07 AWS Management

July 2, 2020

1 Logging

Logging provides visibility into your cloud resources and applications. Logging and auditing services help proactively monitor your resources and application

Logging allows you to answer important questions: - How is the server performing - What is the current load on the server - What is the root cause of an application error your seeing - What is the path that leads to this error

2 Cloud Trail

- Cloud Trail allows you to audit or monitor everything on your AWS account
- Logs actions through AWS management console and AWS SDK
- means SDK, command line tools, AWS services are loged

3 Cloud Watch

Cloud watch is a service that monitors resources and applications that run on AWS by collecting data in the form of logs, metrics and events

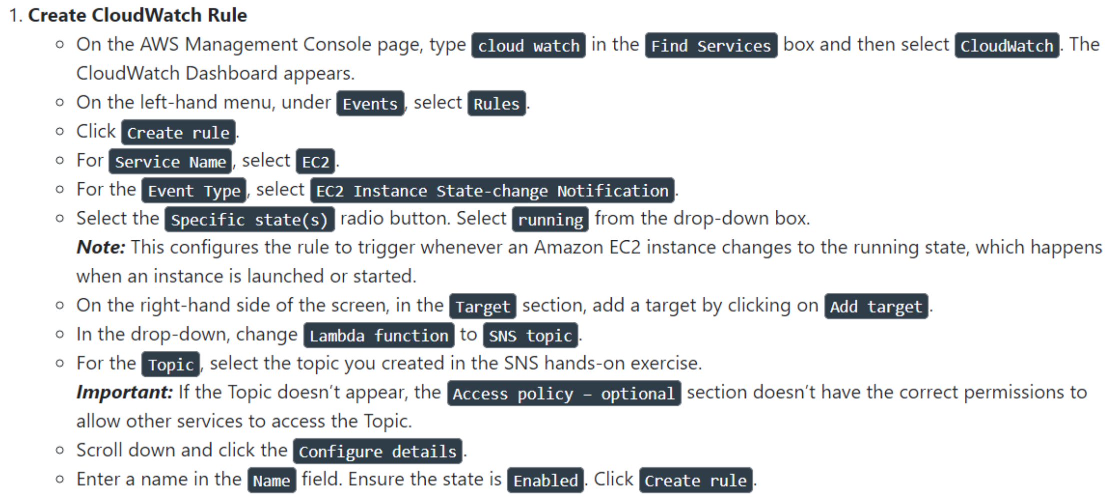
3.1 Features

- Collect and track metrics
- Collect and monitor log files
- Set alarms and create triggers to run your AWS resources
- React to changes in your AWS resources

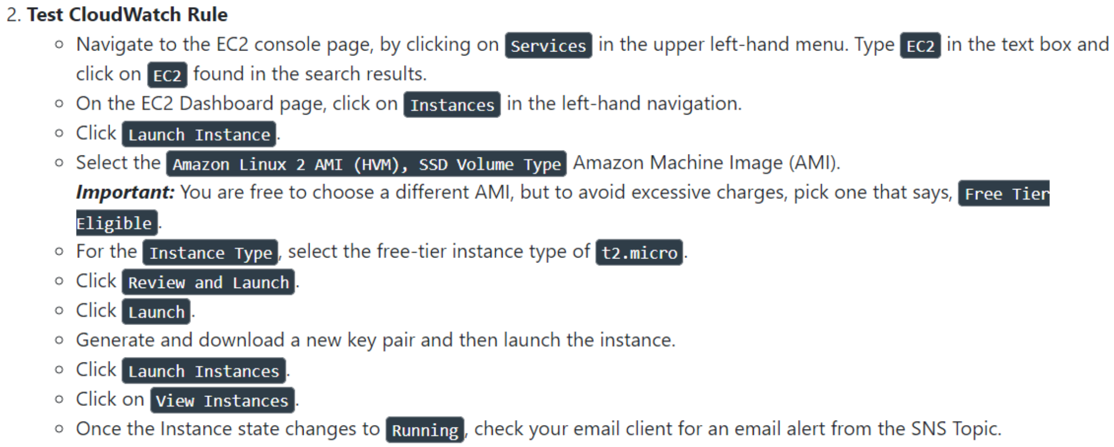
4 Uses

- use CloudWatch logs written from lambda functions to diagnose and monitor issues and application flow
- Also use CloudWatch as a trigger for lambda function

