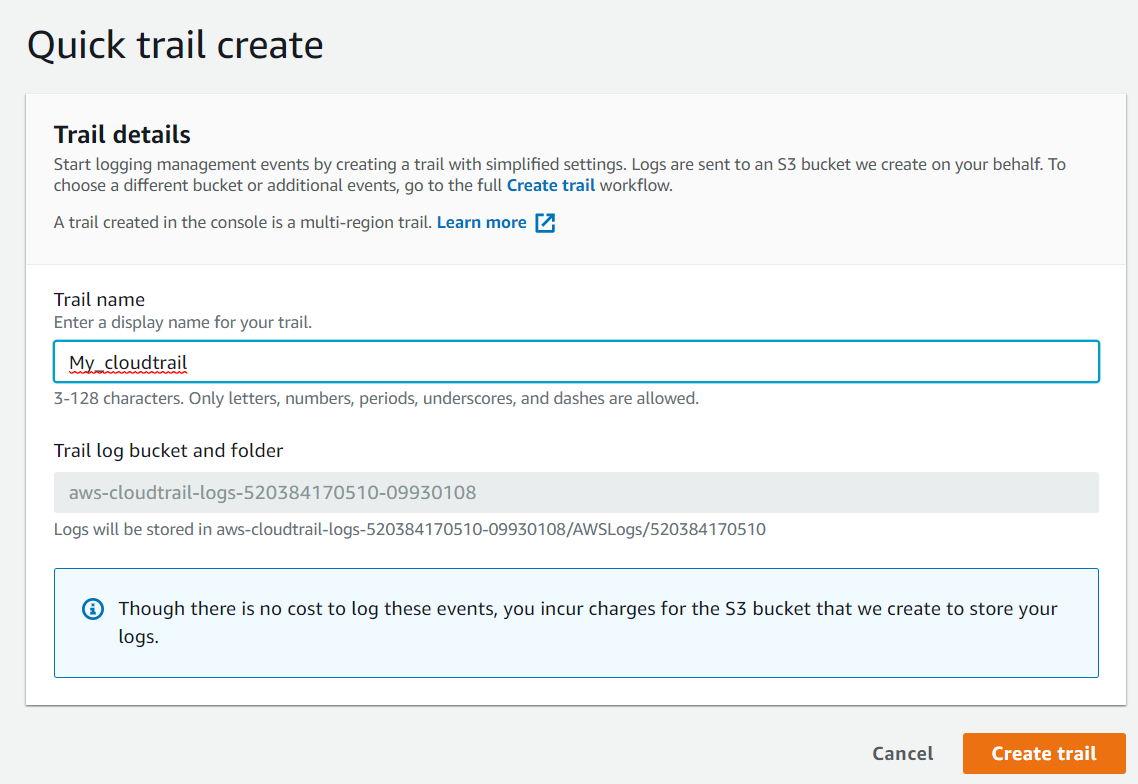
**Lab Steps**

Task 1: Sign in to AWS Management Console

1. Click on the  button, and you will get redirected to AWS Console in a new browser tab.
2. On the AWS sign-in page,
   * Leave the Account ID as default. Never edit/remove the 12 digit Account ID present in the AWS Console. otherwise, you cannot proceed with the lab.
   * Now copy your **User Name** and **Password** in the Lab Console to the **IAM Username and Password** in AWS Console and click on the **Sign in** button.
3. Once Signed In to the AWS Management Console, Make the default AWS Region as **US East (N. Virginia) us-east-1.**
4. Select Maybe later in New AWS Console Home page pop-up

Task 2: Creating a CloudTrail

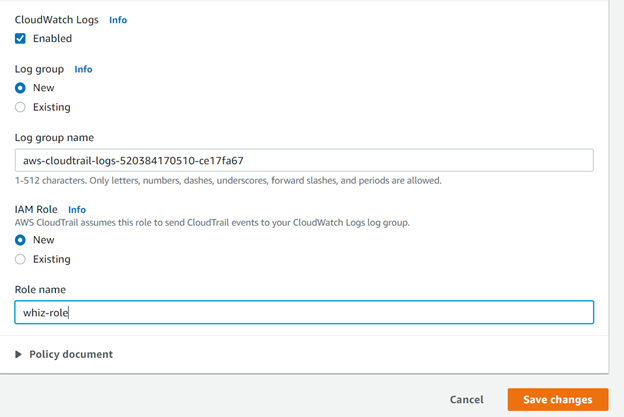
1. Make sure to choose the **N.Virginia**region in the AWS Management console dashboard (present in the top right corner).
2. Navigate and click on **CloudTrail,** which will be available under the  **Management & Governance** section of **Services**.
3. Click on **Trails** from the left sidebar and then Click on 
4. Under Create Trail, enter these details:
   * Trail name: Enter ***My\_cloudtrail***and click on **Create Trail** button.



