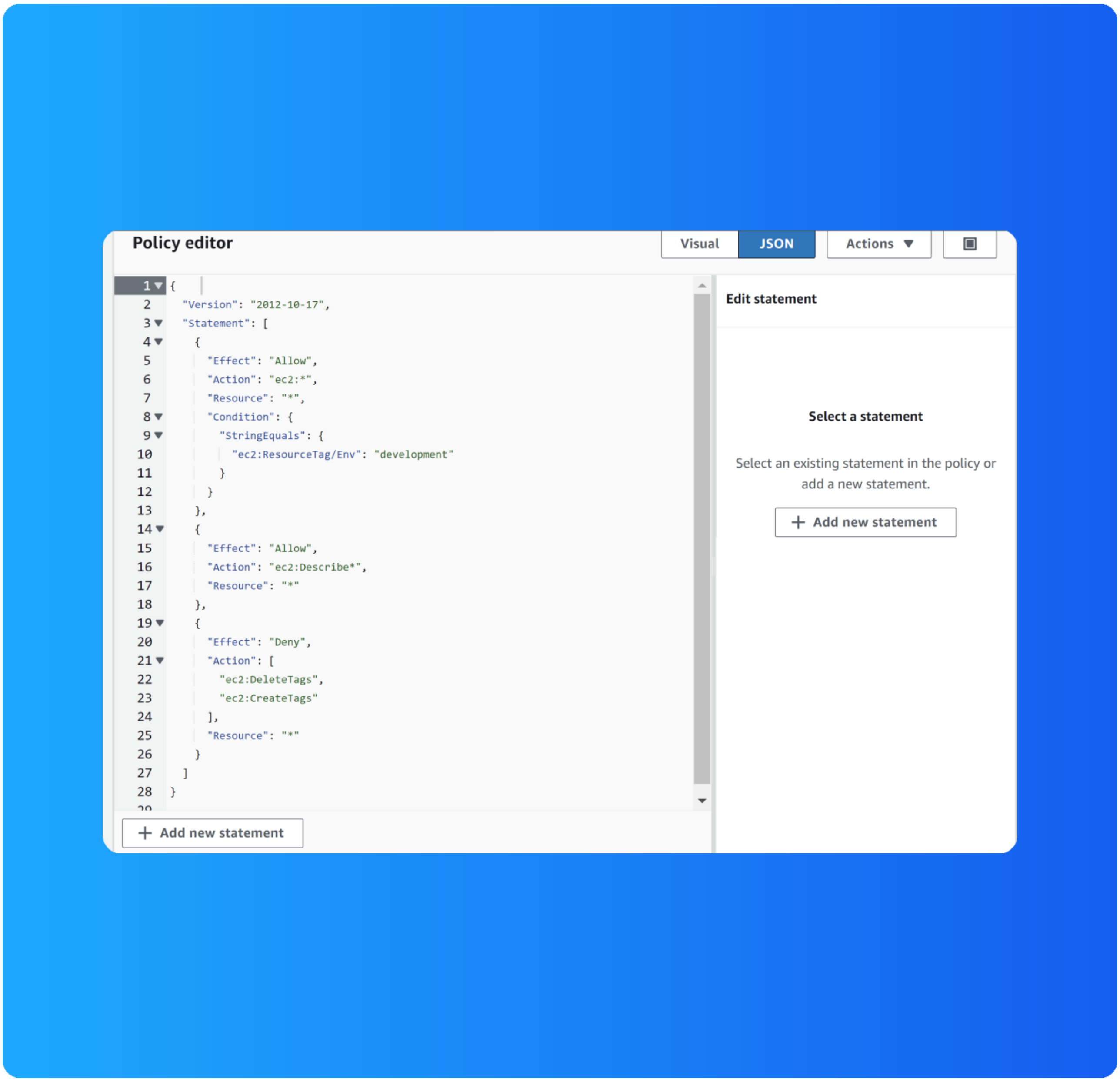
For this project, I used JSON editor to set up a policy.

I created a policy that permits all EC2-related actions for instances with the "development" Environment (Env) tag. However, it denies the creation and deletion of tags for any EC2 instances.

