



Cloud Security with AWS IAM



Clive Maduke

Policy editor

Visual | **JSON**

```
1▼ {
2    "Version": "2012-10-17",
3▼   "Statement": [
4▼     {
5         "Effect": "Allow",
6         "Action": "ec2:*",
7         "Resource": "*",
8▼         "Condition": {
9▼             "StringEquals": {
10                "ec2:ResourceTag/Env": "development"
11            }
12        }
13    },
14▼    {
15        "Effect": "Allow",
16        "Action": "ec2:Describe*",
17        "Resource": "*"
18    },
19▼    {
20        "Effect": "Deny",
21▼        "Action": [
22            "ec2:DeleteTags",
23            "ec2:CreateTags"
24        ],
25        "Resource": "*"
26    }
27  ]
```



Clive Maduke

NextWork Student

nextwork.org

Introducing Today's Project!

Project overview

In this project, I will demonstrate how to use AWS IAM to control access and permission settings in my AWS account. I'm doing this project to learn about cloud security from absolute foundations- every company has to think about access permissions, and there are even entire careers or jobs called IAM Engineers' focused on the skills I am demonstrating on this project today.

Tools and concepts

Services I used were Amazon EC2 and AWS IAM Key concepts I learnt include IAM users, policies, user groups and account aliases. I also learnt how to use policy simulator and how JSON policies work How to launch and tag an instance how to log in as another user

Project reflection

This project took me approximately 1 hours... The most challenging part was trying to figure out policies ... It was most rewarding was being able to see how the policies that were put in place will work for the user from using their account to using the Policy simulator to...



Clive Maduke

NextWork Student

nextwork.org

Tags

What I did in this step

In this step, I will be launching two EC2 instances because they will be needed to boost NextWork's computing power because NextWork will be expecting an influx of traffic over the Summer break!

Understanding tags

Tags are organisational tools that let us label resources. They are useful for grouping resources, cost allocation and applying policies for all resources with the same tag

My tag configuration

The tag I have used on my EC2 instances is called Environment. The values I've assigned for my instances are production and development.



Clive Maduke
NextWork Student

nextwork.org

The screenshot shows the 'Launch an instance' wizard in the AWS Management Console. The current step is 'Name and tags'. Two tags are defined:

- Name**: Value `nextwork-dev-clivem`, Resource types: `Select resource types`, Instances: `Instances`
- Env**: Value `development`, Resource types: `Select resource types`, Instances: `Instances`

An 'Add new tag' button is available. A note indicates you can add up to 48 more tags.

Application and OS Images (Amazon Machine Image)

An AMI contains the operating system, application server, and applications for your instance. If you don't see a suitable AMI below, use the search field or choose [Browse more AMIs](#).

Search our full catalog including 1000s of application and OS Images



Clive Maduke

NextWork Student

nextwork.org

IAM Policies

What I did in this step

In this step, I 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 environment) but NOT the production environment.

Understanding IAM policies

IAM Policies are rules that govern who can do what in our account. I will be using policies today to control who has access to our production or environment instance.

The policy I set up

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

Policy effect

I've created a policy that allow the holder ie the intern to have permission to do anything they want to any instance tagged with "development". They can also see informationfor any instance, but there are denied access to deleting or creating tags for any instance

Understanding Effect, Action, and Resource

The Effect, Action, and Resource attributes of a JSON policy means wheather or not the policy is allowing or denying action (ie Effect); what the holder can or cannot do (i.e. action) and the specific AWS resources that the policy relates to (i.e resource)

Clive Maduke

NextWork Student

nextwork.org

My JSON Policy

Policy editor

Visual | **JSON**

```
1▼ {
2    "Version": "2012-10-17",
3▼   "Statement": [
4▼     {
5        "Effect": "Allow",
6        "Action": "ec2:*",
7        "Resource": "*",
8▼       "Condition": {
9▼         "StringEquals": {
10            "ec2:ResourceTag/Env": "development"
11          }
12        }
13      },
14▼     {
15        "Effect": "Allow",
16        "Action": "ec2:Describe*",
17        "Resource": "*"
18      },
19▼     {
20        "Effect": "Deny",
21        "Action": [
22          "ec2:DeleteTags",
23          "ec2:CreateTags"
24        ],
25        "Resource": "*"
26      }
27    ]
```



Clive Maduke

NextWork Student

nextwork.org

Account Alias

What I did in this step

In this step, I will set up an Account Alias, which is like a nickname for our AWS console's login. This because it makes it simple for the users logging easily.

