NEXTWORK

MANAGING PERMISSIONS WITH AWS IAM

Arijit Naskar

github.com/Arijitto

linkedin.com/in/arijit-naskar

WHAT IS AWS IAM?

What it does:

- AWS IAM securely controls access to AWS services/resources by managing users, groups, roles, and policies. It enhances security, provides granular access control, supports MFA, and simplifies centralized, scalable user management.

Why it’s useful:

- It enhances security by ensuring that only authorized users and services have the appropriate access to AWS resources, supporting compliance and reducing the risk of unauthorized actions.

How I’m using it in today’s project:

- In this project, I used IAM to: a. Create IAM Policy to restrict access to EC2 instance based on tags. b. Create IAM group that manage permissions efficiently. c. Create IAM user and assign them to user groups to grant specific permissions.

Arijit Naskar

github.com/Arijitto

linkedin.com/in/arijit-naskar

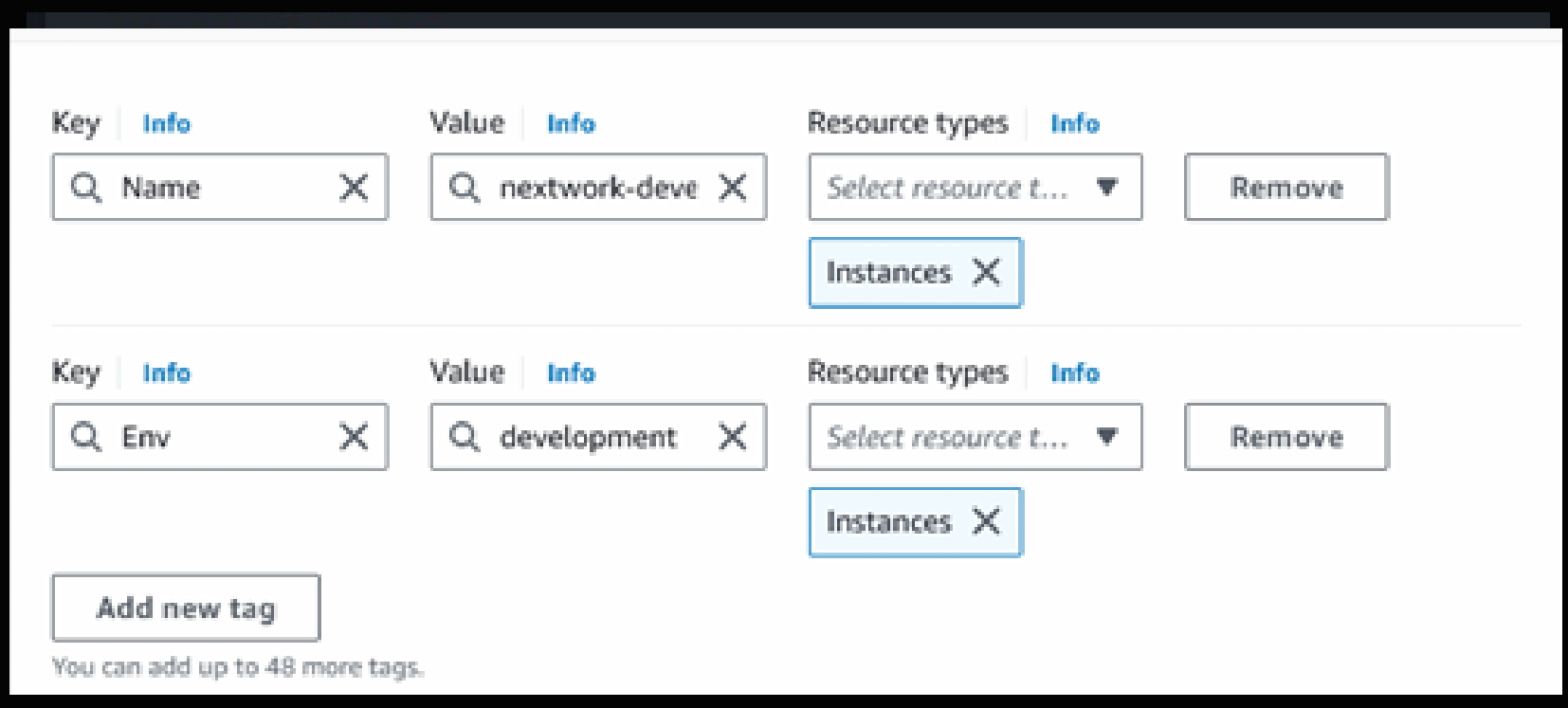
SETTING UP TAGS

- I've set up two EC2 instances to test the effectiveness of the permission settings I'll set up in AWS IAM. Ive used **tags** to label them.