5. Select the trail you have created, click on it. Scroll down you can see Cloudwatch logs section.

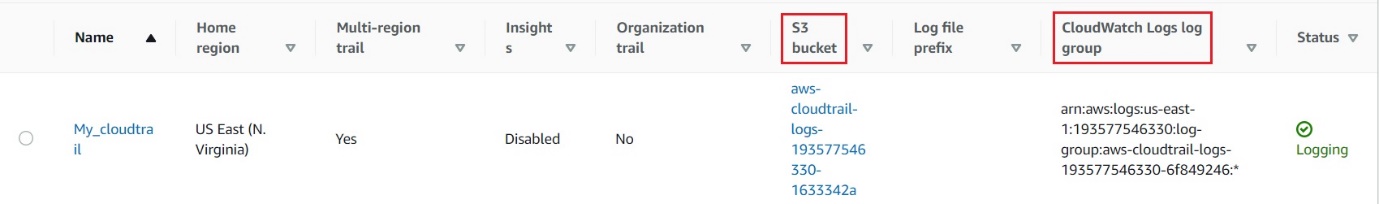
    Click on the **Edit** button.

* CloudWatch Logs  :  **Check**Enabled
  + Log group : Leave it as default (i.e New and default log group name)
  + IAM Role : Select **New** and give the Role name as ***whiz\_role.***
  + Click on **Save changes** button



* Tags-optional:
  + Key: Enter ***Name***
  + Value: Enter ***my\_logs***
* Click on **Next**button
* Choose Log Events:
  + Leave everything as default and click on **Next**.
* Review and click on .

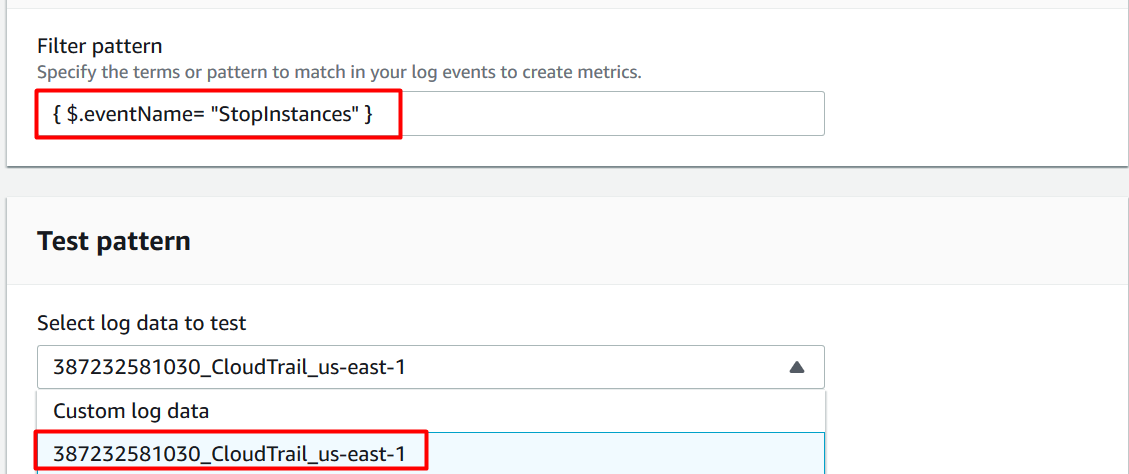
          ?6.    A CloudTrail instance that delivers logs to an S3 bucket has now been created.



Task 3: Creating Metric Filters for Log Groups in Cloudwatch

1. Make sure you are in the **N.Virginia**Region.
2. Click on **Services** and navigate to the **CloudWatch** dashboard under **Management & Governance**.
3. Click on  under**Logs**in the left panel.
4. Click on the log group we just created and click on the **Actions.**
5. Click on  as shown below:
6. Under **Create filter pattern,** provide the pattern you need to filter on. For this lab, we are **going to filter for stopped instances**.