Understanding account aliases

An account alias is a nick name for an AWS account instead of a long account ID, which means now I can reference my account instead.

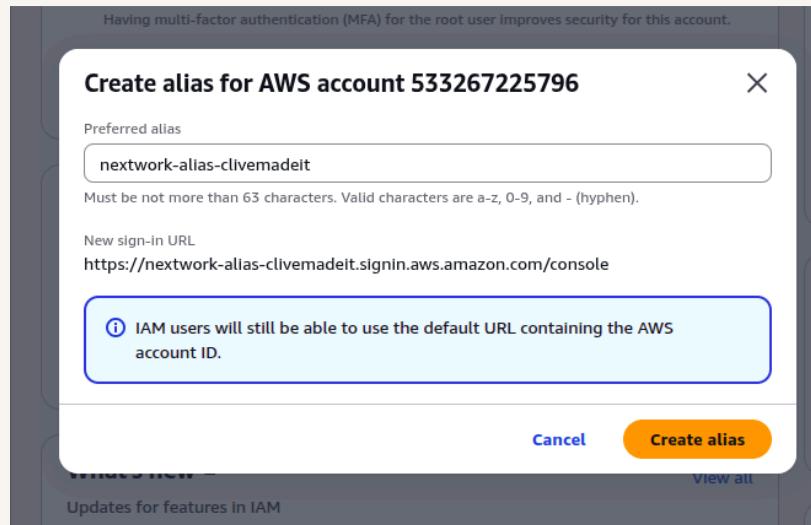
Setting up my account alias

Creating an account alias took me less than a minute it was the simplest configuration. Now, my new AWS console sign-in URL uses the alias that I have created instead of my account ID

Clive Maduke

NextWork Student

nextwork.org





Clive Maduke

NextWork Student

nextwork.org

IAM Users and User Groups

What I did in this step

In this step, I will set up two dedicated IAM resources IAM users, and IAM user groups. This is because IAM users are like logins for people that want access to our AWS account, while user groups are like folders to manage users that have the same level of access.

Understanding user groups

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

Attaching policies to user groups

I attached the policy I created to this user group, which means any user created will automatically get the permissions our NextWorkDevEnvironment policy IAM Policy

Understanding IAM users

IAM users are people or entities that have or entities login to our AWS account.



Clive Maduke

NextWork Student

nextwork.org

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 to download the .csv file with the sign details inside.

Observations from the IAM user dashboard

Once I logged in as my IAM user, I noticed that the user is already denied access to panels on the main AWS console dashboard. This was because the permissions have been set up to our development EC2 instance, so our intern will not be able to have access to even see anything else



Clive Maduke
NextWork Student

nextwork.org

User created successfully

You can view and download the user's password and email instructions for signing in to the AWS Management Console.

[View user](#)

Step 1
Specify user details
Step 2
Set permissions
Step 3
Review and create
Step 4
Retrieve password

Retrieve password

You can view and download the user's password below or email users instructions for signing in to the AWS Management Console. This is the only time you can view and download this password.

Console sign-in details

Console sign-in URL
 <https://clivethatmadelt.signin.aws.amazon.com/console>

User name
 nextwork-dev-clivemadelt

Console password
 ***** [Show](#)

[Email sign-in instructions](#)

[Cancel](#) [Download .csv file](#) [Return to users list](#)



Clive Maduke

NextWork Student

nextwork.org

Testing IAM Policies

What I did in this step

In this step, I will login to own AWS account as the intern and test access to the production and development instances because I have to make sure the intern has been given the right permissions and that they can not affect our production environment.

Testing policy actions

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 encountered an error. This was because our production instance is tagged with the production label which is out of the scope of our permission policy. Interns are only allowed to do things to the development instances.