- Tags in AWS EC2 are key-value pairs that you attach to instances to categorize and organize them. Tags help manage resources by enabling cost tracking, automation, access control, and easy resource discovery.

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

How the tags are set up for my EC2 instances



Arijit Naskar

github.com/Arijitto

linkedin.com/in/arijit-naskar

IAM POLICIES

- **IAM Policies** are rules which define who can do what with AWS resources.

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

- I've created a policy that all EC2 related actions to all EC2 instances that have Environment "Env" tag "deployment". But it also denies creating and deleting tags for ALL EC2 Instances.

- When creating JSON Policy, you have to specify the: Effect: Hold Allow or Deny. Indicate whether the policy allows or denies a certain action. Action: List of actions that the policy control Resource: Which resources does this policy apply to

The policy I've set up in the IAM Policies page!

Arijit Naskar

github.com/Arijitto

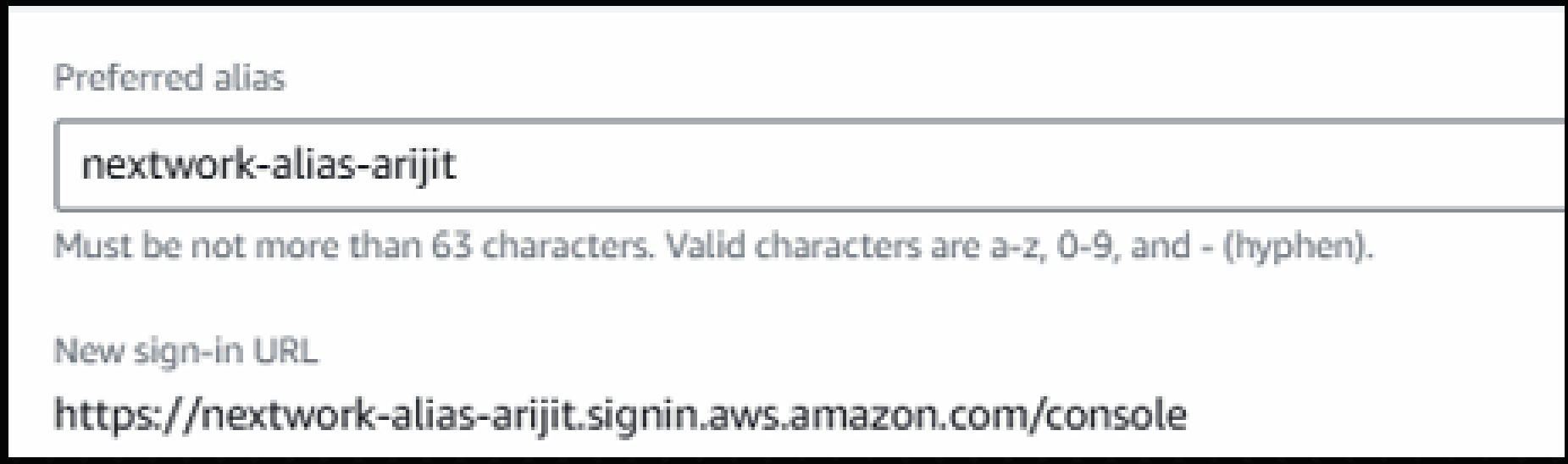
linkedin.com/in/arijit-naskar

AWS ACCOUNT ALIAS

- When new users get onboarded onto my AWS account, they get access by signing into a unique URL created for my account' Account ID.

- An account alias is a friendly name for your AWS account
- Creating an account alias took me a few seconds. Now, my new AWS console sign-in URL is https://nextwork-alias-arijit.signin.aws.amazon.com/console

You get to set up your own account alias name!



