

USING CLOUDWATCH FOR RESOURCE MONITORING, CREATE CLOUDWATCH ALARMS AND DASHBOARDS

Task Duration: 90 Minutes

AWS Region: US East (N.Virginia)
us-east-1.



TASK DETAILS

- Launching Lab Environment.
- Create EC2 Instance.
- SSH into EC2 Instance and install necessary Softwares.
- Create SNS Topic.
- Subscribe to an SNS Topic.
- Check EC2 CPU Utilization Metrics in CloudWatch Metrics.
- Create CloudWatch Alarm.
- Testing CloudWatch Alarm by Stressing CPU Utilization.
- Checking For an Email from the SNS Topic.
- Checking the CloudWatch Alarm Graph.
- Create a CloudWatch Dashboard.
- Validation of the lab.

TASK 1 & 2

- Task 1: is launching the AWS Management Console
 - Then Login using IAM username and Password
 - Make sure to select: N.Virginia Region
- Task 2: is Creating EC2 Instance
- Choose an AMI: Amazon Linux (Free tier eligible)
- Choose Instance Type: t2.micro; Next Configure Instance details
- Add storage: leave the values as default
- Add Tags:
 - Key: Name
 - Value: MyEC2Server
- Configure Security Group: Add SSH
 - Source: Anywhere
- Review and Launch: Using Key pair.
- Details shown in the next slide.

EC2 LAUNCH DETAILS.

New EC2 Experience

Tell us what you think

EC2 Dashboard

EC2 Global View

Events

Tags

Limits

Instances (1/1)

Search

I-0260510001c04bb6d

Clear filters

<input checked="" type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP
<input checked="" type="checkbox"/>	MyEC2Server1	I-0260510001c04bb6d	Running	t2.micro	-	No alarms	us-east-1a	ec2-3-91-52-78.comput...	3.91.52.78	-

Instance: i-0260510001c04bb6d

Details

Security

Networking

Storage

Status checks

Monitoring

Tags

▼ Instance summary

Instance ID

I-0260510001c04bb6d

IPv6 address

-

Hostname type

IP name: ip-172-31-26-169.ec2.internal

Answer private resource DNS name

IPv4 (A)

Auto-assigned IP address

3.91.52.78 [Public IP]

IAM Role

-

Public IPv4 address

3.91.52.78 | open address

Instance state

Running

Private IP DNS name (IPv4 only)

ip-172-31-26-169.ec2.internal

Instance type

t2.micro

VPC ID

vpc-6bf89d16 (Default VPC)

Subnet ID

subnet-c52e3688

Private IPv4 addresses

172.31.26.169

Public IPv4 DNS

ec2-3-91-52-78.compute-1.amazonaws.com | open address

Elastic IP addresses

-

AWS Compute Optimizer finding

Opt-in to AWS Compute Optimizer for recommendations. | Learn more

Auto Scaling Group name

-

▼ Instance details

Platform

Amazon Linux (Inferred)

AMI ID

ami-0c7f7528ff583bf9a

Monitoring

disabled

TASK3

- Task3: Is to SSH into EC2 Instance and install necessary Softwares
- Once SSH into my Terminal then I switched to root user
- Using these commands: Sudo su,
- Then update: yum update -y
- After the update I installed Stress Tool using these commands:
 - `sudo amazon-linux-extras install epel -y`
 - `yum install stress -y`
 - Details are shown next slide.

DETAILS OF SSH INTO THE EC2 INSTANCE AND INSTALLING SOFTWARES.

The screenshot displays the AWS Management Console interface for connecting to an EC2 instance and a terminal window showing the connection process and software installation.

AWS Console: Connect to instance

Connect to your instance i-0260510001c04bb6d using any of these options

EC2 Instance Connect | Session Manager | **SSH client** | EC2 serial console

Instance ID
i-0260510001c04bb6d

1. Open an SSH client.
2. Locate your private key file. The key used to launch this Instance is Myname10.pem
3. Run this command, if necessary, to ensure your key is not publicly viewable.
chmod 400 Myname10.pem
4. Connect to your instance using its Public DNS:
ec2-3-91-52-78.compute-1.amazonaws.com

Example:
~user@ec2-3-91-52-78.compute-1.amazonaws.com

Note: In most cases, the guessed user name is correct. However, read your AMI usage instructions to check the AMI owner has changed the default AMI user name.