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

The Effect, Action, and Resource attributes in a JSON policy mean: Effect: Permit or block. Action: The specific action to allow or deny. Resource: The AWS resource(s) this policy applies to.

My JSON Policy



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

Edit statement

Select a statement

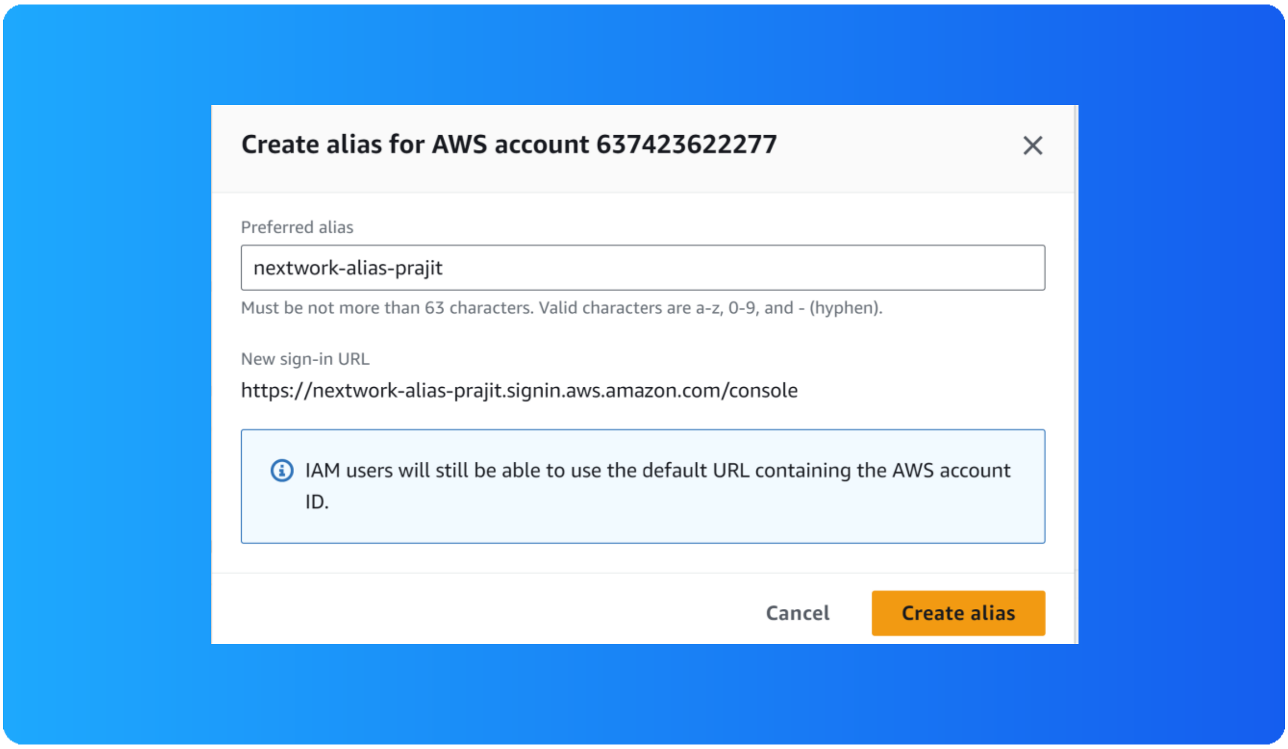
Select an existing statement in the policy or add a new statement.

Account Alias

An Account Alias is a custom name I can assign to my AWS account, replacing the Account ID in the account's login URL for easier identification.

Creating an account alias took me about a minute or lesser.

Now, my new AWS console sign-in URL is https://nextwork-alias-prajit.signin.aws.amazon.com/console



Create alias for AWS account 637423622277

Preferred alias

nextwork-alias-prajit

Must be not more than 63 characters. Valid characters are a-z, 0-9, and - (hyphen).

New sign-in URL

https://nextwork-alias-prajit.signin.aws.amazon.com/console

IAM users will still be able to use the default URL containing the AWS account ID.

IAM Users and User Groups

Users

IAM users are additional logins or individuals who have access to my AWS account, created by me through the AWS IAM service. I can control each user's access to my account's resources and services.