Arijit Naskar
github.com/Arijitto
linkedin.com/in/arijit-naskar

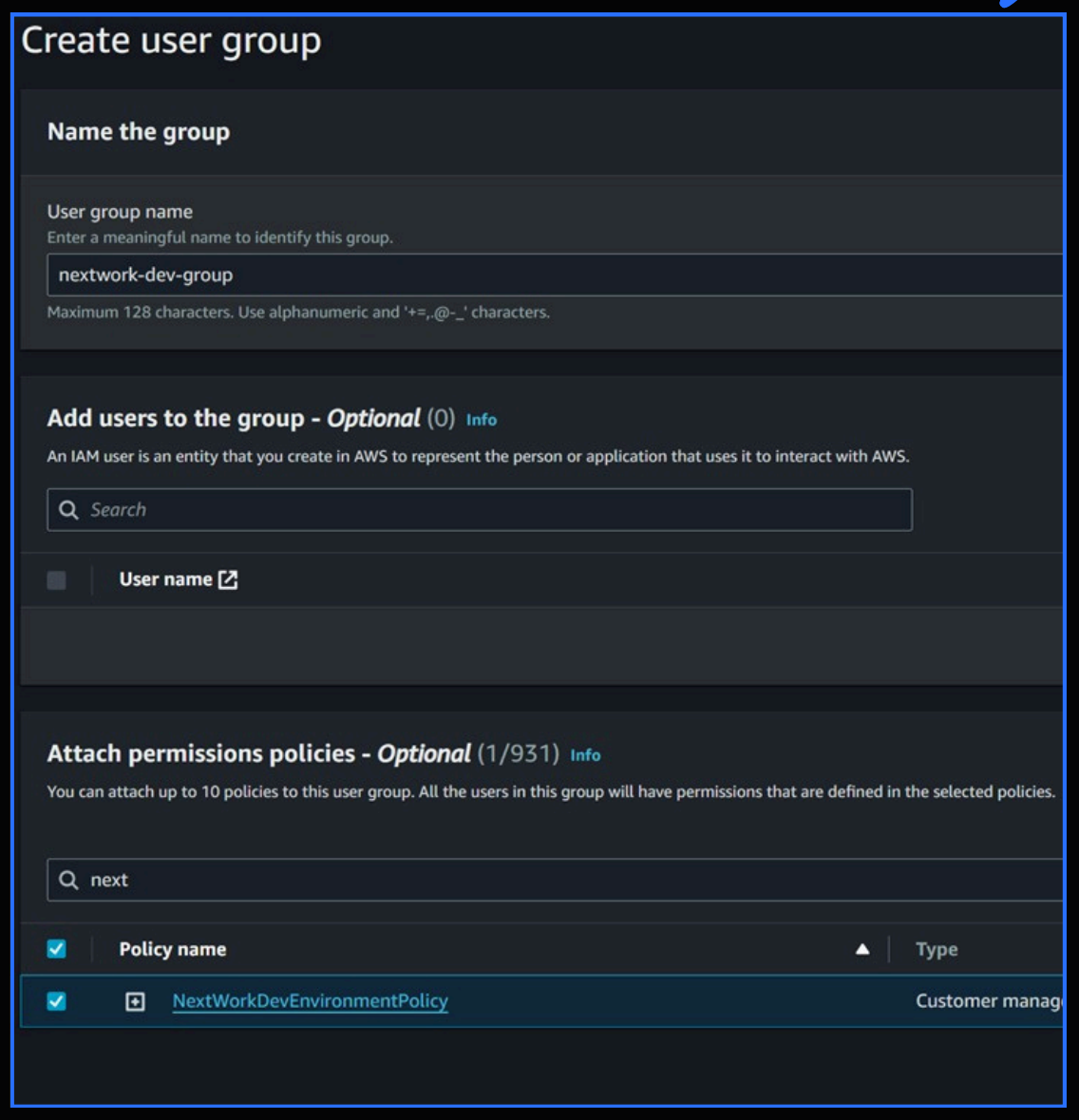
IAM USERS + USER GROUPS

- **IAM Users** are individual identities with specific credentials and permissions, allowing them to access and manage AWS resources.

- I also created a **User Group**. User Groups are useful for simplifying the management of permissions by allowing you to assign and manage policies for multiple users collectively, ensuring consistent access control and easier administration.

My User Group is called nextwork-dev-group. I attached the Policy I created to this User Group, which means any users in this group will be controlled by the policy attached.

My User Group!