* Filter pattern                 : Enter the pattern ***{ $.eventName= "StopInstances" }***
* Select log data to test  : Select the **cloudtrail log**  in drop-down.



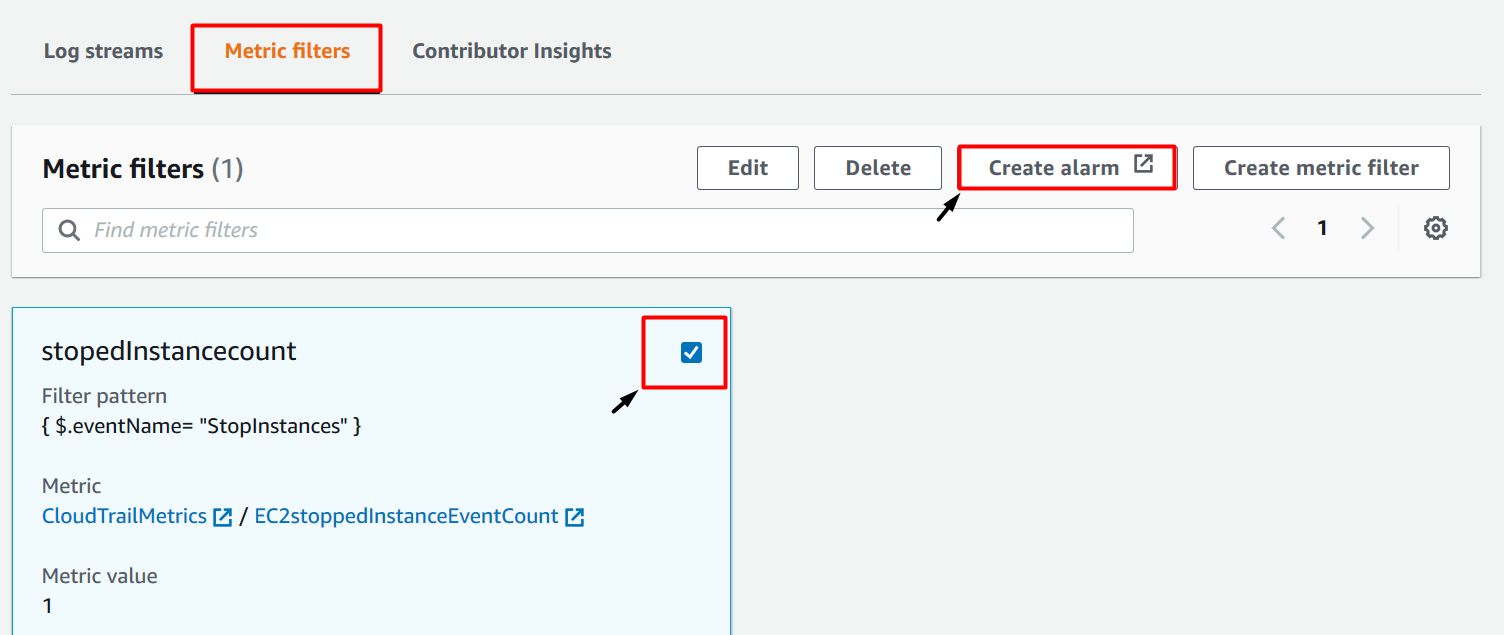
* After completing the above steps, click on .

1. Next we will create a filter using the following details:

* Filter name: Enter ***stoppedInstancecount***
* Metric details:
  + Metric namespace  : Enter ***CloudTrailMetrics***
  + Metric name            : Enter ***EC2stoppedInstanceEventCount***
  + Metric value            : Enter ***1***
  + Default value           : Leave default
* Finally, click on  and review the given details. Click on **Create metric filter** to complete the metric filter creation.