Terminal Output:

```
(base) munasarabdi10@Mune-MacBook-Pro ~ % downloads
zsh: command not found: downloads
(base) munasarabdi10@Mune-MacBook-Pro ~ % dc Downloads
(base) munasarabdi10@Mune-MacBook-Pro ~ % chmod 400 Myname10.pem
chmod: Myname10.pem: No such file or directory
(base) munasarabdi10@Mune-MacBook-Pro ~ % cd downloads/
(base) munasarabdi10@Mune-MacBook-Pro downloads % chmod 400 Myname10.pem
(base) munasarabdi10@Mune-MacBook-Pro downloads % ssh -i "Myname10.pem" ec2-user@ec2-3-91-52-78.compute-1.amazonaws.com
The authenticity of host 'ec2-3-91-52-78.compute-1.amazonaws.com (3.91.52.78)' can't be established.
ED25519 key fingerprint is SHA256:ah9wM02DolURNOvrp9Q0dKKNZMkfu3u09bWltkpm8.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-3-91-52-78.compute-1.amazonaws.com' (ED25519) to the list of known hosts.

 _ _ | _ _ | _ _ |
--| _ _ | _ _ | _ _ |
--| _ _ | _ _ | _ _ |

Amazon Linux 2 AMI

https://aws.amazon.com/amazon-linux-2/
5 package(s) needed for security, out of 14 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-26-169 ~]$ sudo su
[root@ip-172-31-26-169 ec2-user]# yum update -y
bash: yum: command not found
[root@ip-172-31-26-169 ec2-user]# yum update -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core

Resolving Dependencies
--> Running transaction check
--> Package amazon-ssm-agent.x86_64 0:3.1.1188.0-1.amzn2 will be updated
--> Package amazon-ssm-agent.x86_64 0:3.1.1575.0-1.amzn2 will be an update
--> Package curl.x86_64 0:7.79.1-2.amzn2.0.1 will be updated
--> Package curl.x86_64 0:7.79.1-4.amzn2.0.1 will be an update
--> Package expat.x86_64 0:2.1.0-12.amzn2.0.4 will be updated
--> Package expat.x86_64 0:2.1.0-14.amzn2.0.1 will be an update
--> Package glibc.x86_64 0:2.26-58.amzn2 will be updated
--> Package glibc.x86_64 0:2.26-59.amzn2 will be an update
--> Package glibc-all-langpacks.x86_64 0:2.26-58.amzn2 will be updated
--> Package glibc-all-langpacks.x86_64 0:2.26-59.amzn2 will be an update
--> Package glibc-common.x86_64 0:2.26-58.amzn2 will be updated
--> Package glibc-common.x86_64 0:2.26-59.amzn2 will be an update
--> Package glibc-locale-source.x86_64 0:2.26-58.amzn2 will be updated
--> Package glibc-locale-source.x86_64 0:2.26-59.amzn2 will be an update
--> Package glibc-minimal-langpack.x86_64 0:2.26-58.amzn2 will be updated
--> Package glibc-minimal-langpack.x86_64 0:2.26-59.amzn2 will be an update
--> Package initscripts.x86_64 0:9.49.47-1.amzn2.0.1 will be updated
--> Package initscripts.x86_64 0:9.49.47-1.amzn2.0.2 will be an update
--> Package kernel.x86_64 0:5.10.126-117.558.amzn2 will be installed
--> Package libcrypt.x86_64 0:2.26-58.amzn2 will be updated
--> Package libcrypt.x86_64 0:2.26-59.amzn2 will be an update
--> Package libcurl.x86_64 0:7.79.1-2.amzn2.0.1 will be updated
--> Package libcurl.x86_64 0:7.79.1-4.amzn2.0.1 will be an update
```

TASK4

- Task 4: Create SNS Topic
- Make sure it's N.Virginia
- Navigate to Simple Notification Service
- Under details:
- Type; Select standard
- Name: MyServerMonitor
- Display name: MyServerMonitor
- Leave all other options as default and click create topic.
- Details is shown next slide.

CREATION OF SNS TOPIC.

Amazon SNS

×

Dashboard

Topics

Subscriptions

▼ Mobile

Push notifications

Text messaging (SMS)

Origination numbers

Amazon SNS > Topics > MyServerMonitor

MyServerMonitor

EditDeletePublish message

Details

Name
MyServerMonitor

Display name
MyServerMonitor

ARN
arn:aws:sns:us-east-1:072408161221:MyServerMonitor

Topic owner
072408161221

Type
Standard

SubscriptionsAccess policyDelivery retry policy (HTTP/S)Delivery status loggingEncryptionTags

Subscriptions (0)EditDeleteRequest confirmationConfirm subscriptionCreate subscription