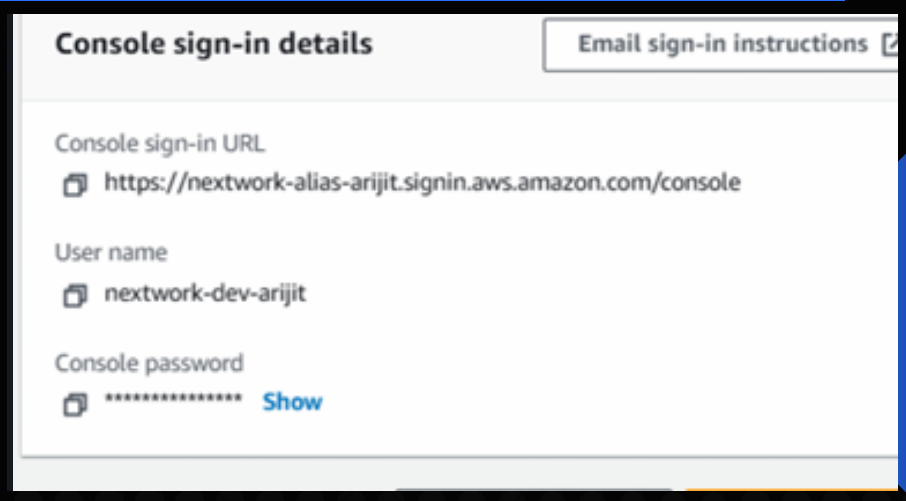


- When I created a new User, I had to tick the box that provided user access to the AWS Management Console.

- Once my new user was set up, there were two ways I could share its sign-in details:

- My new user had a unique sign-in URL - this is my Account Alias at work!

My User's sign-in details!