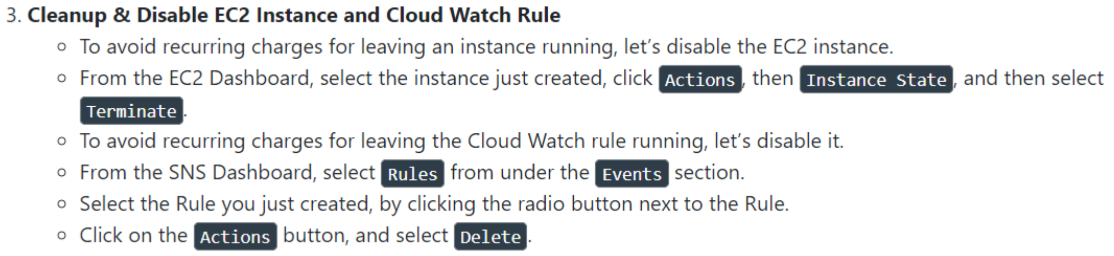
5 Cloud Watch Lab



1. **Create CloudWatch Rule**
   - On the AWS Management Console page, type `cloud watch` in the `Find Services` box and then select `CloudWatch`. The CloudWatch Dashboard appears.
   - On the left-hand menu, under `Events`, select `Rules`.
   - Click `Create rule`.
   - For `Service Name`, select `EC2`.
   - For the `Event Type`, select `EC2 Instance State-change Notification`.
   - Select the `Specific state(s)` radio button. Select `running` from the drop-down box.
     Note: This configures the rule to trigger whenever an Amazon EC2 instance changes to the running state, which happens when an instance is launched or started.
   - On the right-hand side of the screen, in the `Target` section, add a target by clicking on `Add target`.
   - In the drop-down, change `Lambda function` to `SNS topic`.
   - For the `Topic`, select the topic you created in the SNS hands-on exercise.
     Important: If the Topic doesn't appear, the `Access policy - optional` section doesn't have the correct permissions to allow other services to access the Topic.
   - Scroll down and click the `Configure details`.
   - Enter a name in the `Name` field. Ensure the state is `Enabled`. Click `Create rule`.



2. **Test CloudWatch Rule**
   - Navigate to the EC2 console page, by clicking on `Services` in the upper left-hand menu. Type `EC2` in the text box and click on `EC2` found in the search results.
   - On the EC2 Dashboard page, click on `Instances` in the left-hand navigation.
   - Click `Launch Instance`.
   - Select the `Amazon Linux 2 AMI (HVM), SSD Volume Type` Amazon Machine Image (AMI).
     Important: You are free to choose a different AMI, but to avoid excessive charges, pick one that says, `Free Tier Eligible`.
   - For the `Instance Type`, select the free-tier instance type of `t2.micro`.
   - Click `Review and Launch`.
   - Click `Launch`.
   - Generate and download a new key pair and then launch the instance.
   - Click `Launch Instances`.
   - Click on `View Instances`.
   - Once the Instance state changes to `Running`, check your email client for an email alert from the SNS Topic.



3. **Cleanup & Disable EC2 Instance and Cloud Watch Rule**
   - To avoid recurring charges for leaving an instance running, let's disable the EC2 instance.
   - From the EC2 Dashboard, select the instance just created, click `Actions`, then `Instance State`, and then select `Terminate`.
   - To avoid recurring charges for leaving the Cloud Watch rule running, let's disable it.
   - From the SNS Dashboard, select `Rules` from under the `Events` section.
   - Select the Rule you just created, by clicking the radio button next to the Rule.
   - Click on the `Actions` button, and select `Delete`.