Search

< 1 > ⓘ

ID

Endpoint

Status

Protocol

No subscriptions found

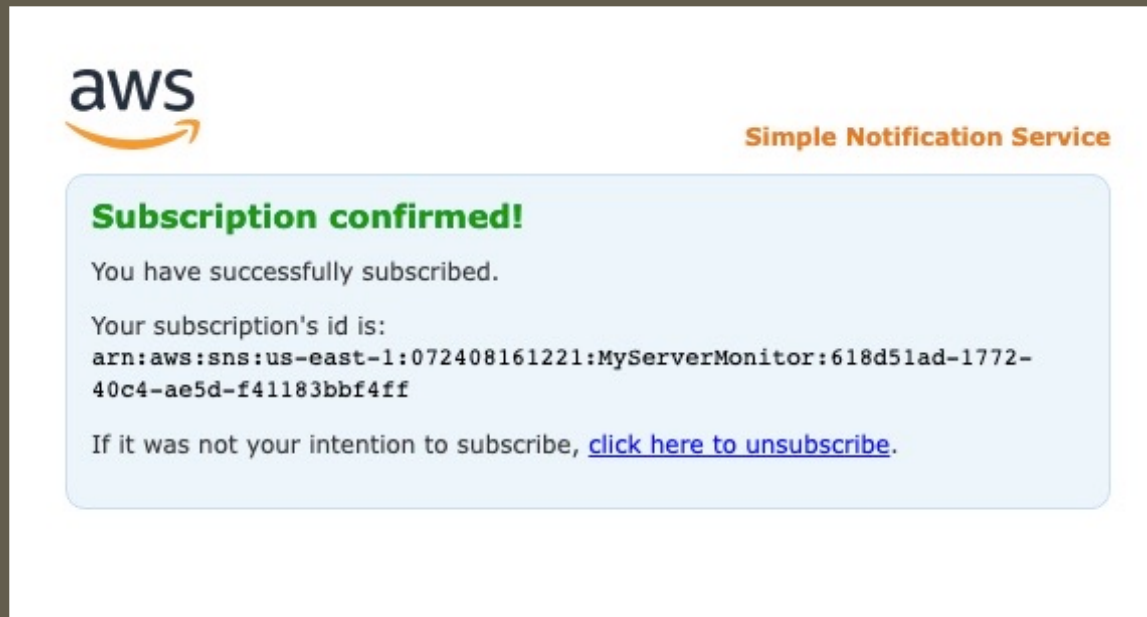
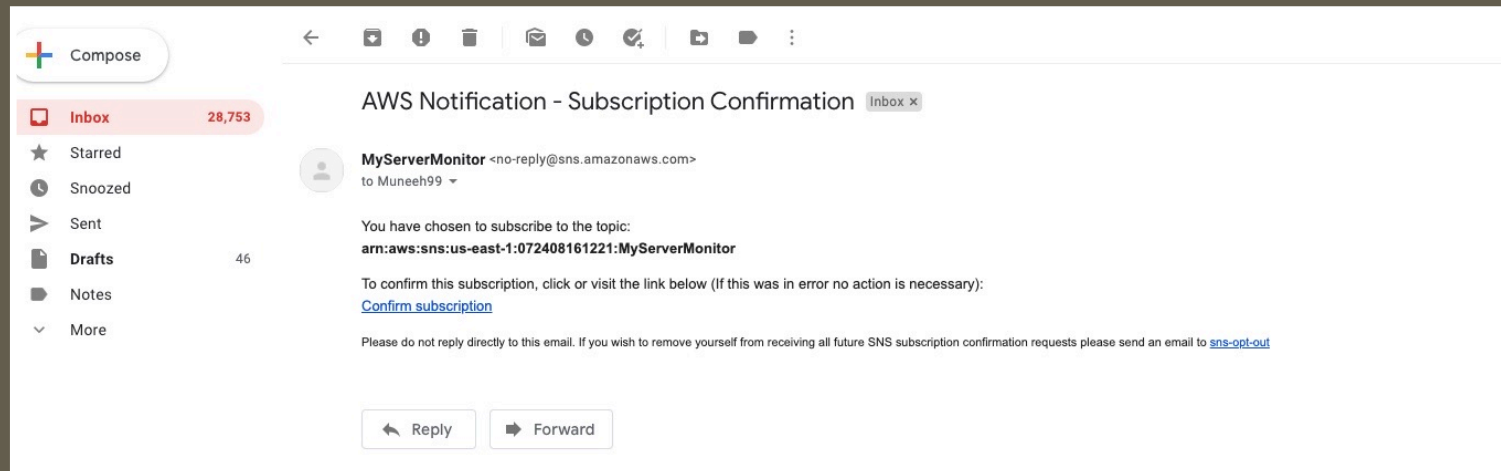
You don't have any subscriptions to this topic.

