|  |
| --- |
| **PROJECT** |
| **CLOUD WATCH ALARM** |
|  |
|  |
|  |
|  |
|  |
|  |

Cloud Watch Alarm

**Aim**

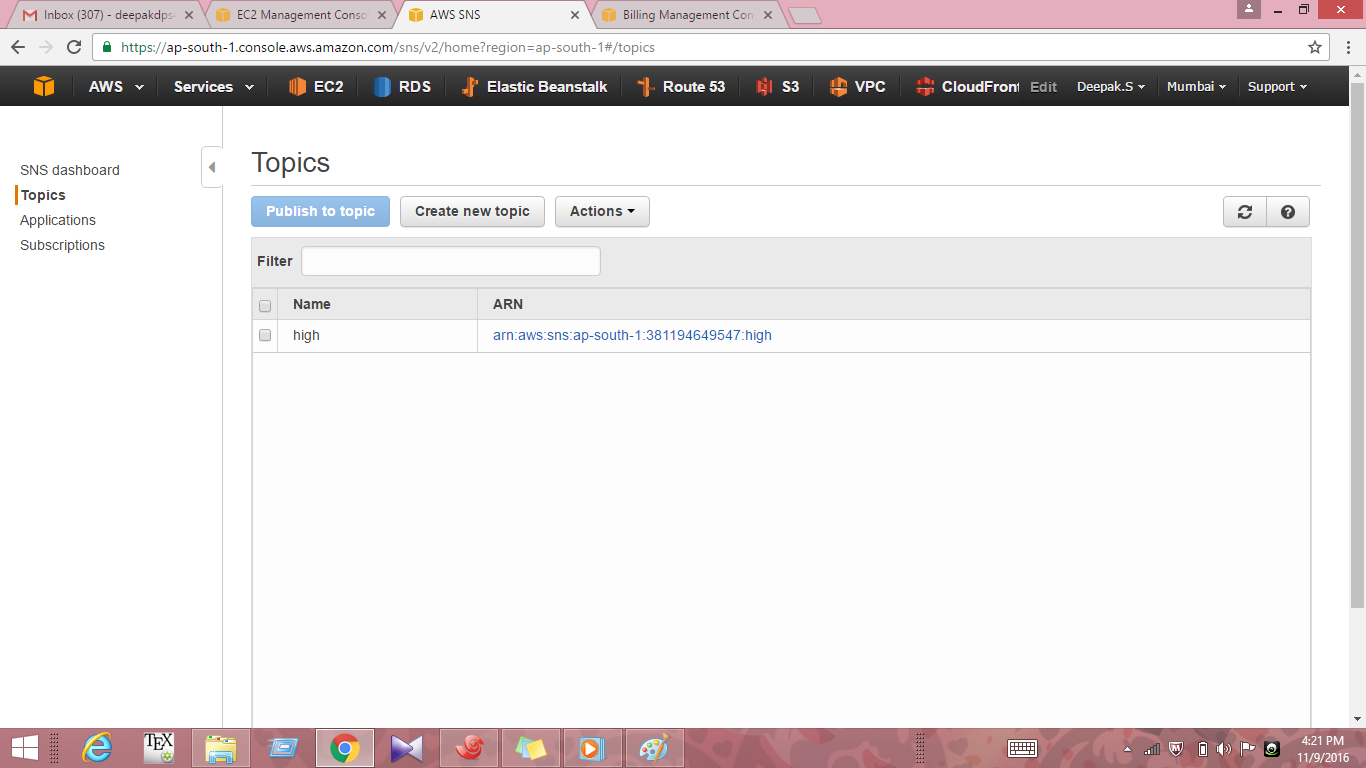
To create a CloudWatch Alarm to monitor the distribution

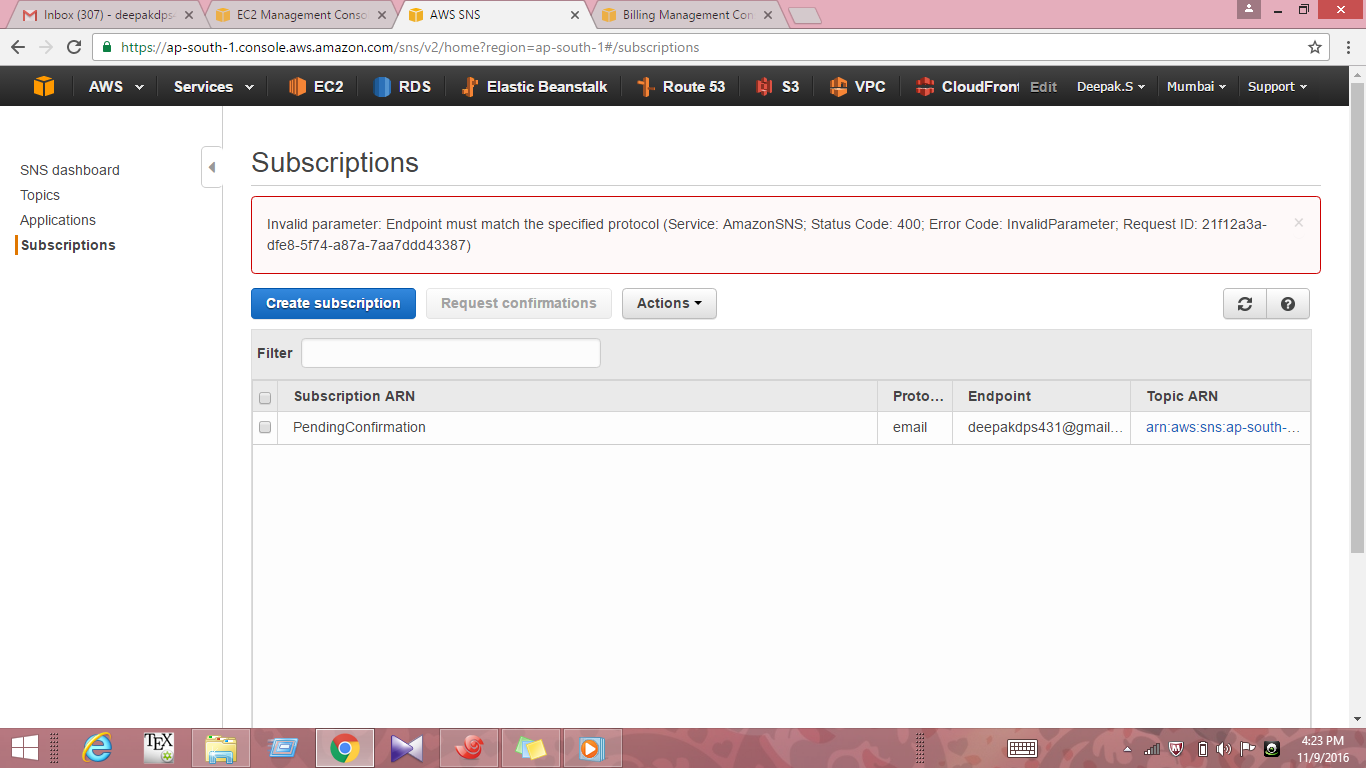
**Prerequisites**

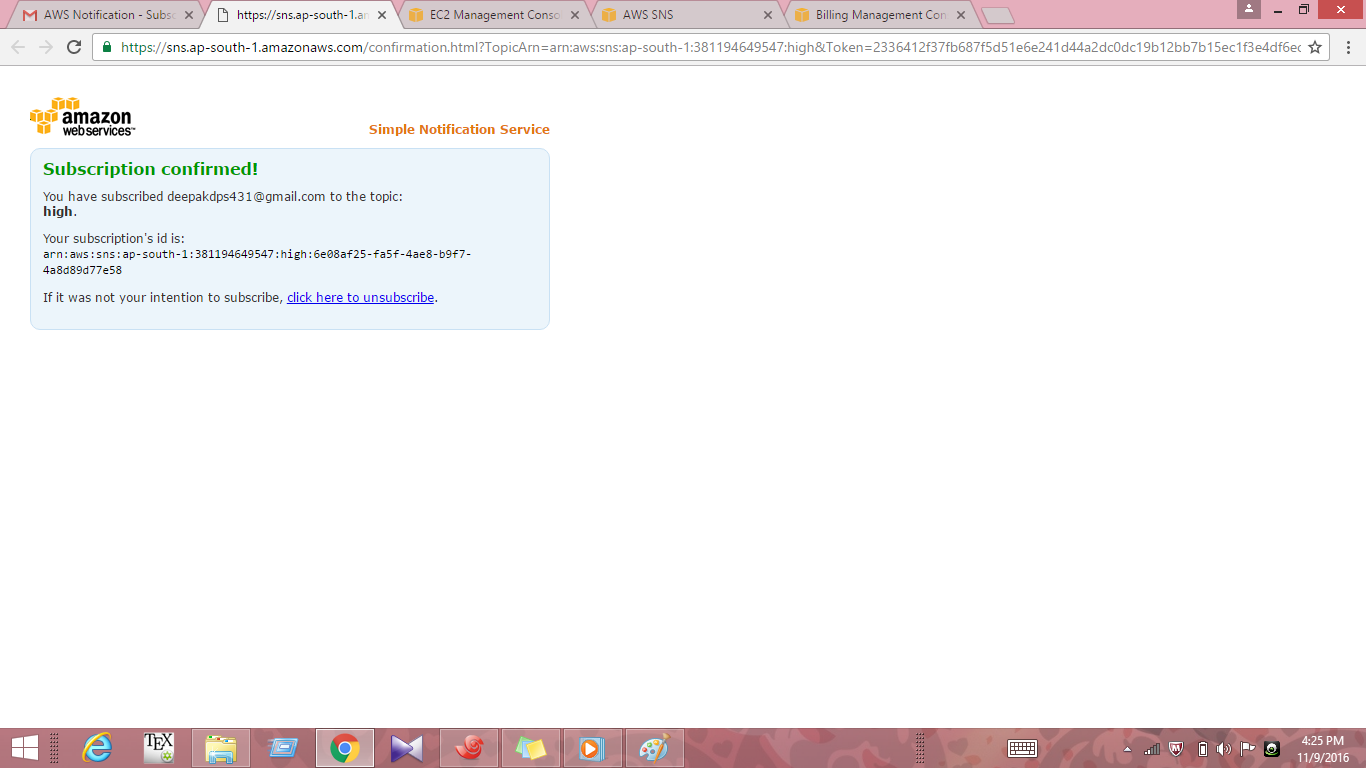
Amazon sns,ELB,autoscalling,EC2,Basic Linux commands