User Groups

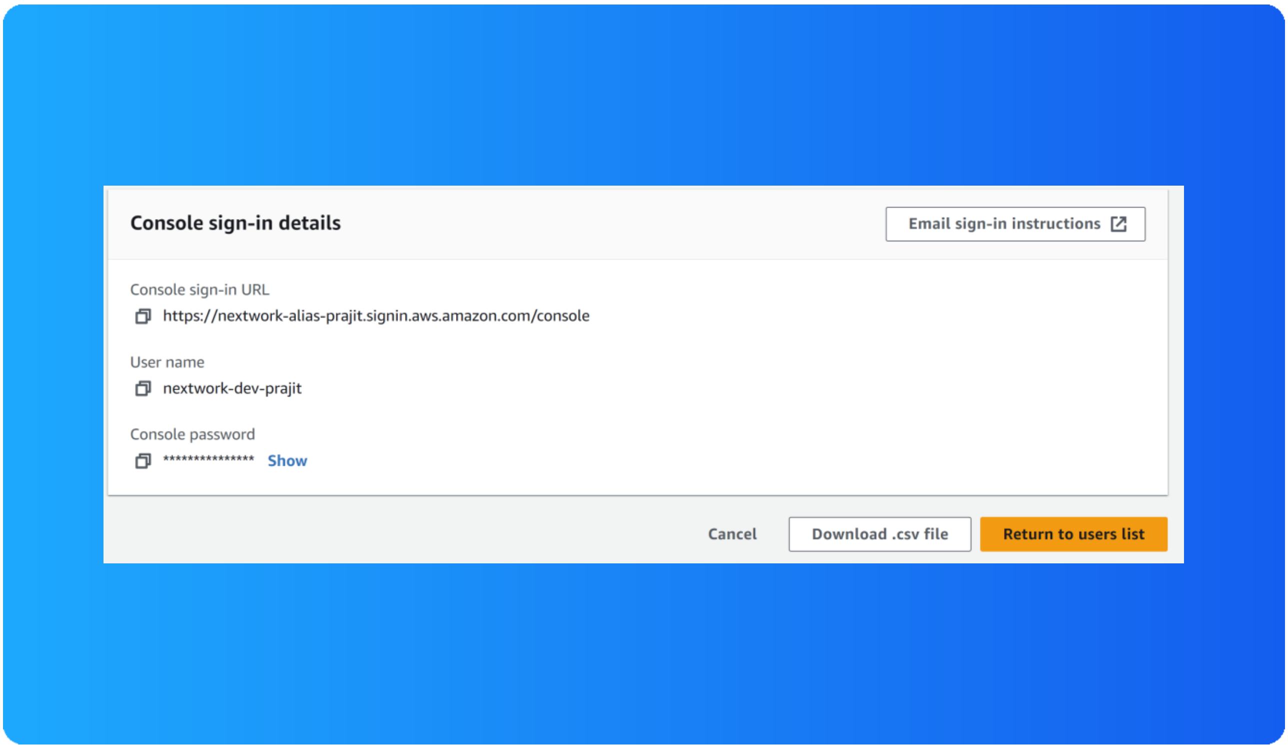
IAM user groups allow for managing user permissions collectively at a group level. They function like folders, simplifying the process of assigning permissions and policies to multiple users at once.

I attached the policy I created to my user group, nextwork-dev-group, so all users added to this group will automatically inherit its access permissions.

Logging in as an IAM User

There are two ways to provide sign-in instructions: by emailing them or downloading the .csv file. Additionally, I selected the option to allow the new user to access the management console when I created the user.

Once I logged in as my IAM user, I noticed that many panels displayed "Access denied." This was a clear difference from the dashboard I usually see in my AWS account, where there are no restrictions on access.