Create subscription

TASK5

- Task 5 : Subscribe to an SNS Topic
- Once SNS topic is created, click the topic to create subscription
- Under details
- Protocols
- Endpoint
- The email received and confirmation subscription is shown the next slide.

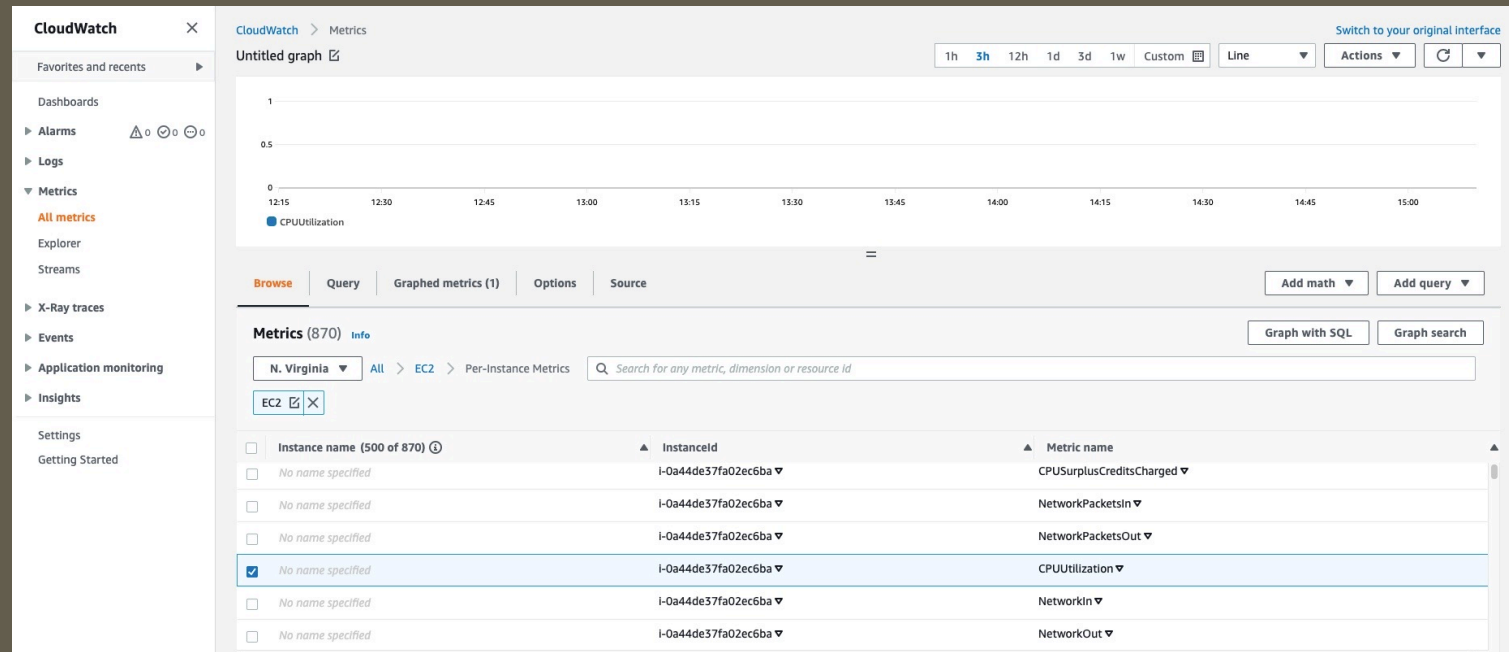
SNS TOPIC EMAIL & CONFIRMATION.



TASK 6

- Task 6: Using CloudWatch to Check EC2 CPU Utilization Metrics in CloudWatch Metrics.
- Navigate to CloudWatch & under management and Governance
- Click all metrics in the panel
- Select EC2 metrics under all metrics
- Wait 5-10 minutes after creation of EC2 to start fetching metrics details
- Select EC2 per-Instance Metrics
- You can see all the various metrics in the next slide.