Arijit Naskar
github.com/Arijitto
linkedin.com/in/arijit-naskar

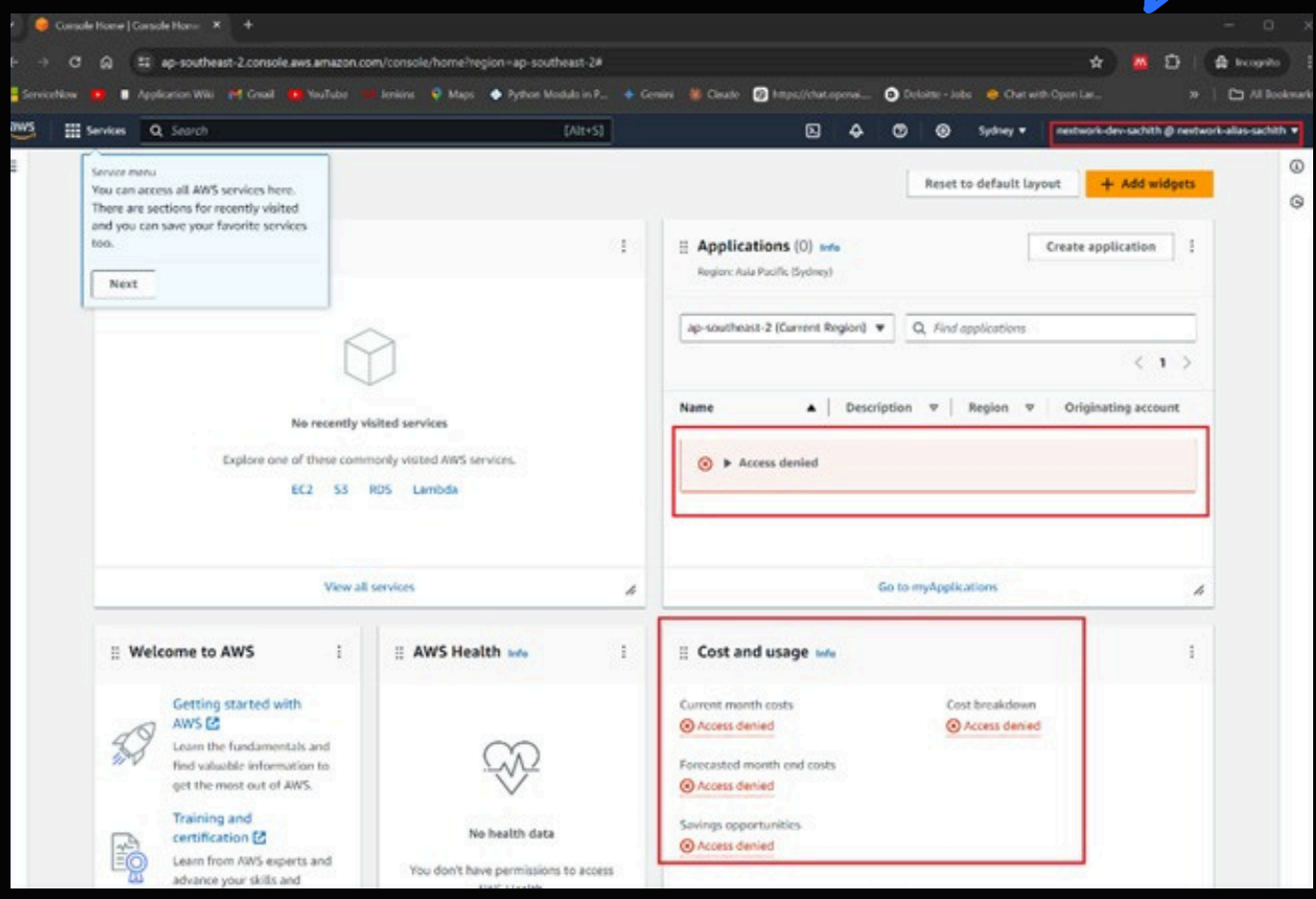
IAM USER IN ACTION

- Now with my IAM Policy, IAM User Group and IAM User all set up, lets put it all together! To do this, I logged into my AWS account as a new user.

- To log in as my IAM User, I use the URL given while creating the user.

- Once I logged in as my IAM user, noticed that a lot of panels displayed "Access denied". This was a clear difference to the dashboard I usually see in my AWS Account (Where I had unrestricted access to resources and wasn't denied access to anything)

Some of my dashboard's panels showed access denied!



Arijit Naskar

github.com/Arijitto

linkedin.com/in/arijit-naskar

IAM POLICIES IN ACTION

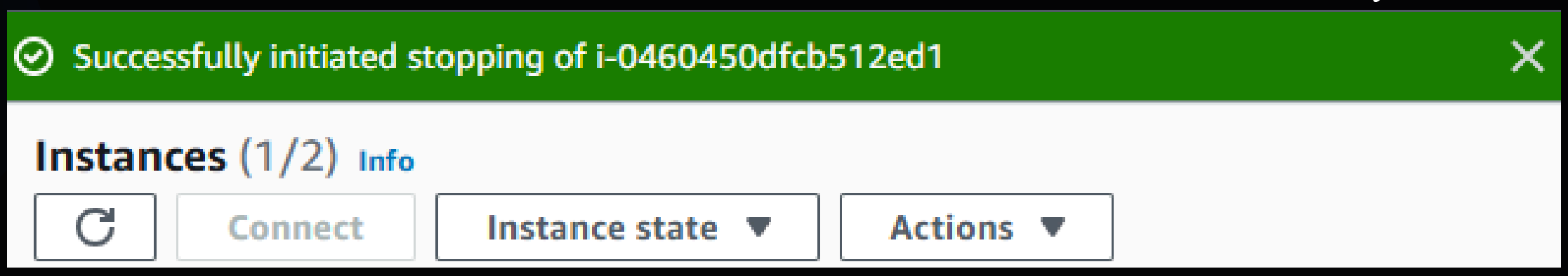
- Then, I tested the JSON IAM policy I set up by stopping the EC2 production instance.

- When I tried to stop the production instance, it failed. The message clearly stated that the user was not authorized for this action.

Woah! A red fail banner pops up if I stop the production instance

- Next, when I tried to stop the development instance, it stopped. Because we have given permission in policy to such actions with resources tagged as development.