6 Infrastructure As Code

- Infrastructure as Code allows you to describe and provision all the infrastructure resources in your cloud envoirment
- You can stand up servers, databaases, runtime parameters, resources, etc based on scripts you write
- IAS is a time-saving feature because it allws you to provision (or stand up) resources in a reproducible way

6.1 How It Works

- Scripts out infrastructure
- which makes infrastructure into code
- you can manage a collection of related resources and treat them as one logical unit

7 Logical Unit

- Imagine you had to do the following
  - Configure a VPC security group
  - Launch an EC2 Instance
  - Create load balancers
  - create an RDS instance
  - Create AutoScaling
- Instead of manually doing each of these things, we can write scripts to do it

8 Cloud Formation

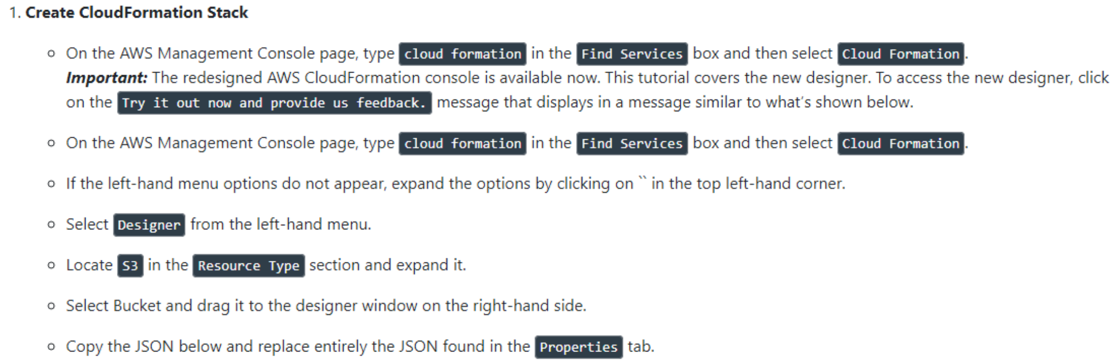
Awa Cloud Formation allows you to model your entire infrastructure in a text file template allowing you to provision AWS resources based on the script you write

8.1 Tips

- Cloud formation is found under the Management and Governance section on AWS
- Cloud Formation templates are written using JSON or YAML
- You can still individually manage AWS resources that are part of a CloudFormation stack

```
{"AWSTemplateFormatVersion":"2010-0909",
    "Description": "...",
    "Paramaters": {
        "Vpcid":{
            "Types":"AWS::EC2::VPC::id",
            "Description":"...".
            "ConstraintDescription":"...".
            }
      }
}
```