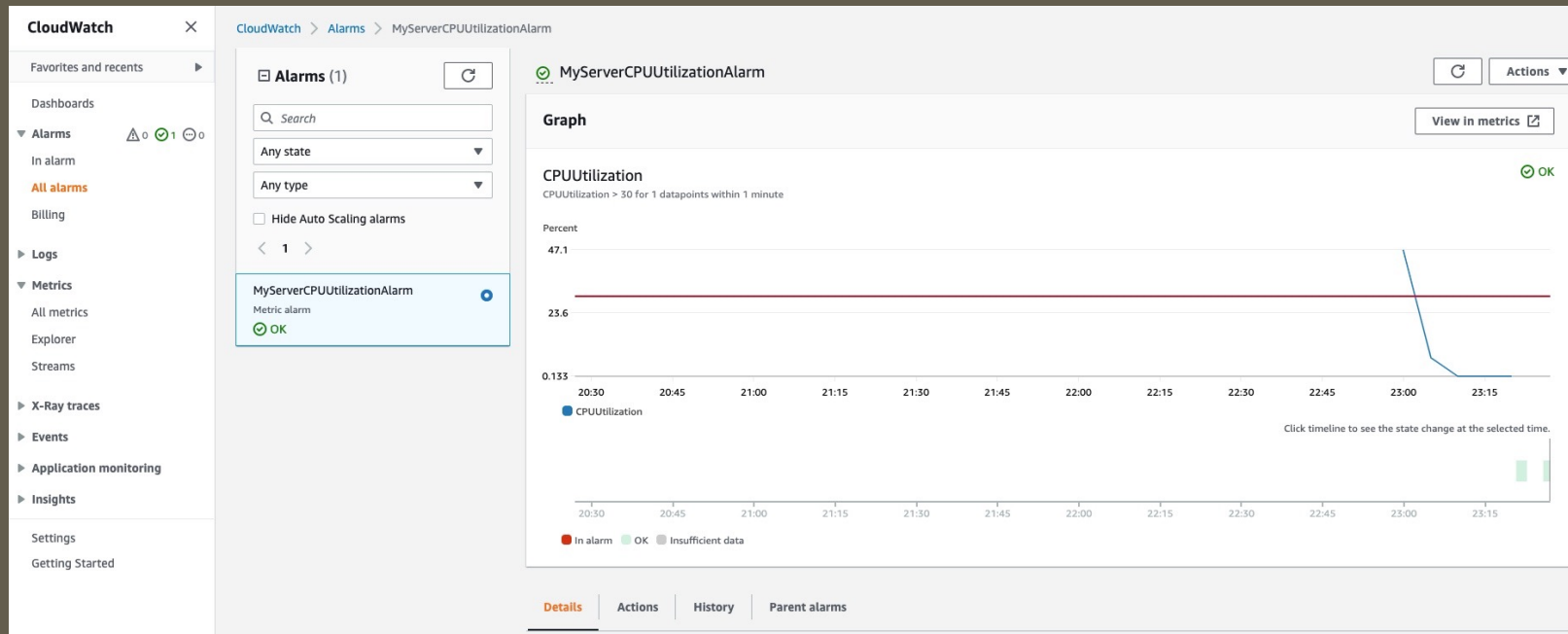
CLOUDWATCH CPU METRICS.



TASK7

- Task 7: Create CloudWatch Alarm
- In Alarms in the left panel of the CloudWatch dashboard
- In the specify metric and conditions page:
- Selected EC2
- Selected per-instance Metrics
- In the search bar; selected CPU Utilization metric
- Click select metric
- Configuration alarm
 - Under metrics: period 1 minute
 - Under condition: Static
 - CPUUtilization is: Choose greater than 30 minute
- Configure action:
- Alarm state trigger: in Alarm
- Select SNS Topic: select an existing SNS topic
- Send a notification to: Choose MyServerMonitor SNS topic created earlier
- A new CloudWatch Alarm is now created.
- Details are shown next slide.

CLOUDWATCH ALARM.



TASK 8

- Task8 :Testing CloudWatch Alarm by Stressing CPU Utilization
- SSH back into the EC2 instance –MyEC2Server
- The stress tool has already been installed, so let's run the below command to increase the CPU Utilization manually
- `sudo stress --cpu 10 -v --timeout 400s`
- This command will monitor the process created by the stress tool
- This runs 6 minutes and 40 seconds
- It will monitor CPU utilization; which should remain very near 100% for that amount of time.
- Details are shown in the next slide.

CLOUDWATCH ALARM BY STRESSING CPU UTILIZATION.