Console sign-in details

Console sign-in URL
https://nextwork-alias-prajit.signin.aws.amazon.com/console

User name
nextwork-dev-prajit

Console password
*************** Show

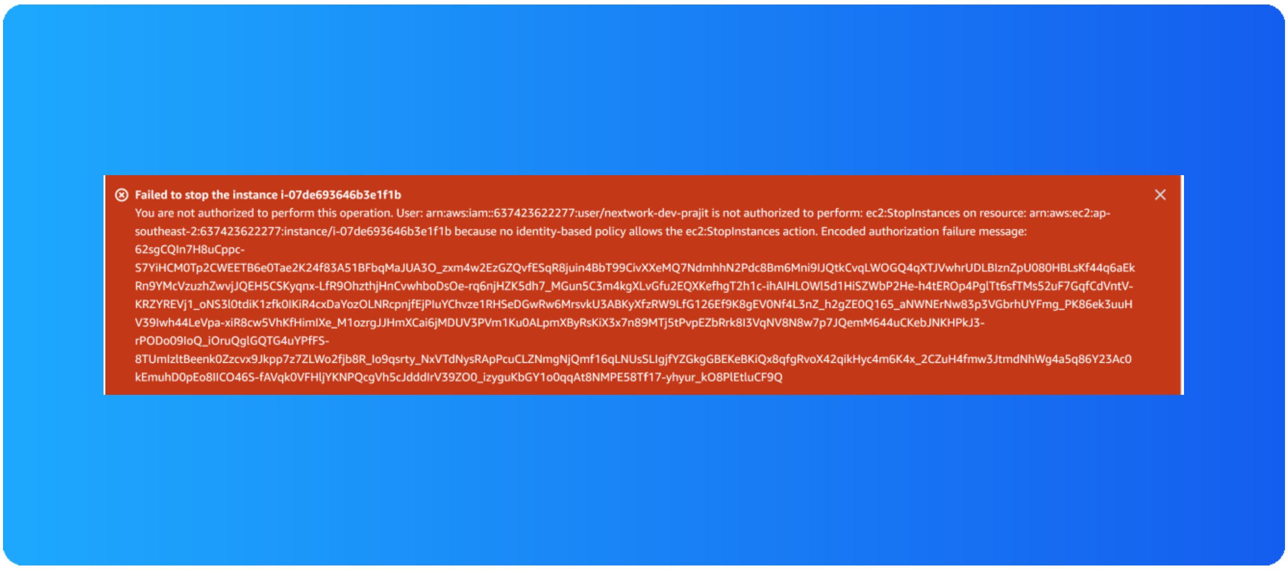
V Prajit Venkatachalam
NextWork Student

Testing IAM Policies

I tested my JSON IAM policy by trying to stop the production and development instances. (triggering the StopInstances action)

Stopping the production instance

When I tried to stop the production instance, a red banner appeared with an error message saying that I am not authorized to stop the production instance.



```
Failed to stop the instance i-07de693646b3e1f1b
You are not authorized to perform this operation. User: arn:aws:iam::637423622277:user/nextwork-dev-prajit is not authorized to perform: ec2:StopInstances on resource: arn:aws:ec2:ap-
southeast-2:637423622277:instance/i-07de693646b3e1f1b because no identity-based policy allows the ec2:StopInstances action. Encoded authorization failure message:
62sgCQIn7H8uCppc-
S7YiHCM0Tp2CWEETB6e0Tae2K24f83A51BFbqMaJUA3O_zxm4w2EzGZQvfESqR8juin4BbT99CivXXeMQ7NdmhhN2Pdc8Bm6Mni9IJQtkCvqLWOGQ4qXTJVwhrUDLBIznZpU080HBLsKf44q6aEk
Rn9YMcVzuzhZwvjJQEH5CSKyqnx-LfR9OhzthjHnCvwhboDsOe-rq6njHZK5dh7_MGun5C3m4kgXLvGfu2EQXKefhgT2h1c-ihAIHLOWl5d1HiSZWbP2He-h4tEROp4PglTt6sfTMs52uF7GqfCdVntV-
KRZYREVj1_oNS3l0tdiK1zfk0IKiR4cxDaYozOLNRcpnjfEjPIuYChvze1RHSeDGwRw6MrsvkU3ABKyXfzRW9LfG126Ef9K8gEV0Nf4L3nZ_h2gZE0Q165_aNWNErNw83p3VGbrhUYFmg_PK86ek3uuH
V39lwh44LeVpa-xiR8cw5VhKfHimIXe_M1ozrgJJHmXCai6jMDUV3PVm1Ku0ALpmXByRsKiX3x7n89MTj5tPvpEZbRrk8I3VqNV8N8w7p7JQemM644uCKebJNKHPkJ3-
rPODo09IoQ_iOruQglGQTG4uYPfFS-
8TUmIzltBeenk0Zzcvx9Jkpp7z7ZLWo2fjb8R_Io9qsrty_NxVTdNysRApPcuCLZNmgNjQmf16qLNUsSLIgjfYZGkgGBEKeBKiQx8qfgRvoX42qikHyc4m6K4x_2CZuH4fmw3JtmdNhWg4a5q86Y23Ac0
kEmuhD0pEo8IICO46S-fAVqk0VFHljYKNPQcgVh5cJdddIrV39ZO0_izyguKbGY1o0qqAt8NMPE58Tf17-yhyur_kO8PlEtluCF9Q
```