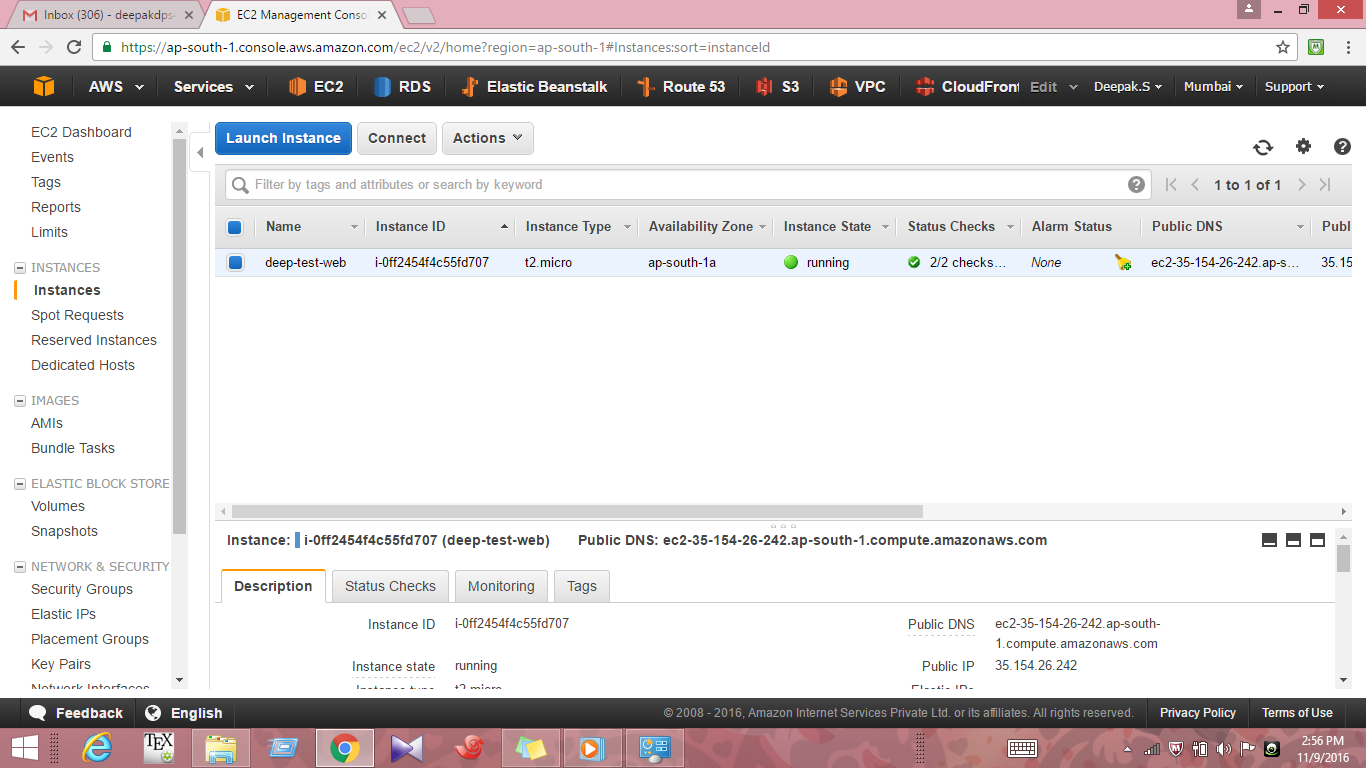
**Steps**

1.Log in to the aws management console

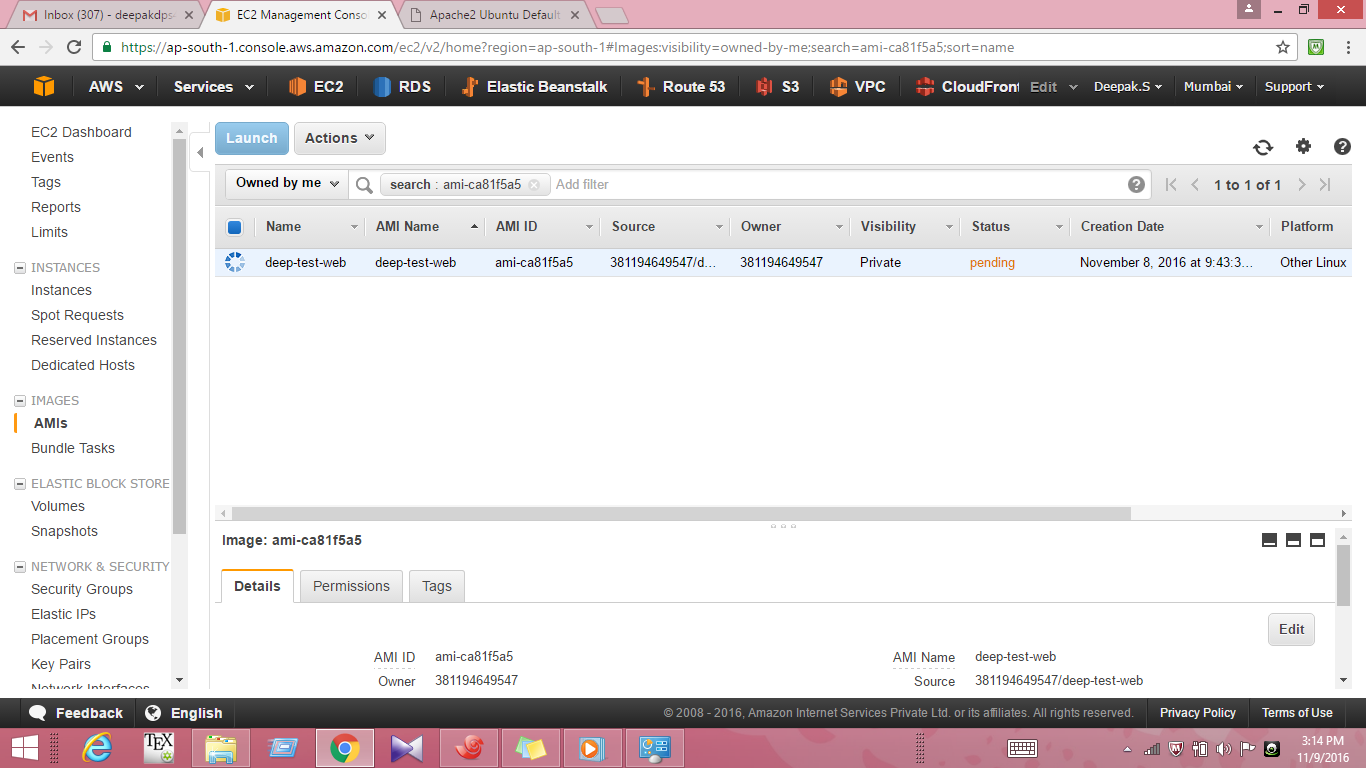
2.go to Services>SNS >Create a topic name’ high’. 