Task 4: Creating an Alarm

1. In CloudWatch, select the log group created for our CloudTrail and then click on  at the bottom.
2. Select the Metric filter created in the above step and then click on  as shown below:

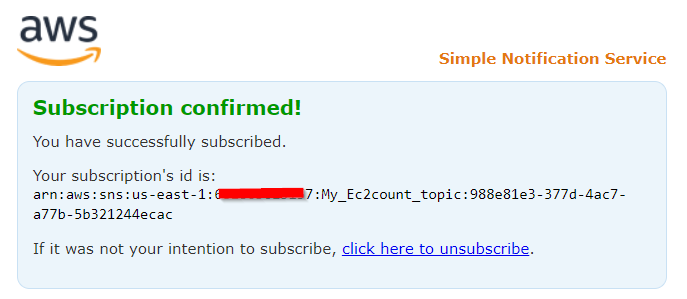


1. Specify the metric conditions as follows:

* Namespace : **CloudTrailMetrics (default)**
* Metric name : **EC2stoppedInstanceEventCount (default)**
* Statistic        : **sum (default)**
* Period          : **5 minute (default)**
* Conditions:
  + Threshold type: Select **Static**
  + Whenever EC2stoppedInstanceEventCount is :    than ***1***.
* Click on .

1. Next we'll **configure actions**

* Alarm state trigger                                              : Select 
* Select an SNS topic                                            : select **Create new topic**
* Create a new topic                                              : Enter the topic name as ***My\_Ec2count\_topic***
* Email endpoints that will receive the notification :  Enter your Email address to receive the alert
* Once you provide these details, click on .
  + AWS will send a confirmation email to the Email address provided above. You will need to confirm the email subscription

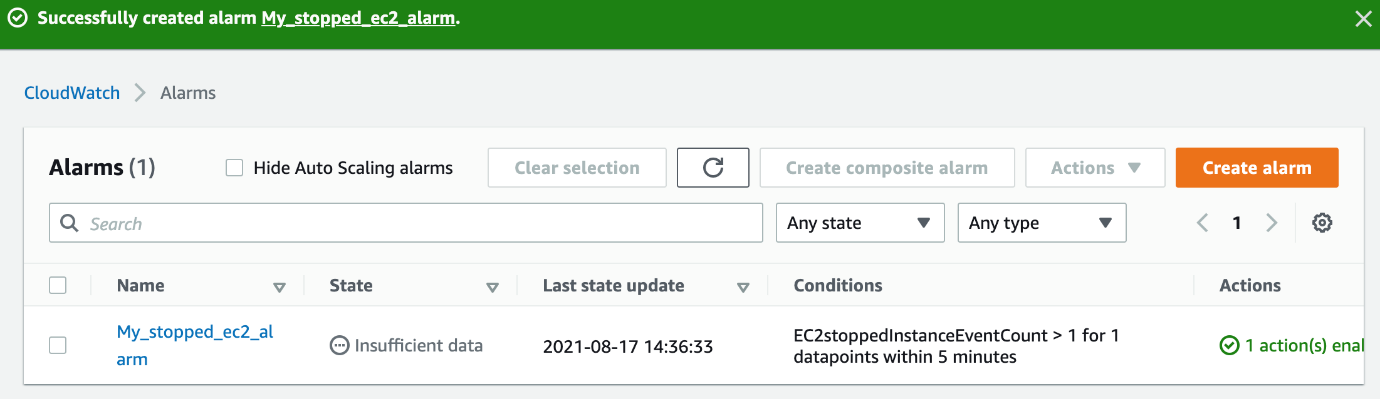


* Click on the  button to complete the alarm creation.

1. Give the name for your alarm and complete the steps as shown below:

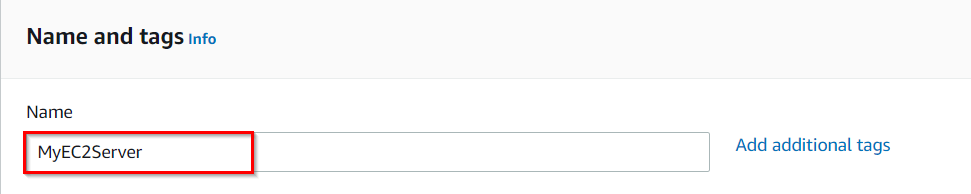
* Alarm name : Enter ***My\_stopped\_ec2\_alarm***
* Alarm description        : Enter ***Alarm to count the stopped instances count***
* Review the details and click on .