V

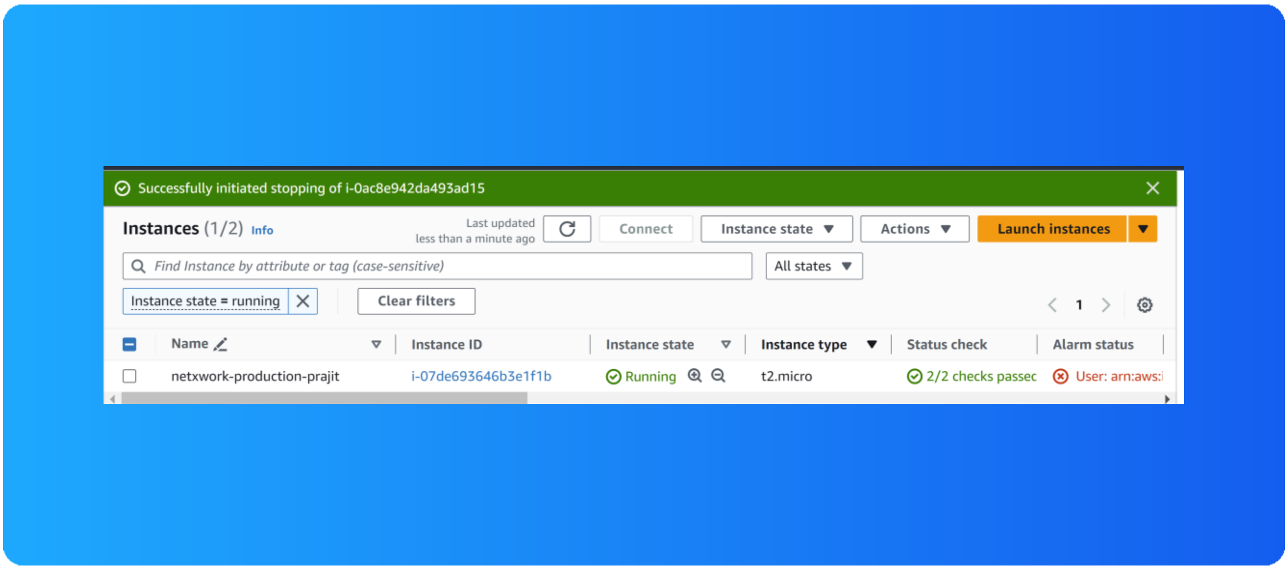
v.prajit98@gmail.com
NextWork Student

NextWork.org

Testing IAM Policies

Stopping the development instance

Next, when I tried to stop the development instance, it was successfully stopped. This was because the policy I created and attached to the user group allowed all EC2 actions for instances with the "Env: development" tag.



Successfully initiated stopping of i-0ac8e942da493ad15

Instances (1/2)

Instance state = running

☐	Name	Instance ID	Instance state	Instance type	Status check	Alarm status
☐	netxwork-production-prajit	i-07de693646b3e1f1b	✓ Running	t2.micro	✓ 2/2 checks passec	⊗ User: arn:aws:i

NextWork.org

Everyone should be in a job they love.

Check out nextwork.org for more projects