3. Create subscription to the email 

4.Confirm subscription by logging in to the gmail account 

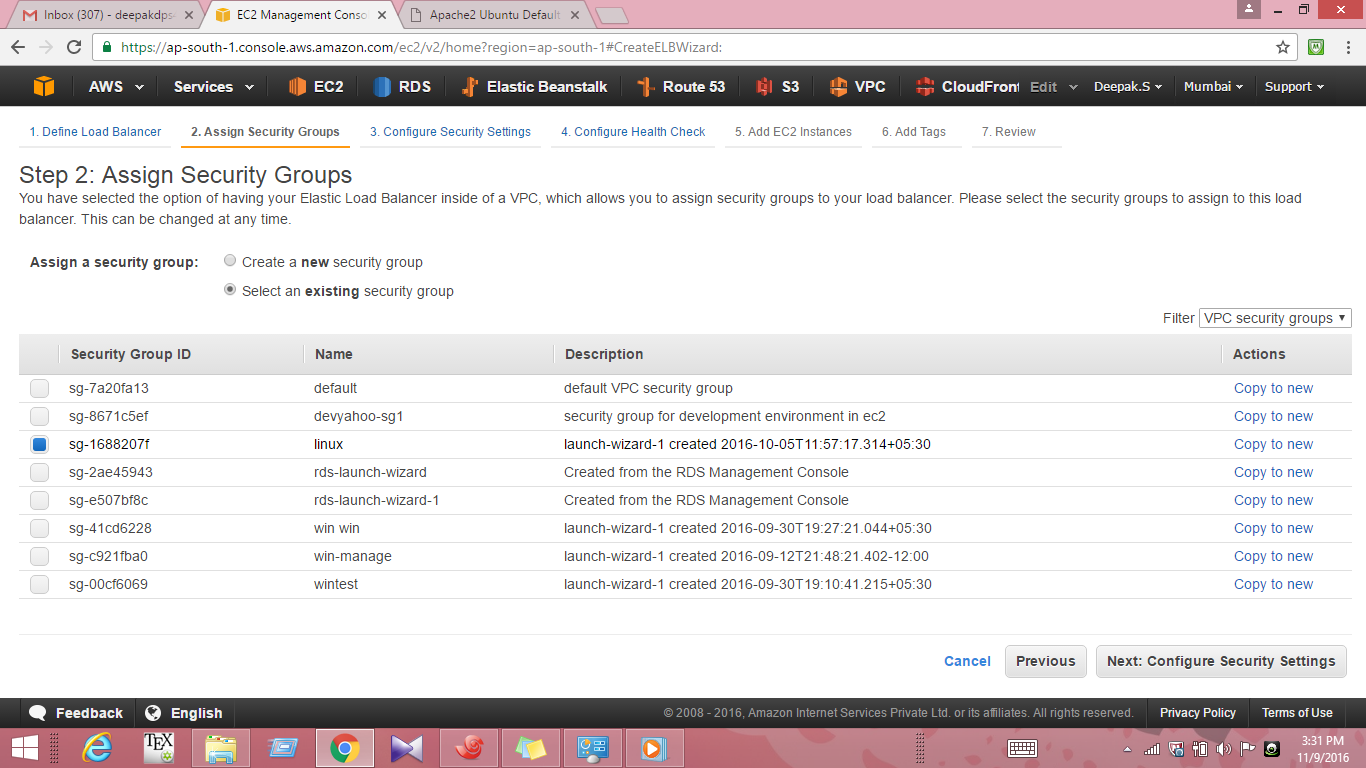
5. 

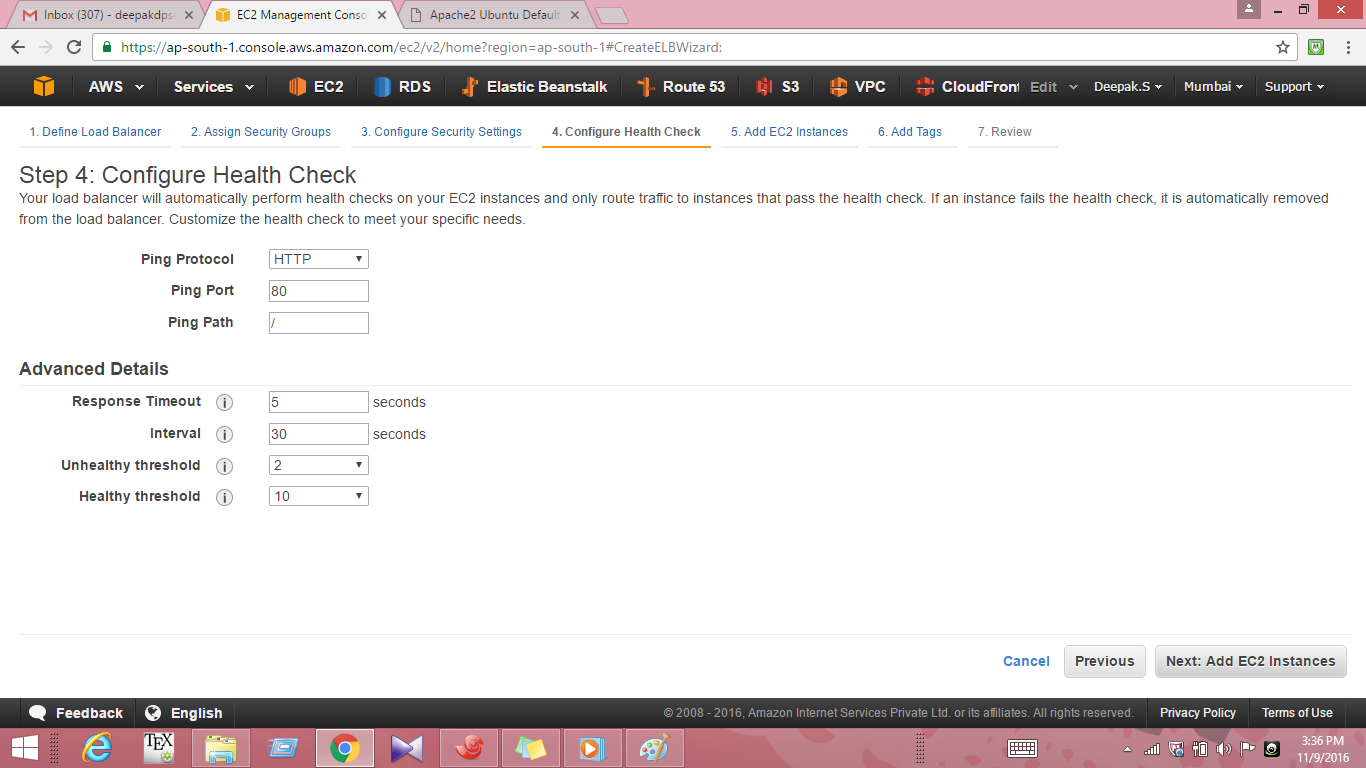
Create a t2micro ubuntu instance with name deep-test-web and configure the security group with ssh,http,https as open in inbound keep outbound as default.

6.Create an Image AMI of the ubuntu machine that was created previously 

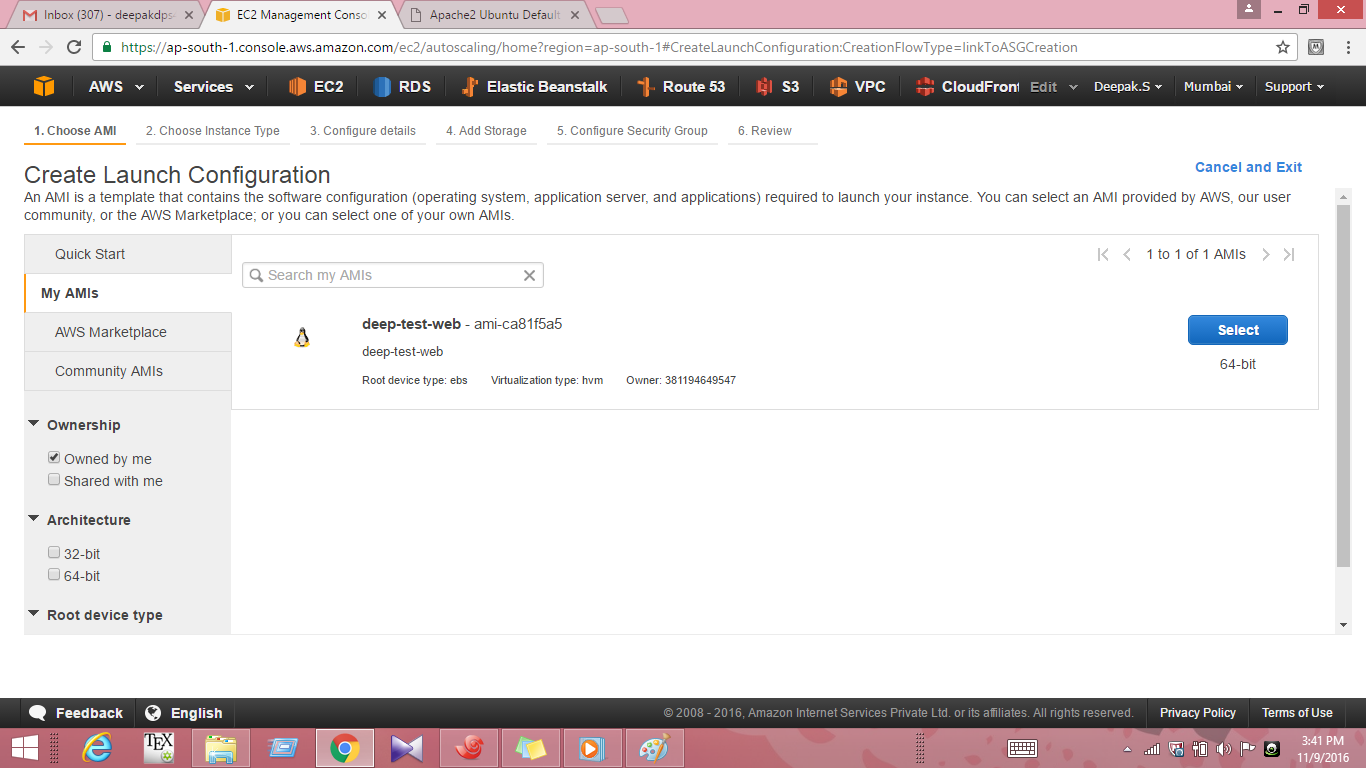
7.Now create a Load Balancer by clicking on the load balancer option in AWS Dashboard .Enter the details as below and click next