Phew! A green success banner pops up if I stop the development instance



Arijit Naskar

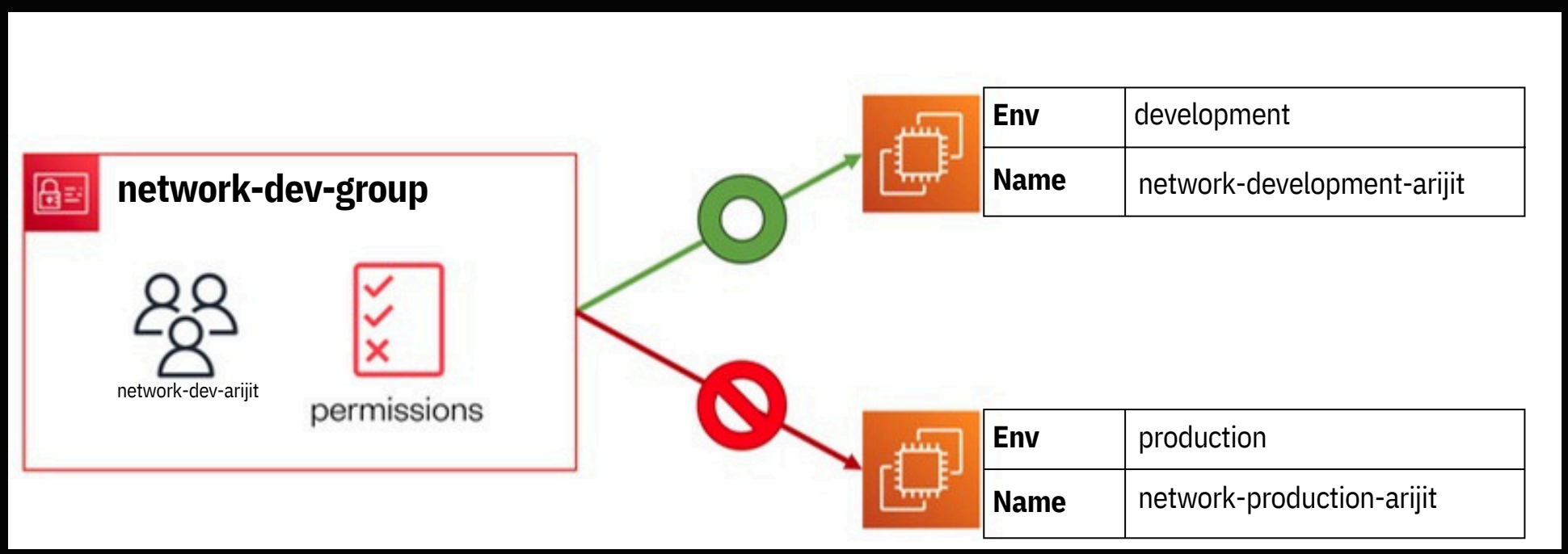
github.com/Arijitto

linkedin.com/in/arijit-naskar

TO SUMMARISE

I created :

- An IAM User Group called **nextwork-dev-group** with defined permissions using an IAM Policy
- An IAM User called **nextwork-dev-sachith** that is added to the user group
- An EC2 instance with the Env tag **development** and Name **nextwork-development-sachith**
- An EC2 instance with the Env tag **production** and Name **nextwork-production-sachith**



Arijit Naskar

github.com/Arijitto

linkedin.com/in/arijit-naskar

My Key Learnings

01 What are IAM Policies? IAM Policies are documents that define permissions to specify what actions are allowed or denied on AWS resources.

02 What are IAM Users? Why would you create one? IAM Users are individual identities with specific credentials, created to provide unique access and manage permissions for individual users in AWS.

03 What are IAM User Groups? Why would you create one? IAM User Groups

04 are collections of users that simplify permissions management by allowing you to apply policies to multiple users at once.

05 What is an AWS Account Alias? An AWS Account Alias is a user-friendly name that you can set to replace the default AWS account ID in the web address for your AWS Management Console.

I learned about using tags in EC2 to organize and manage resources effectively through categorization and cost tracking.

Arijit Naskar

github.com/Arijitto

linkedin.com/in/arijit-naskar

Final Thoughts...

- This project took me less than 1 hour to complete.
- Delete EVERYTHING at the end! Let’s keep this project free :)

- One thing I didn’t expect was for Policy descriptions to be validated for special characters.

- Now that I know how IAM could be used to enhance security and permissions in my AWS account, some real-world use cases of what I’ve learnt are:
    - Creating different IAM Users for my personal AWS account to enhance security when I do personal projects that will enable public access to my account’s resources.
    - Creating user groups for different company departments e.g. marketing, finance, development
    - Using an AWS Account Alias to create a user friendly console log in URL for a company’s AWS account.]

Arijit Naskar

github.com/Arijitto

linkedin.com/in/arijit-naskar

Find this helpful?

Like this post

Leave a comment

Save for later

Let’s connect!

pssst... if you want to get this free project guide and documentation template, **check out NextWork!**

Arijit Naskar

github.com/Arijitto

linkedin.com/in/arijit-naskar

Thanks NextWork for the free project guide!

NEXTWORK