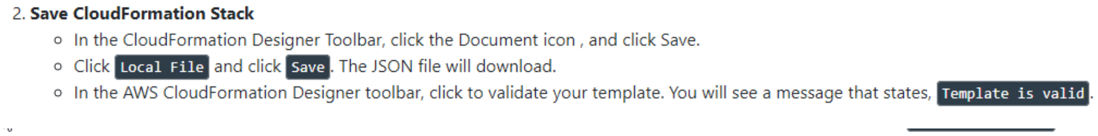
9 Cloud Formation Lab



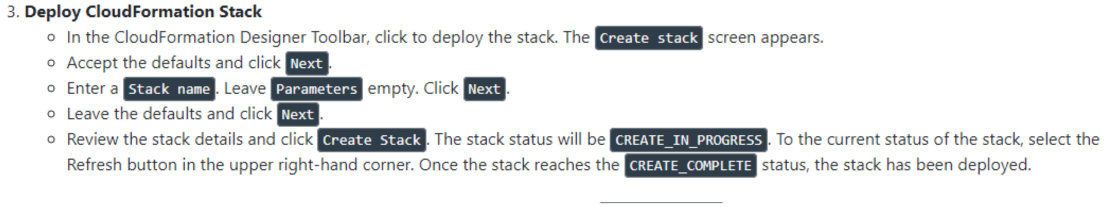
1. **Create CloudFormation Stack**
   - On the AWS Management Console page, type `cloud formation` in the `Find Services` box and then select `Cloud Formation`.
     Important: The redesigned AWS CloudFormation console is available now. This tutorial covers the new designer. To access the new designer, click on the `Try it out now and provide us feedback.` message that displays in a message similar to what's shown below.
   - On the AWS Management Console page, type `cloud formation` in the `Find Services` box and then select `Cloud Formation`.
   - If the left-hand menu options do not appear, expand the options by clicking on `` in the top left-hand corner.
   - Select `Designer` from the left-hand menu.
   - Locate `S3` in the `Resource Type` section and expand it.
   - Select Bucket and drag it to the designer window on the right-hand side.
   - Copy the JSON below and replace entirely the JSON found in the `Properties` tab.

```
{
  "AWSTemplateFormatVersion": "2010-09-09",
  "Description": "Basic S3 Bucket CloudFormation template",
  "Resources": {
    "S3BucketCreatedByCloudFormation": {
      "Type": "AWS::S3::Bucket",
      "Properties": {
        "AccessControl": "PublicRead"
      }
    }
  },
  "Outputs": {
    "BucketName": {
      "Value": {
        "Ref": "S3BucketCreatedByCloudFormation"
      },
  "Description": "Name of the newly created Amazon S3 Bucket"
    }
  }
}
```

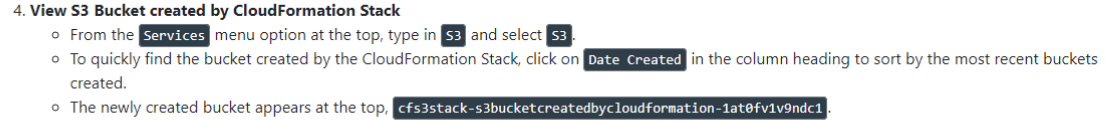
Hit the Refresh button in the upper right-hand corner so that the Designer is not out of date.



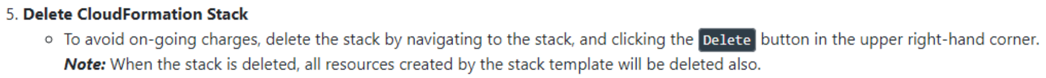
2. **Save CloudFormation Stack**
   - In the CloudFormation Designer Toolbar, click the Document icon , and click Save.
   - Click `Local File` and click `Save`. The JSON file will download.
   - In the AWS CloudFormation Designer toolbar, click to validate your template. You will see a message that states, `Template is valid`.



3. **Deploy CloudFormation Stack**
   - In the CloudFormation Designer Toolbar, click to deploy the stack. The `Create stack` screen appears.
   - Accept the defaults and click `Next`.
   - Enter a `Stack name`. Leave `Parameters` empty. Click `Next`.
   - Leave the defaults and click `Next`.
   - Review the stack details and click `Create Stack`. The stack status will be `CREATE_IN_PROGRESS`. To the current status of the stack, select the Refresh button in the upper right-hand corner. Once the stack reaches the `CREATE_COMPLETE` status, the stack has been deployed.



4. **View S3 Bucket created by CloudFormation Stack**
   - From the `Services` menu option at the top, type in `S3` and select `S3`.
   - To quickly find the bucket created by the CloudFormation Stack, click on `Date Created` in the column heading to sort by the most recent buckets created.
   - The newly created bucket appears at the top, `cfs3stack-s3bucketcreatedbycloudformation-1at0fv1v9ndc1`.



5. **Delete CloudFormation Stack**
   - To avoid on-going charges, delete the stack by navigating to the stack, and clicking the `Delete` button in the upper right-hand corner.
     Note: When the stack is deleted, all resources created by the stack template will be deleted also.

10 AWS CLI

The CLI allows you to access and control services running in your AWS account from the command line