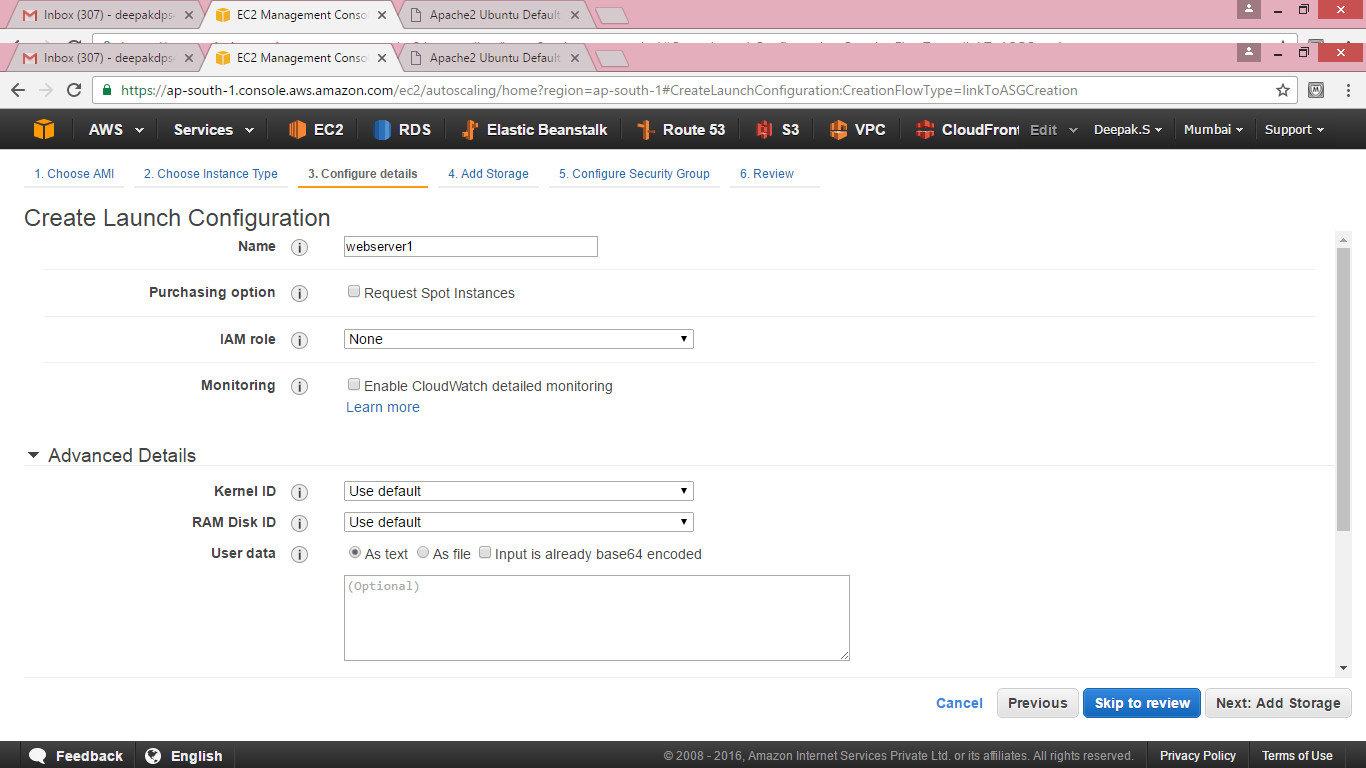


8.Select the same security group of the ubuntu instance 

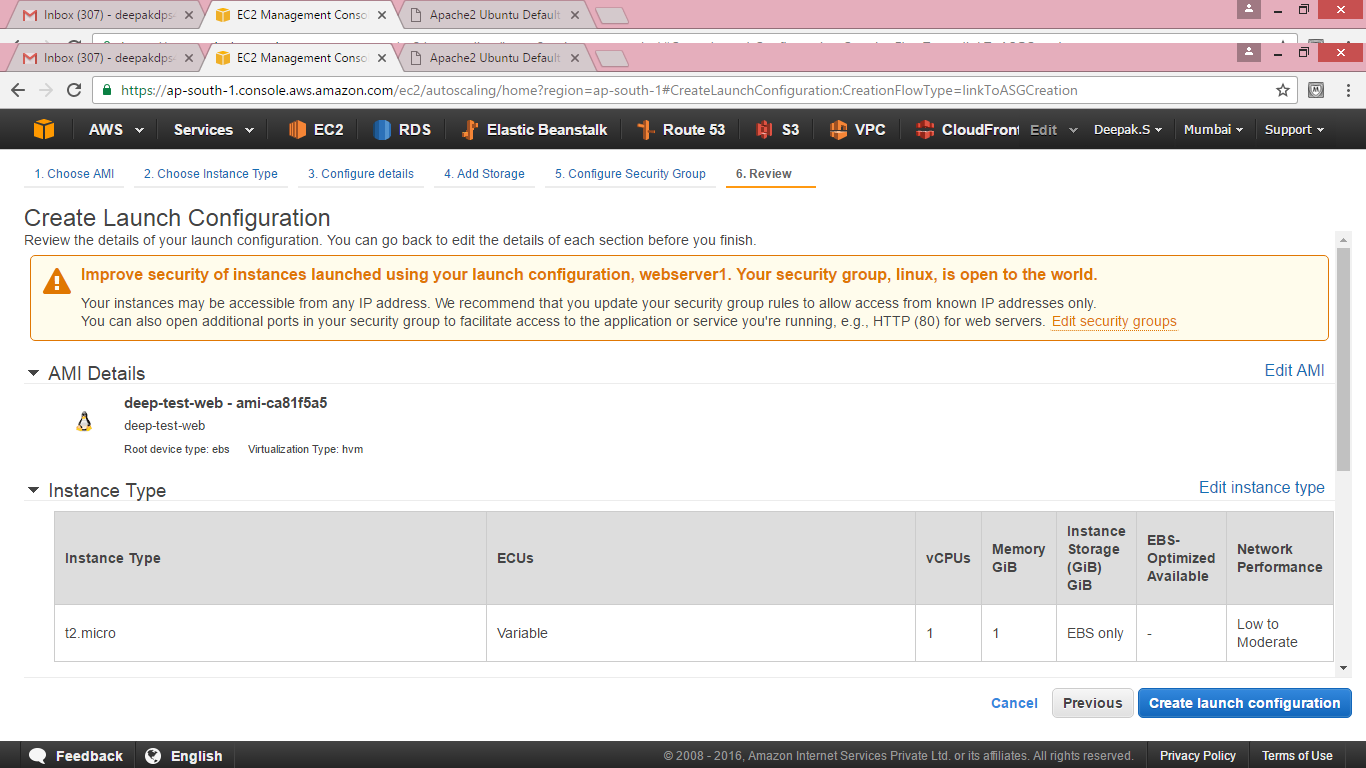
9.In configure health enter the details as below and click next do not add any ec2instances click next to Add tags then click Review and launch. 

10. Click on the autoscailing and click on create autoscailing group.Choose the AMI that was created before (deep-test-web) here



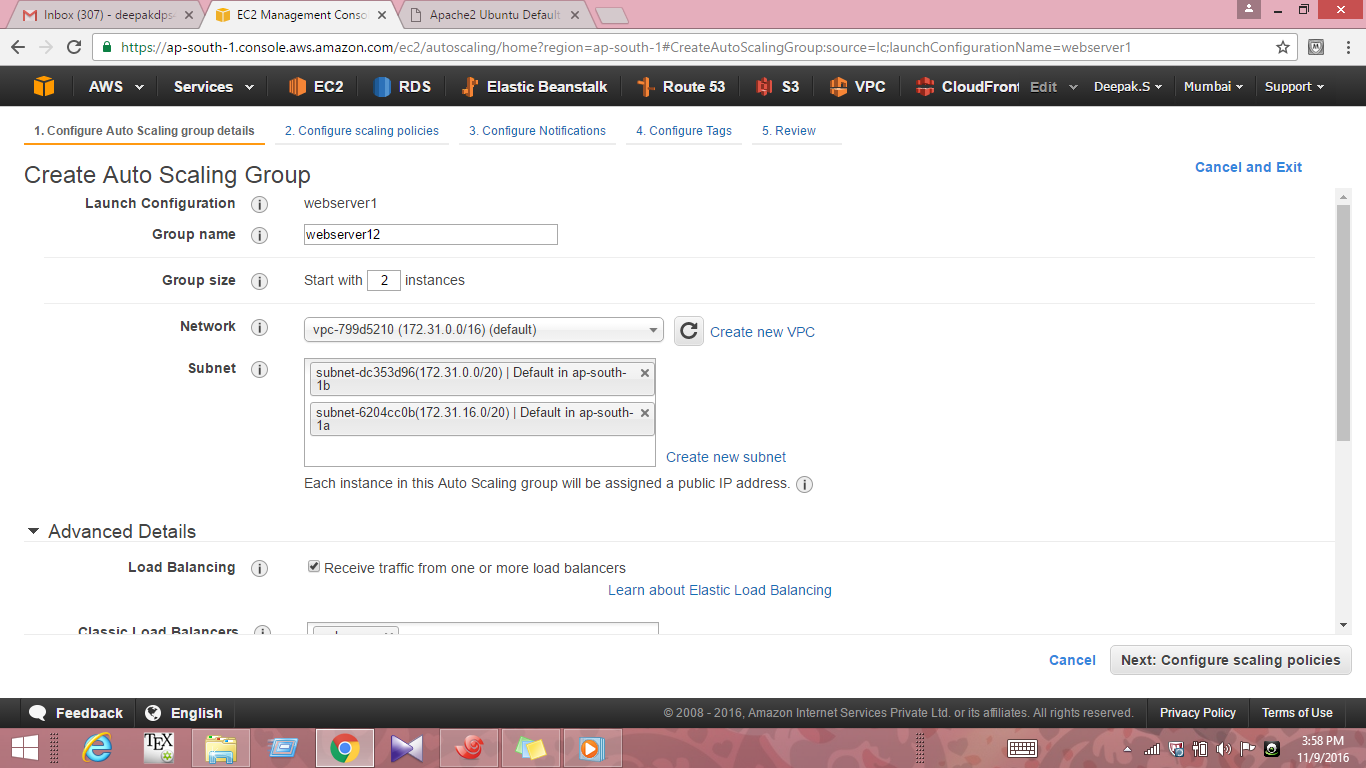
11.Choose the instance as t2micro configure the details as below 