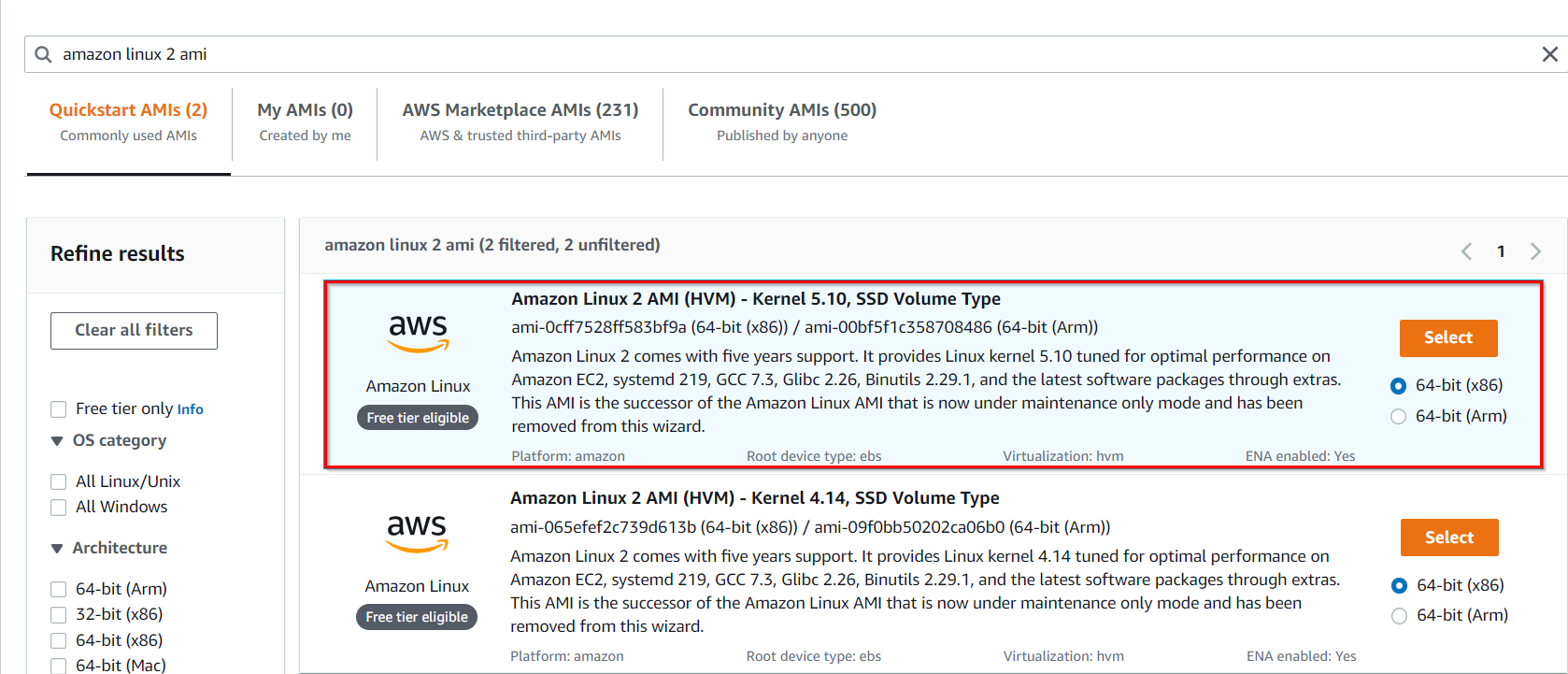
1. Navigate to the CloudWatch dashboard and click on alarms. You should see the alarm created in the above step under**insufficient data** as shown below:



Task 5: Creating an EC2 Instance to Trigger our Alarm

1. Make sure you are in the **N.Virginia**Region.
2. Navigate to **EC2** by clicking on the  menu in the top, then click on **EC2** in the **Compute** section
3. Navigate to **Instances** on the left panel and click on 
4. Name : Enter ***MyEC2Server***

******

1. **For Amazon Machine Image (AMI):** Search for **Amazon Linux 2 AMI** in the search box and click on the **select** button.  
   