EC2 > Instances > i-0382a98af4f02010f > Connect to instance

Connect to instance Info

Connect to your instance i-0382a98af4f02010f (MyEC2Server) using any of these options

EC2 Instance Connect

Session Manager

SSH client

EC2 serial console

Instance ID

[i-0382a98af4f02010f](#) (MyEC2Server)

- Open an SSH client.
- Locate your private key file. The key used to launch this instance is Myec210.pem
- Run this command, if necessary, to ensure your key is not publicly viewable.
`chmod 400 Myec210.pem`
- Connect to your instance using its Public DNS:
`ec2-3-95-191-85.compute-1.amazonaws.com`

Example:

Command copied

`user@ec2-3-95-191-85.compute-1.amazonaws.com`

Note: In most cases, the guessed user name is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI user name.


```
top - 23:33:38 up 30 min, 2 users, load average: 9.78, 5.24, 2.13
Tasks: 111 total, 11 running, 58 sleeping, 0 stopped, 0 zombie
%Cpu(s):100.0 us, 0.0 sy, 0.0 ni, 0.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
KiB Mem : 988672 total, 384964 free, 87432 used, 516276 buff/cache
KiB Swap: 0 total, 0 free, 0 used, 768988 avail Mem

  PID USER      PR  NI   VIRT   RES   SHR  S %CPU  %MEM    TIME+  COMMAND
 6636 root        20   0   7584    96    0 R 10.0   0.0   0:22.20 stress
 6637 root        20   0   7584    96    0 R 10.0   0.0   0:22.20 stress
 6638 root        20   0   7584    96    0 R 10.0   0.0   0:22.20 stress
 6639 root        20   0   7584    96    0 R 10.0   0.0   0:22.19 stress
 6640 root        20   0   7584    96    0 R 10.0   0.0   0:22.20 stress
 6641 root        20   0   7584    96    0 R 10.0   0.0   0:22.20 stress
 6642 root        20   0   7584    96    0 R 10.0   0.0   0:22.20 stress
 6643 root        20   0   7584    96    0 R 10.0   0.0   0:22.20 stress
 6644 root        20   0   7584    96    0 R 10.0   0.0   0:22.20 stress
 6645 root        20   0   7584    96    0 R 10.0   0.0   0:22.20 stress
 6746 ec2-user    20   0 168956   4552 3888 R  0.3   0.5   0:00.03 top
    1 root        20   0 123612  5560 3892 S  0.0   0.6   0:02.13 systemd
    2 root        20   0      0      0  0 S  0.0   0.0   0:00.00 kthreadd
    3 root        0 -20      0      0  0 I  0.0   0.0   0:00.00 rcu_gp
    4 root        0 -20      0      0  0 I  0.0   0.0   0:00.00 rcu_par_gp
    6 root        0 -20      0      0  0 I  0.0   0.0   0:00.00 kworker/0:0H-ev
    7 root        20   0      0      0  0 I  0.0   0.0   0:00.14 kworker/0:1-cgr
    9 root        0 -20      0      0  0 I  0.0   0.0   0:00.00 mm_percpu_wq
   10 root        20   0      0      0  0 S  0.0   0.0   0:00.00 rcu_tasks_rude_
   11 root        20   0      0      0  0 S  0.0   0.0   0:00.00 rcu_tasks_trace
   12 root        20   0      0      0  0 S  0.0   0.0   0:00.06 ksoftirqd/0
   13 root        20   0      0      0  0 I  0.0   0.0   0:00.07 rcu_sched
   14 root        20   0      0      0  0 S  0.0   0.0   0:00.01 migration/0
   15 root        20   0      0      0  0 S  0.0   0.0   0:00.00 cpuhp/0
   17 root        20   0      0      0  0 S  0.0   0.0   0:00.00 kdevtmpfs
   18 root        0 -20      0      0  0 I  0.0   0.0   0:00.00 netns
   19 root        20   0      0      0  0 I  0.0   0.0   0:00.11 kworker/u38:1-e
   21 root        20   0      0      0  0 S  0.0   0.0   0:00.01 kauditd
   261 root        20   0      0      0  0 S  0.0   0.0   0:00.00 khungtaskd
   262 root        20   0      0      0  0 S  0.0   0.0   0:00.00 oom_reaper
```


TASK 9

- Task 9 : Checking For an Email from the SNS Topic
- Navigate to your mailbox and refresh it
- You should be able to see a new email notification for MyServerCPUUtilizationAlarm
- In the next slide I can see that mail I received contains details about CloudWatch Alarm.(name of the alarm, when it was triggered, etc.)

EMAIL NOTIFICATION FOR SNS TOPIC

 **MyServerMonitor** <no-reply@sns.amazonaws.com>
to Muneeh99 ▾

5:35 PM (12 minutes ago) ☆ ↶ ⋮

You are receiving this email because your Amazon CloudWatch Alarm "MyServerCPUUtilizationAlarm" in the US East (N. Virginia) region has entered the ALARM state, because "Threshold Crossed: 1 out of the last 1 datapoints [99.43876076687968 (17/07/22 23:30:00)] was greater than the threshold (30.0) (minimum 1 datapoint for OK -> ALARM transition)." at "Sunday 17 July, 2022 23:35:48 UTC".