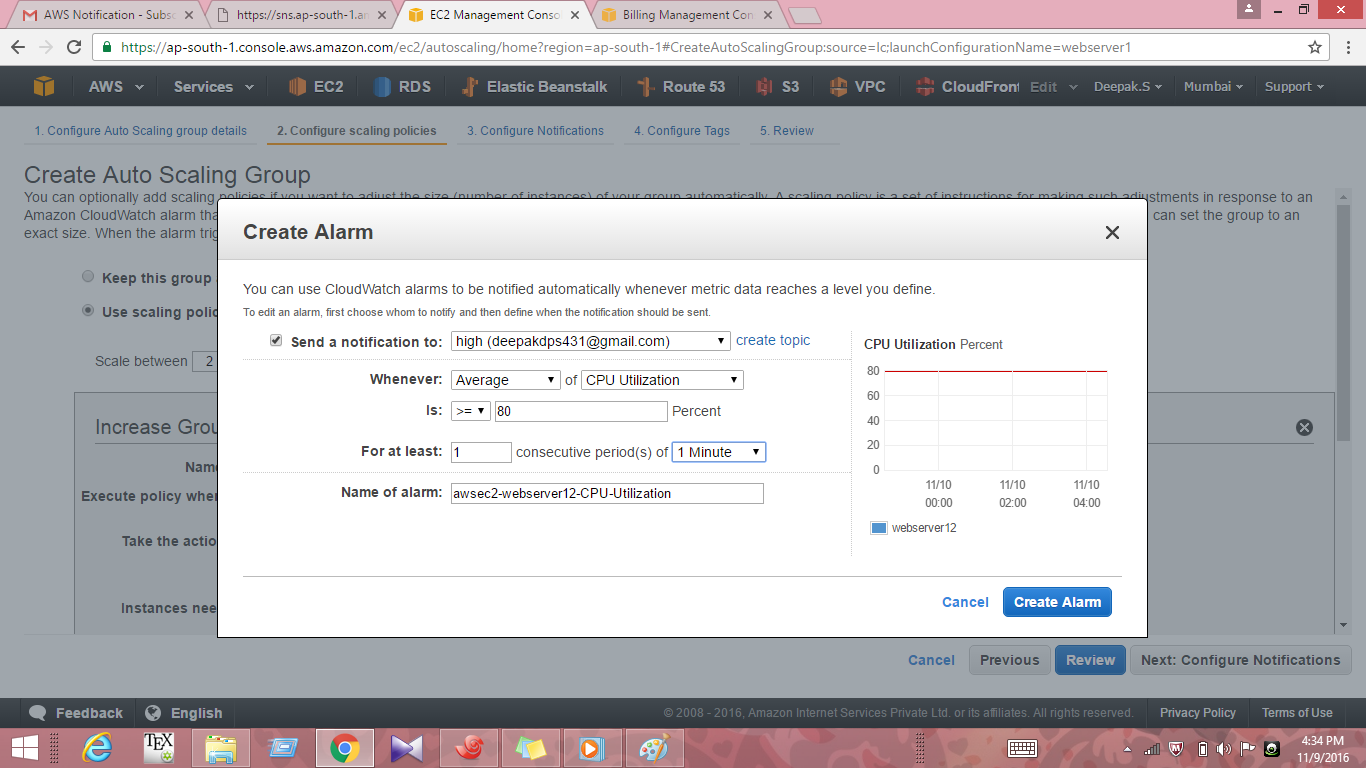
12.Click on create launch configurations

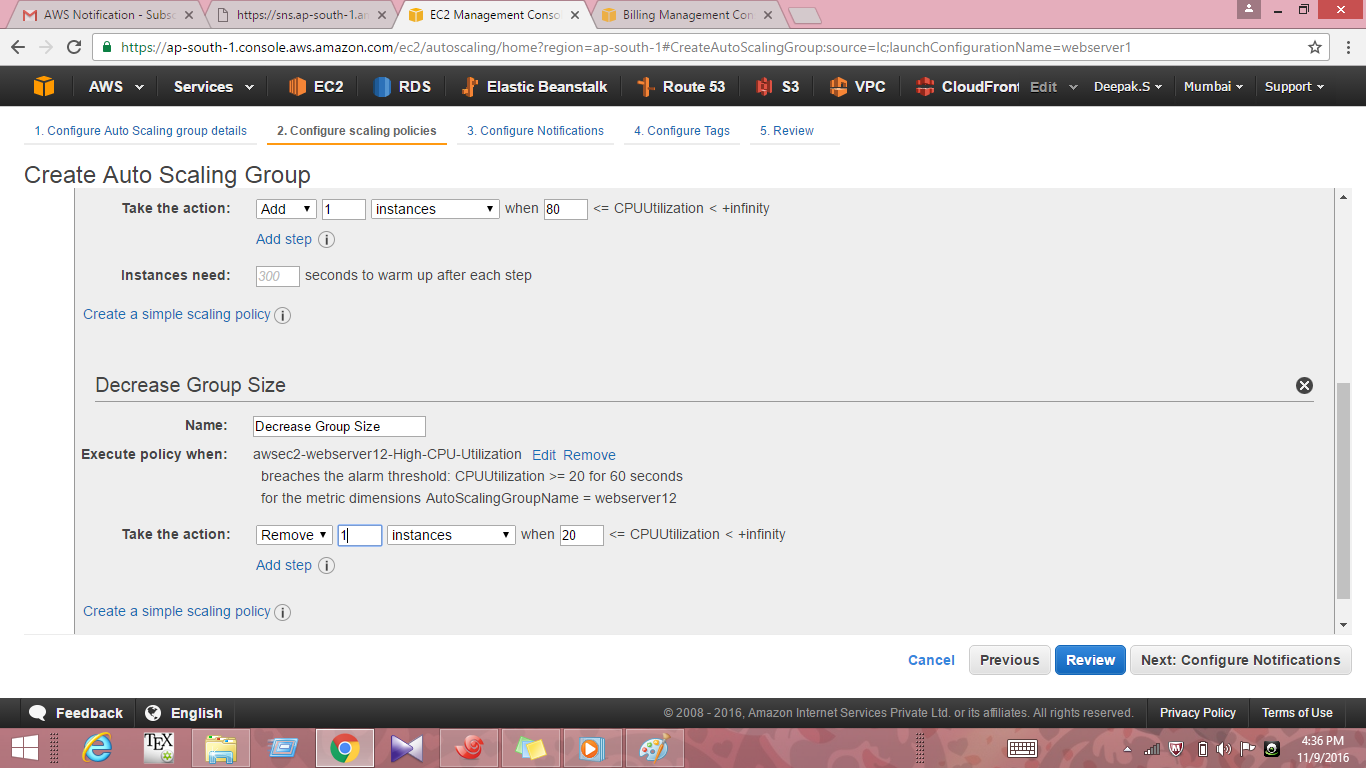


1.

