Importing AWS resources to Terraform

Today we’ll convert an existing CloudWatch Alarm into an easily manageable Terraform manifest.

Let’s look at a company that has relied solely on manually created monitoring for their company website, which is being hosted on CloudFront. We want this monitoring to be completely similar between environments, we want it to be scalable, and we want to be able to use a code repository so that we can audit any changes made to the monitoring resources.

The employee has created a CloudWatch alarm called ‘test’ that monitors the total error rate on our CloudFront distribution in the console. When the SRE team asks them to tweak the alarm, they log into the console, make the changes, and apply them by hand. One time, they accidentally changed the alarm condition from GreaterThan to LessThan. This triggered alarms and a page to the on-call engineer. We will help them import this alarm so it can be managed with Terraform. Ensuring that changes are double checked with a PR and can easily be reverted in case of any problems.

A screenshot of a computer

Description automatically generated

This is the manually created CloudWatch alarm

We will start by creating a Terraform manifest file called main.tf. Inside this file we will define the CloudWatch alarm using the parameters that we see in the manually created alarm.

A screen shot of a computer

Description automatically generated

This is our Terraform manifest that defines the alarm

Then we’ll import the existing alarm into the Terraform state file. This avoids the danger of a lapse or duplicate alarm coverage while performing the migration to TF, because we’re simply taking over the management of the alarm.

The command is:

**terraform import aws\_cloudwatch\_metric\_alarm.foo test**

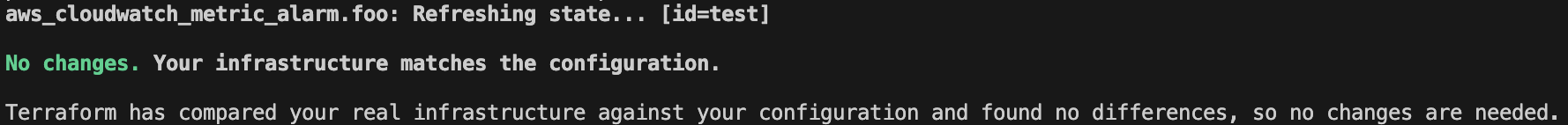
A screen shot of a computer

Description automatically generated

Now we can run the Terraform Development Workflow which is

**terraform init**

**terraform plan**

****

Terraform is now managing this alarm. If changes are required. We’ll be making them by updating the Terraform manifest and publishing them to the repository that holds our Terraform code. This repo can be called something like, company-website-monitoring-terraform, and will contain all of our CloudWatch alarms.

For example, if we want to change the threshold for this alarm from 50% to 25% errors, we simply change the value of **threshold** to .25. We push these changes to our code repository and run it through our Terraform workflow to see the plan, and to apply the changes.

A screenshot of a computer

Description automatically generated