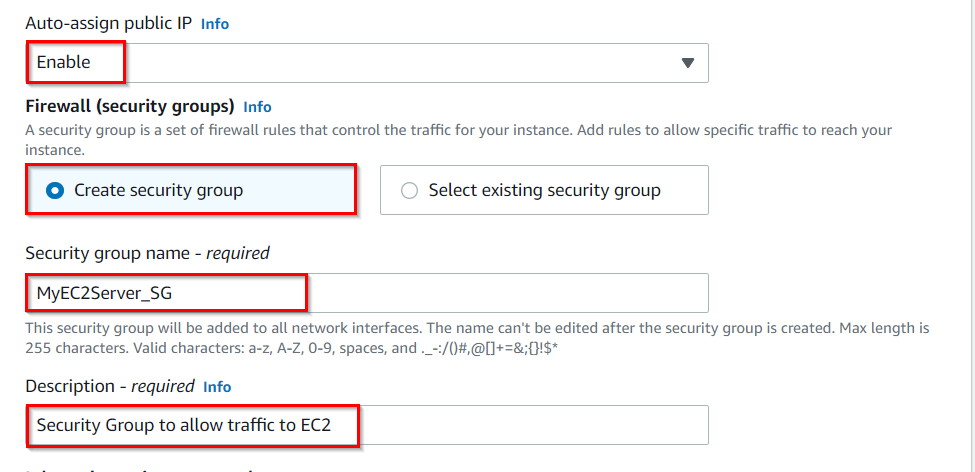
**Note: if there are two AMI's present for Amazon Linux 2 AMI, choose any of them.**

1. **For Instance Type:** select ***t2.micro***

****

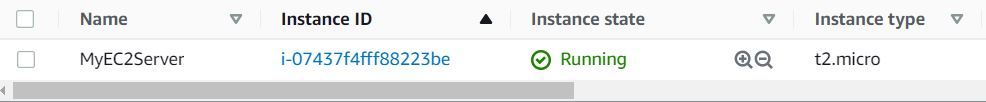
1. **For Key pair:** Select **Create a new key pair** Button
   1. Key pair name: **WhizKey**
   2. Key pair type: **RSA**
   3. Private key file format: **.pem**
2. Select **Create key pair** Button.
3. In Network Settings Click on **Edit**:

* Auto-assign public IP: **Enable**
* Select **Create new Security group**
* Security group name : Enter ***MyEC2Server\_SG***
* Description : Enter ***Security Group to allow traffic to EC2***

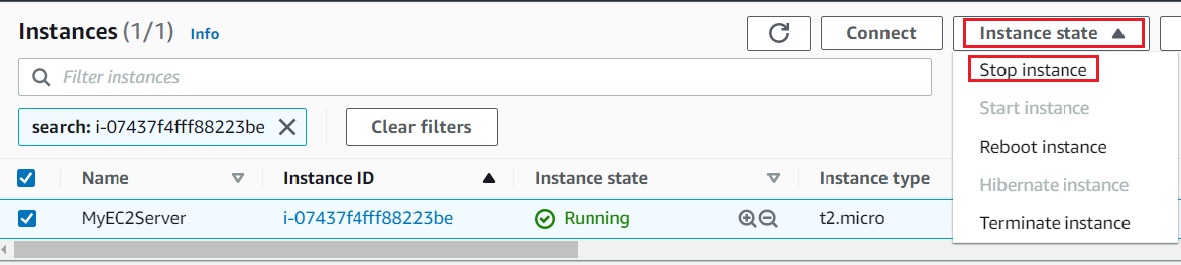
**

* Check **Allow SSH from** and Select **Anywhere** from dropdown
  + To add **SSH**,
    1. Choose Type: 
    2. Source: Select 

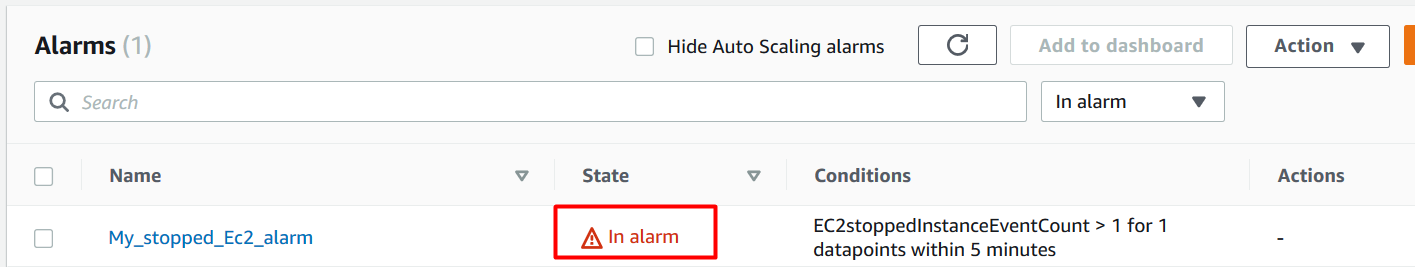
1. Keep Rest thing Default and Click on **Launch Instance** Button.
2. Select **View all Instances** to View Instance you Created
3. **Launch Status:**Your instance is now launching, Click on the instance ID and wait for complete initialization of instance.