13. Proceed for creating the autoscailing group enter the details as below

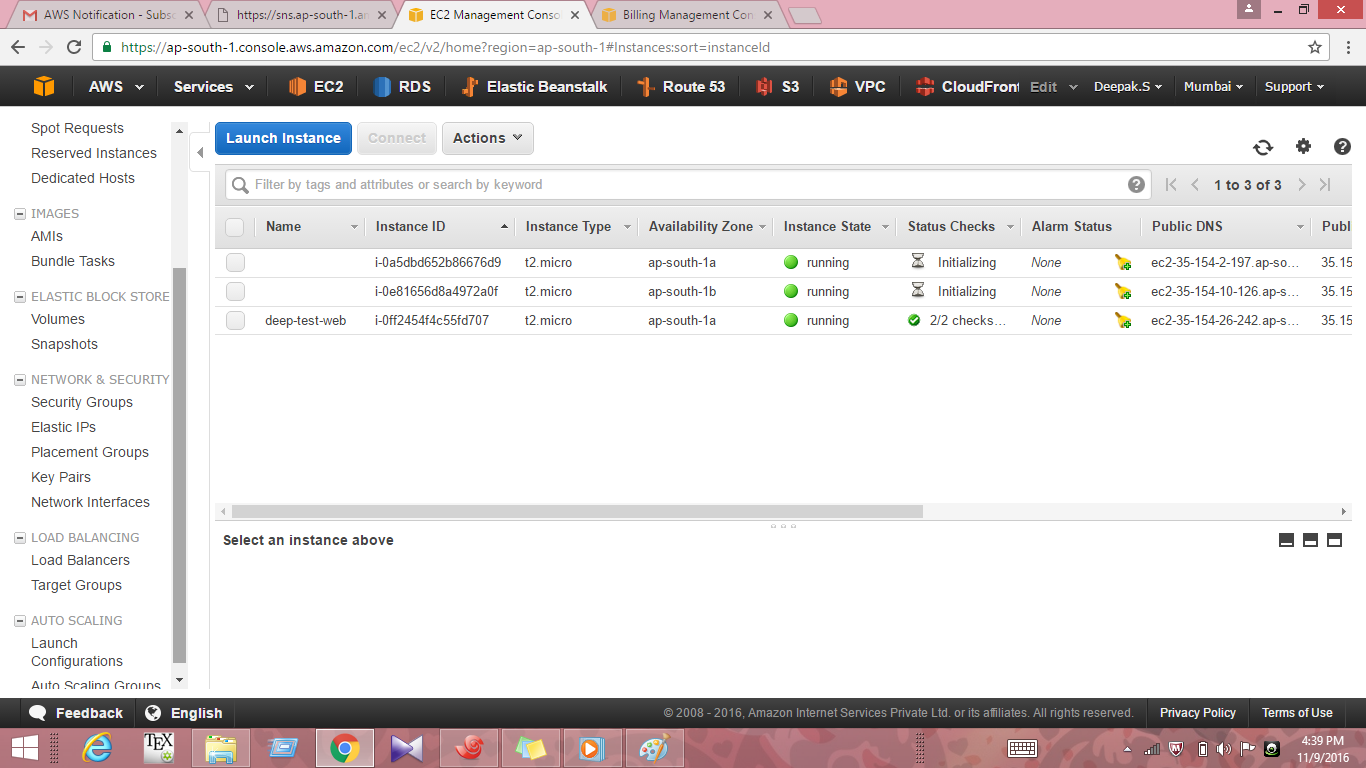


14.Create the first alarm for the case when CPU utilization is greater than 80 percent for 60seconds 

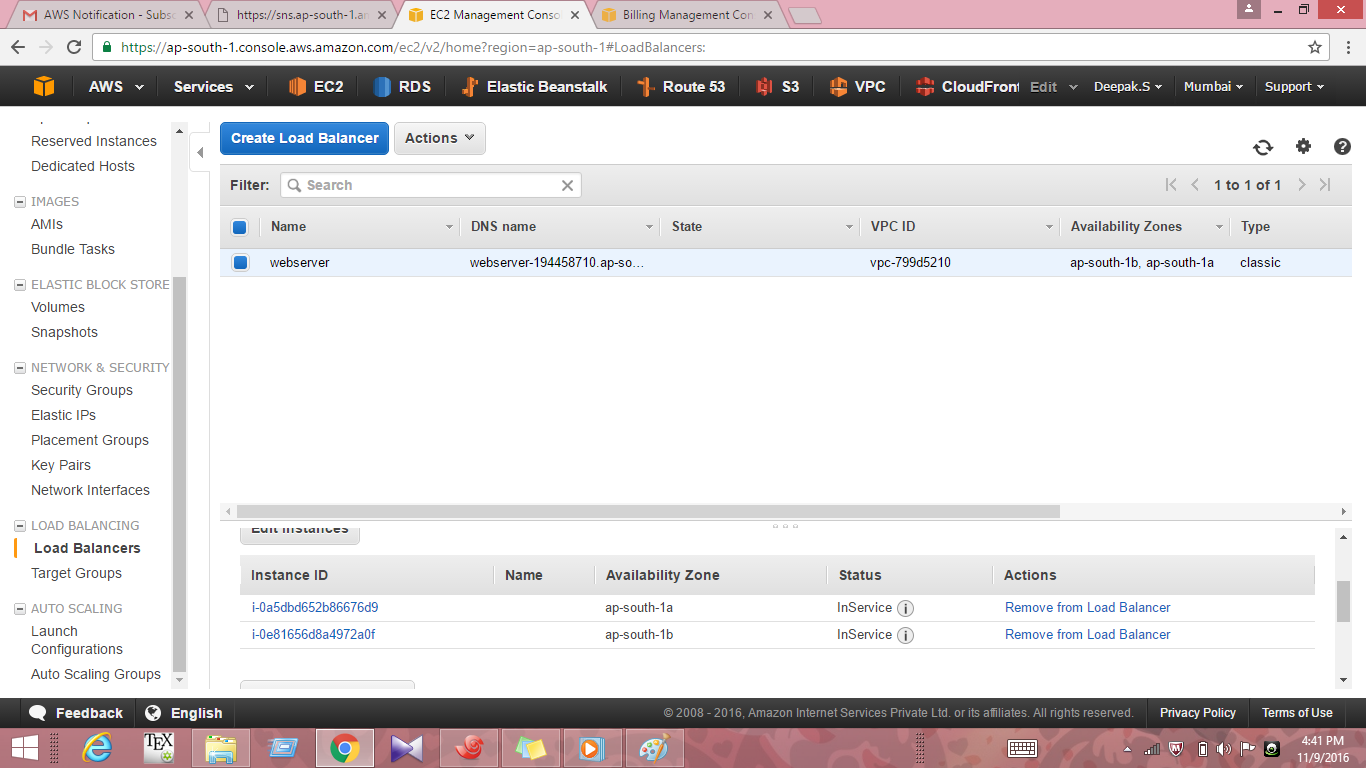


15.Create a similar alarm when the cpu utilization is 20percent then proceed with tags and create the group

16.Note that the two instances are being created in ap -south 1a and 1b as below



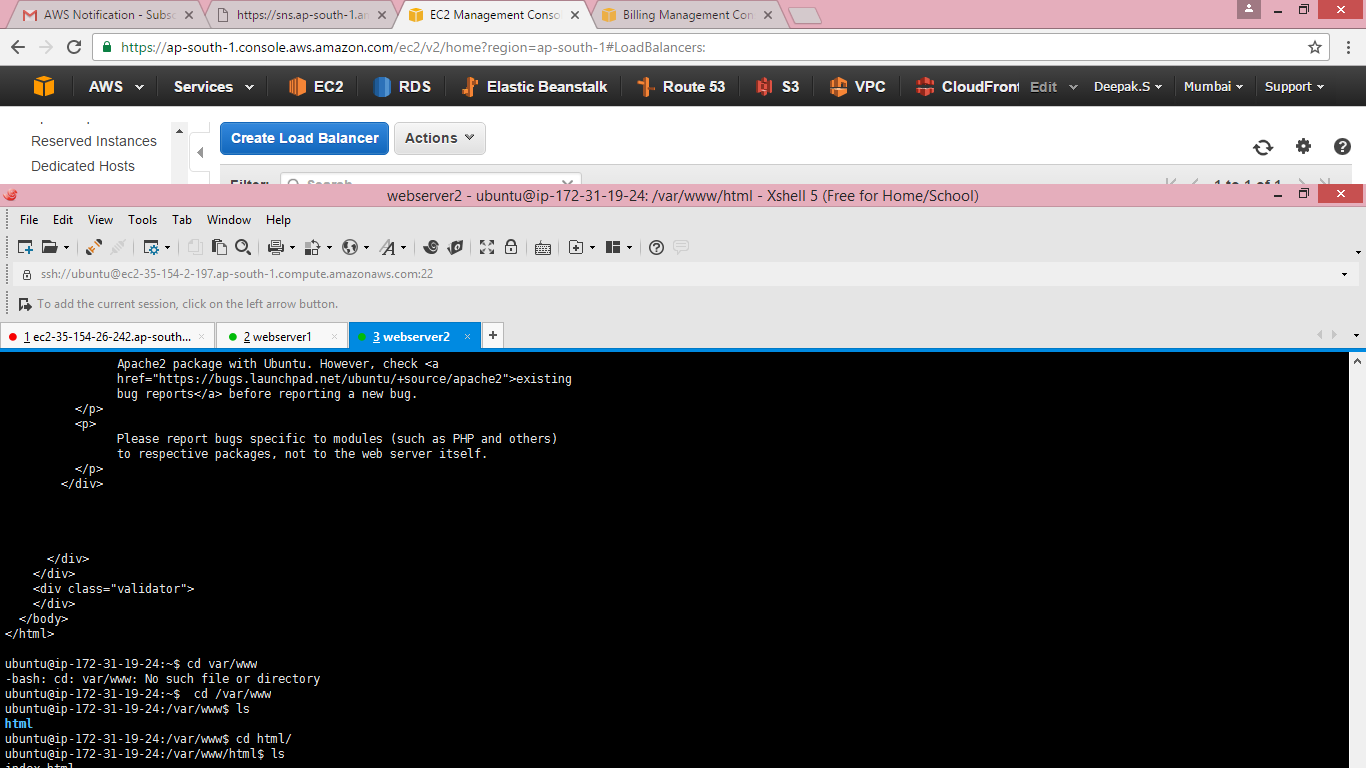
17.After some time we see that the 2 instances are created and attached to LoadBalancer as below