View this alarm in the AWS Management Console:
<https://us-east-1.console.aws.amazon.com/cloudwatch/deeplink.js?region=us-east-1#alarmsV2:alarm/MyServerCPUUtilizationAlarm>

Alarm Details:

- Name: MyServerCPUUtilizationAlarm
- Description:
- State Change: INSUFFICIENT_DATA -> ALARM
- Reason for State Change: Threshold Crossed: 1 out of the last 1 datapoints [99.43876076687968 (17/07/22 23:30:00)] was greater than the threshold (30.0) (minimum 1 datapoint for OK -> ALARM transition).
- Timestamp: Sunday 17 July, 2022 23:35:48 UTC
- AWS Account: 902494929116
- Alarm Arn: arn:aws:cloudwatch:us-east-1:902494929116:alarm:MyServerCPUUtilizationAlarm

Threshold:

- The alarm is in the ALARM state when the metric is GreaterThanThreshold 30.0 for at least 1 of the last 1 period(s) of 60 seconds.

Monitored Metric:

- MetricNamespace: AWS/EC2
- MetricName: CPUUtilization
- Dimensions: [InstanceId = i-0382a98af4f02010f]
- Period: 60 seconds
- Statistic: Average
- Unit: not specified
- TreatMissingData: missing

State Change Actions:

- OK:
- ALARM: [arn:aws:sns:us-east-1:902494929116:MyServerMonitor]
- INSUFFICIENT_DATA:

--

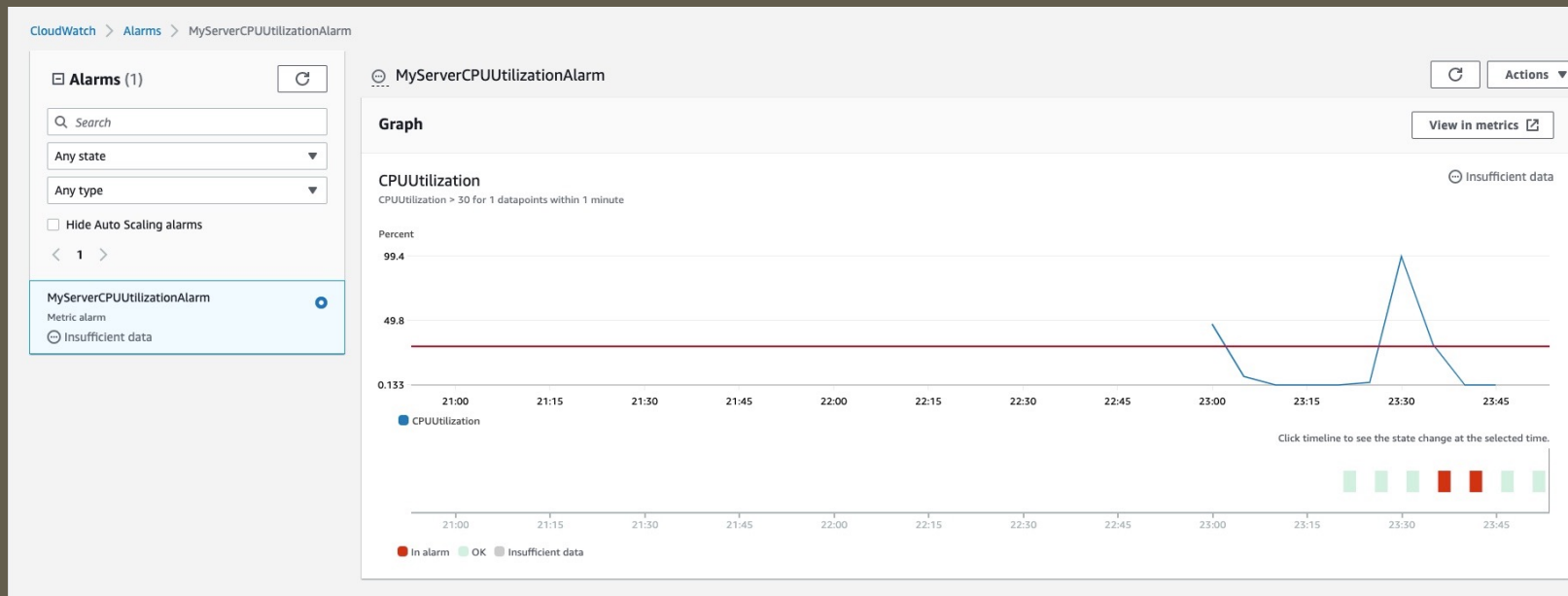
If you wish to stop receiving notifications from this topic, please click or visit the link below to unsubscribe:
<https://sns.us-east-1.amazonaws.com/unsubscribe.html?SubscriptionArn=arn:aws:sns:us-east-1:902494929116:MyServerMonitor:f5ec113e-85c4-4dfe-9d0e-94bb0b950c8e&Endpoint=Muneeh99@gmail.com>