******

1. Once the EC2 instance launches successfully, **start and stop the instance 2 to 3 times** as shown in the below screenshot:

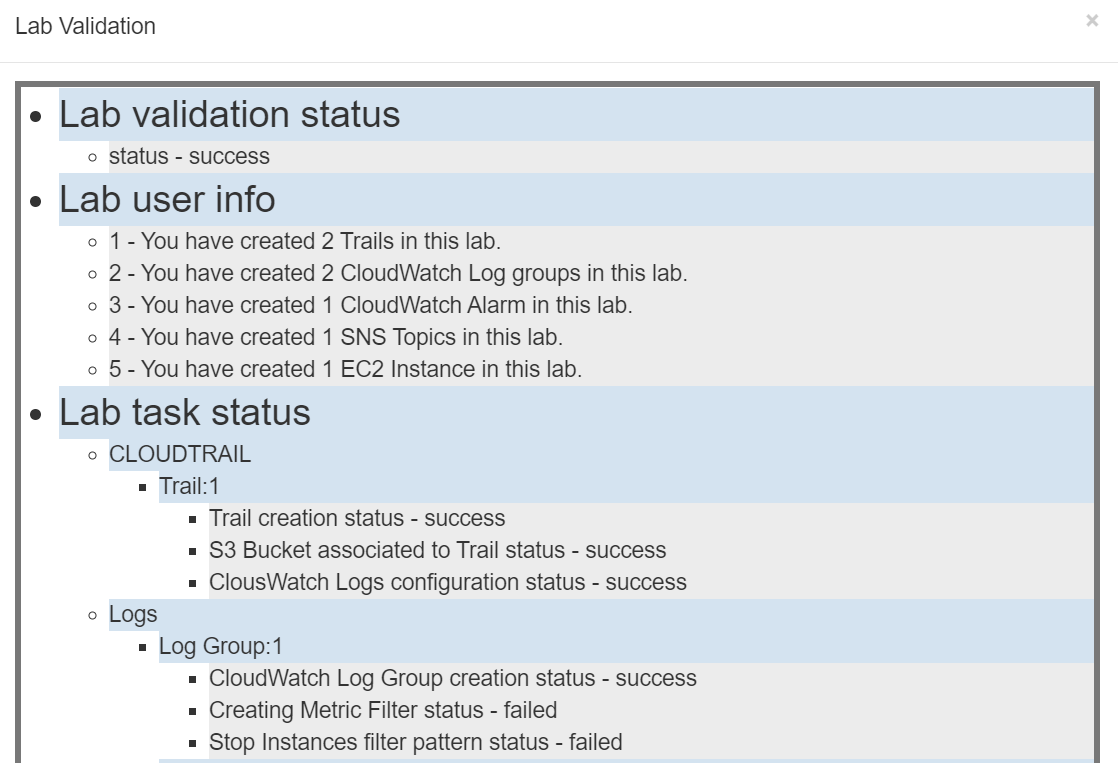


1. Navigate to the CloudWatch console and click on **In alarm**under **Alarms** on the left side panel to see the newly-created alarm. It should show the state as **In alarm**, as shown below:



**Task 6: Validation Test**

1. Once the lab steps are completed, please click on the  button on the left side panel.
2. This will validate the resources in the AWS account and displays whether you have completed this lab successfully or not.
3. Sample output :



**Completion and Conclusion**

1. You have successfully created a CloudTrail and an S3 bucket to store logs.
2. You have created CloudWatch log groups and a metric filter for stopped EC2 instances.
3. You have successfully created SNS topic to receive the alert from CloudWatch.
4. You have successfully launched an EC2 instance.
5. You successfully stopped the instance a few times to check the working of the alarm.

**End Lab**