18.Note the endpoint url of both the webservers in 1a and 1b along with url of the loadbalancer created

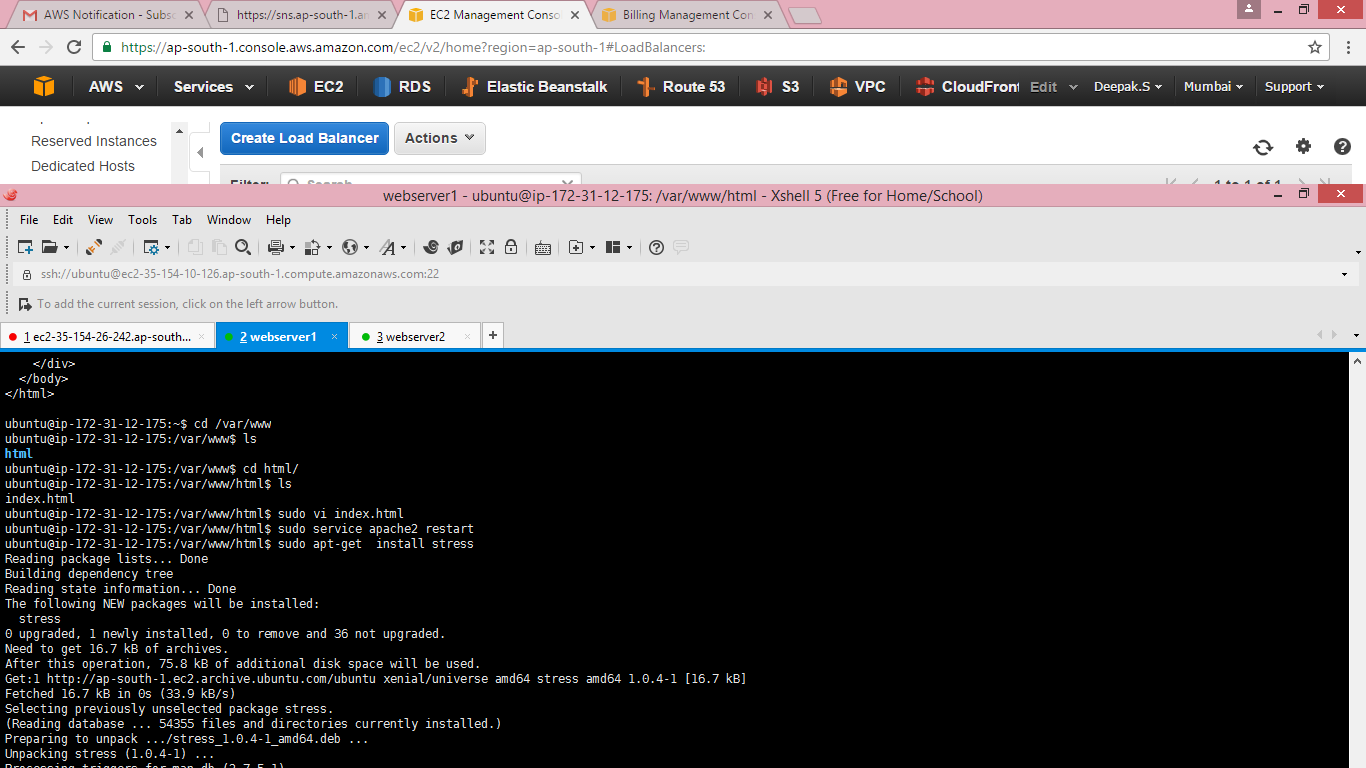


19.Configure the two webservers separately in xshell .Installed apache2 webservers and changed the name to webserver1 and webserver2 respectievely

20. The default apache html page is displayed for webserver1 ,webserver2 and the loadbalancer(webserver) by pasting the respectieve endpoints in the new tab of the browser as below



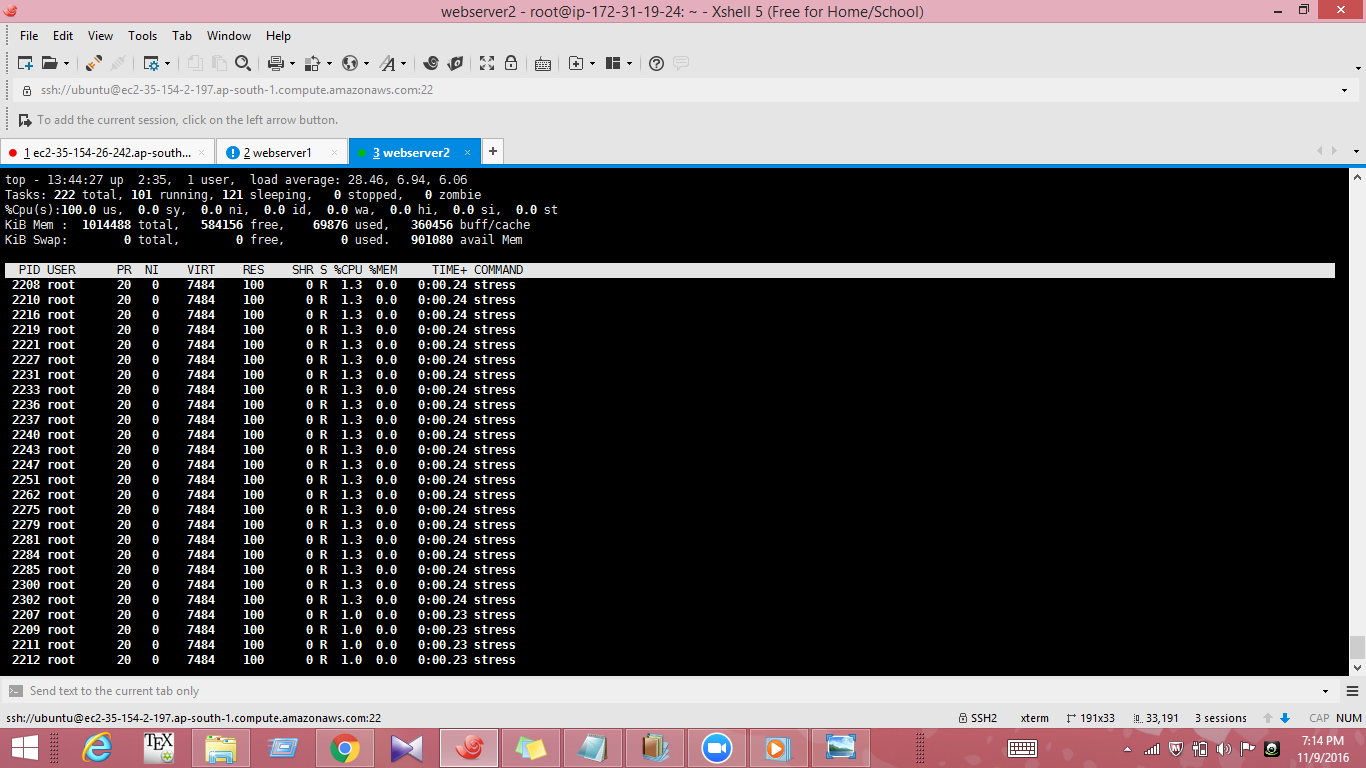
21.Add stress to any of the webservers say webserver1by the commands apt-get install stress and stress –c 30 .We can see The ELB will shift the traffic from Webserver1 to webserver2



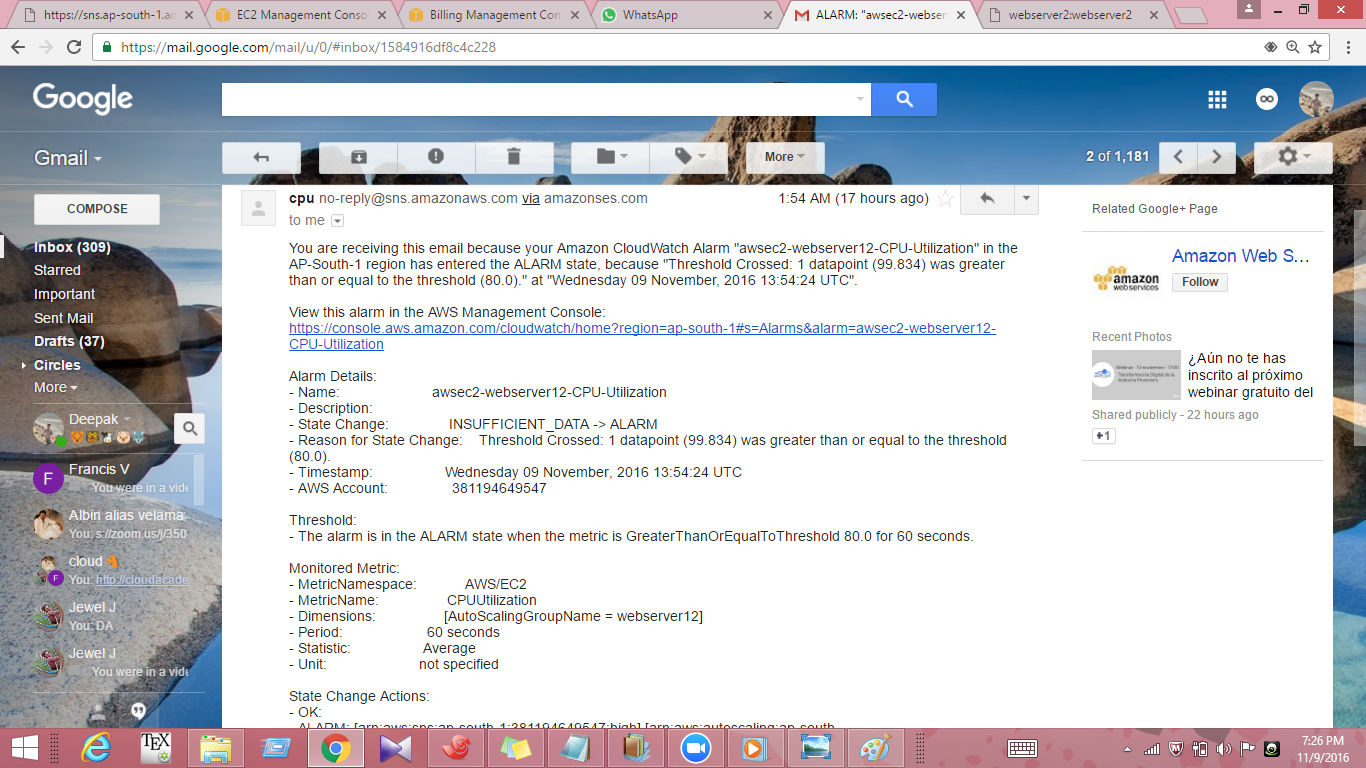
22.Check the mail for the cloud alarm.The corresponding mail when the stress is above 20 is received as below