Please do not reply directly to this email. If you have any questions or comments regarding this email, please contact us at <https://aws.amazon.com/support>

TASK 10

- Task 10 : Checking the CloudWatch Alarm Graph
- I can trigger CPUUtilization multiple times to see the spike on the graph
- Then, I have successfully triggered a CloudWatch Alarm for CPUUtilization.
- Next slide shows more of those details.

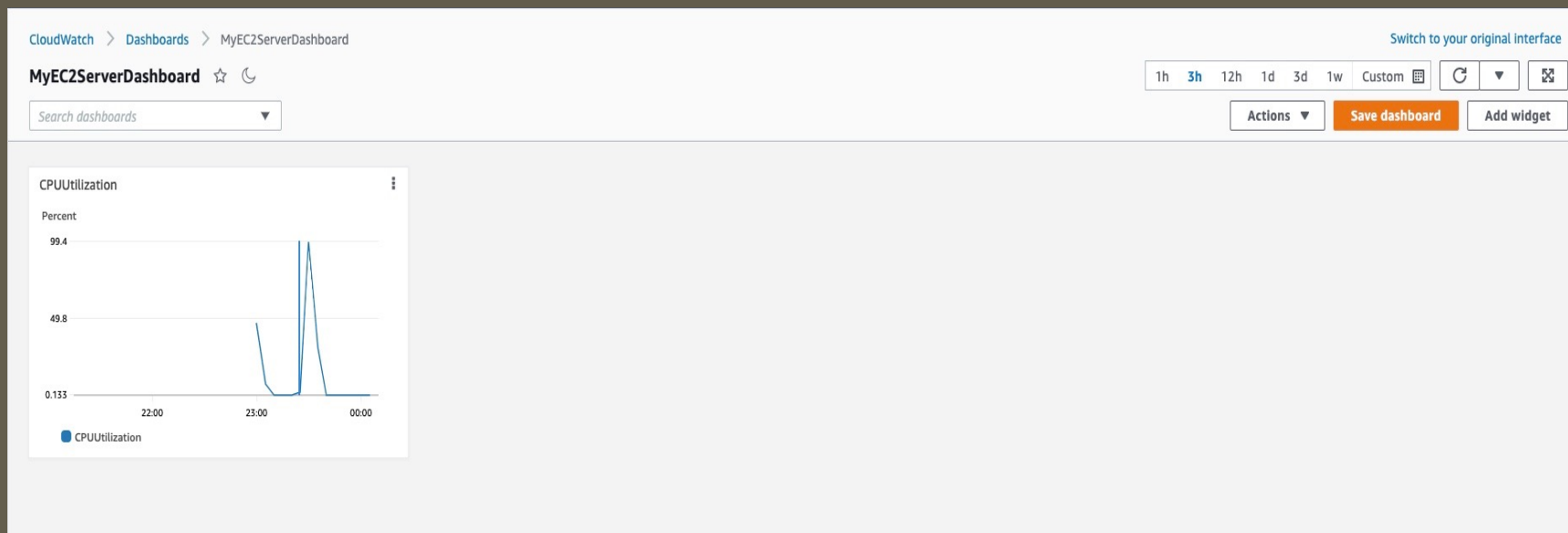
CHECKING THE ALARM GRAPH.



TASK I I

- Task I I: Create a CloudWatch Dashboard
- Dashboard Name: MyEC2ServerDashboard
- Add widget: select line graph
- Select: Metrics
- Choose EC2 under metrics tab. Choose per-instance Metrics
- Click create Widget
- Next slide for more details.

CLOUDWATCH DASHBOARD.



COMPLETION AND CONCLUSION.

1. I have created an EC2 Instance for which CloudWatch Monitoring will be carried out.
2. I have successfully created an Amazon SNS Topic used by CloudWatch.
3. I have successfully subscribed to SNS topic using your email address.
4. I have used CloudWatch to see CPUUtilization Metrics using CloudWatch Metrics.
5. I have successfully created and triggered a CloudWatch Alarm based on the CPUUtilization Metric.