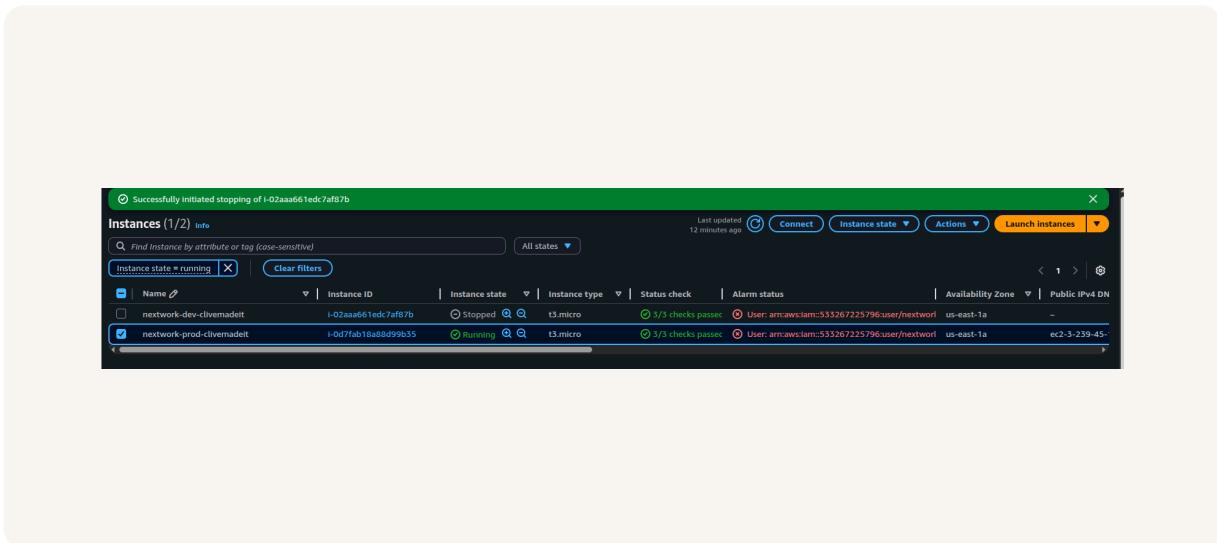
Clive Maduke
NextWork Student

nextwork.org



Stopping the development instance

Next, when I tried to stop the development instance the stoppage was successfull as the instance state changed to stopping and then finally stopped. This this is due to the intern having permission over at the development instance.





Clive Maduke

NextWork Student

nextwork.org

IAM Policy Simulator

To extend my project, I'm going to test our permissions policies in safer and more controlled way using a tool called IAM poliy si,uater ... I'm doing this because... having to change accounts and stopping resources can be disruptive to other users

Understanding the IAM Policy Simulator

The IAM Policy Simulator is a tool that lets one simulat actions and test permission settings by defining a specific user, group or role and action we want to test for ... It's useful for saving time because one does not need to log in as another user to test out the permissions or stopping resources...

How I used the simulator

I set up a simulation for whether our users can access certain permissions to stop instances and delete tags. The results were.. denied for both. I had to adjust the scope to EC2 instances to once that are tagged with development. Once we applied that tag, permission was allowed



Clive Maduke
NextWork Student

nextwork.org

Policy Simulator

Amazon EC2 ▾ 2 Action(s) sele... Select All Deselect All Reset Contexts Clear Results Run Simulation

▶ Global Settings ⓘ

Action Settings and Results [2 actions selected. 0 actions not simulated. 1 actions allowed. 1 actions denied.]

Service	Action	Resource Type	Simulation Resource	Permission
Amazon EC2	StopInstances	instance	*	allowed 1 matching statements.

Show statement in NextWorkDevEnvironmentPolicy (IAM Policy)

Resource You can specify the resource and context keys used to simulate this action. By default the simulation resource is "*".

instance	*
	ec2:resourcetag/env
	development



nextwork.org

The place to learn & showcase your skills

Check out nextwork.org for more projects