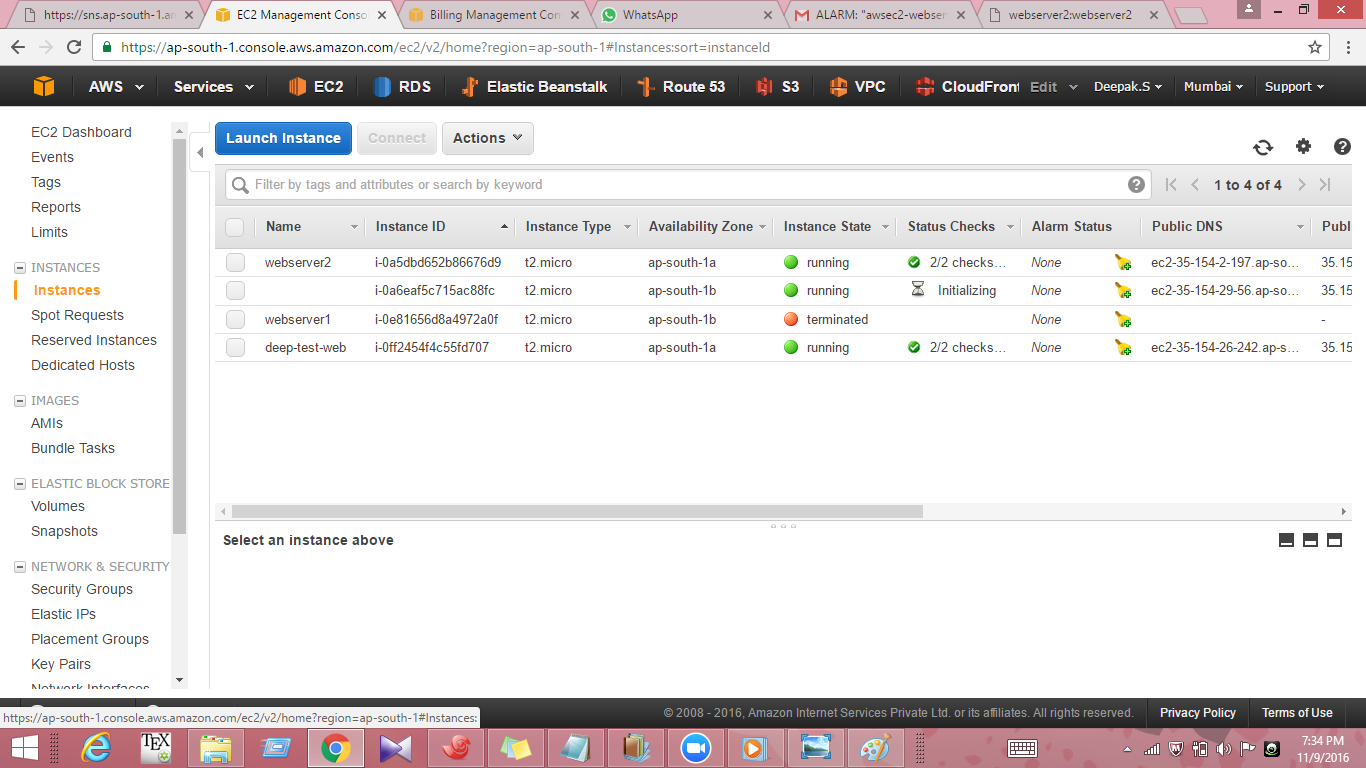
23.Again add stress so that the CPU utilization will be 100 by the command stress –c 100 check the cpu utilization on the required machine by typing top



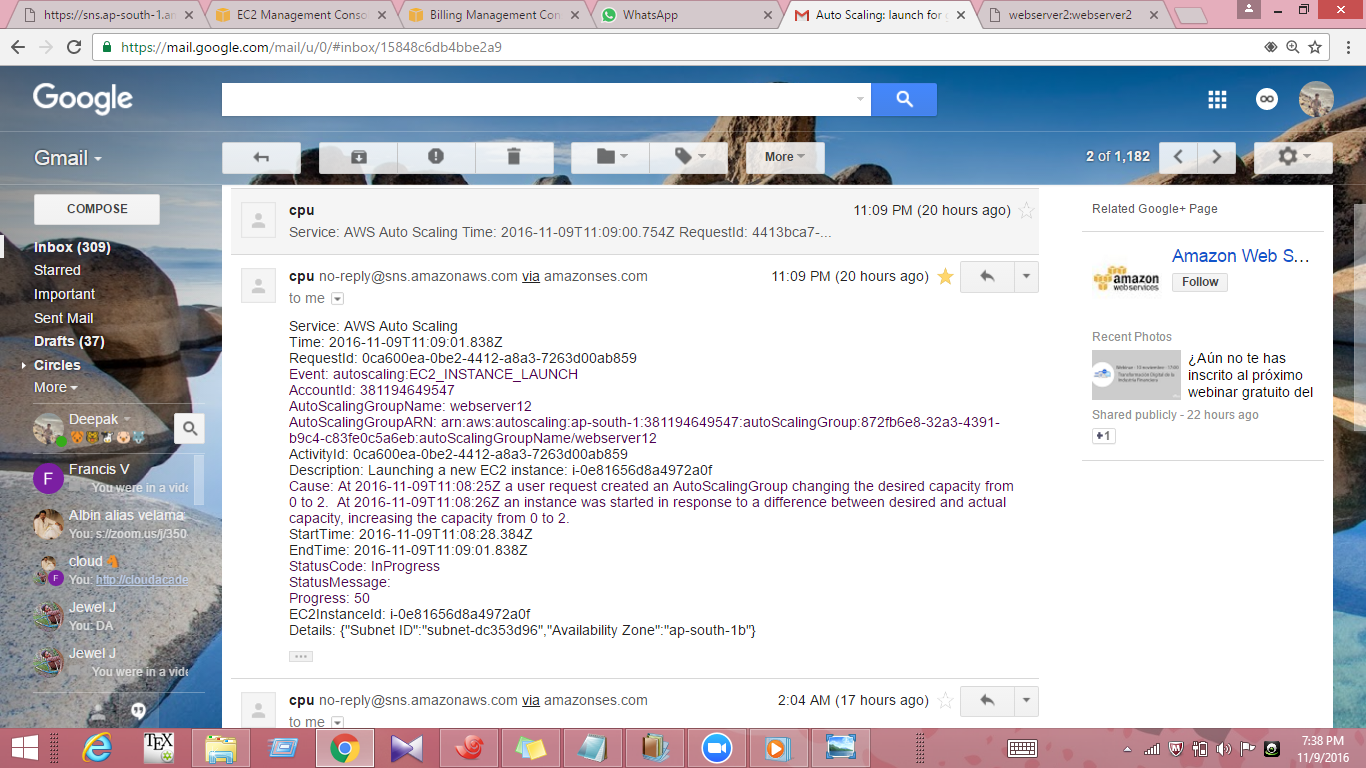
24. .Check the mail for the cloud alarm.The corresponding mail when the stress is above 80 is received as below



25.Stop the apache server on any of the machines we can see that the ELB shift the traffic to the healthy webserver and the autoscailing will creates a new replica of the unhealthy machine.Here



26.The same information will be available on the cloud watch alarm in the gmail as below



27.Also when the autoscailing group is terminated the cloudwatch alarm notifies as below



Result

* Cloud watch alarm was created and subscribed in gmail
* Alarm was successfully attached to an autoscailing group and ELB
* Alarm fired whenever the CPU utilization was above the specified constraints
* Alarm also notified the status of created Autoscailing group