

# AWS - Cloud Watch Practical

Name: Akash Ghuge | Date: 09/09/2024

## Practical:1

Monitor the CPU utilization of instance, set the threshold greater than and equal to 50% & send alert via SNS

### 1) Create new instance

The screenshot shows the AWS Management Console 'Instances' page. The instance 'i-02c4ce931b618af61' (Cloudwatch Practical) is in the 'Running' state. The instance details panel shows the instance ID, public IPv4 address (3.110.47.167), and public IPv4 DNS name (ec2-3-110-47-167.ap-south-1.compute.amazonaws.com).

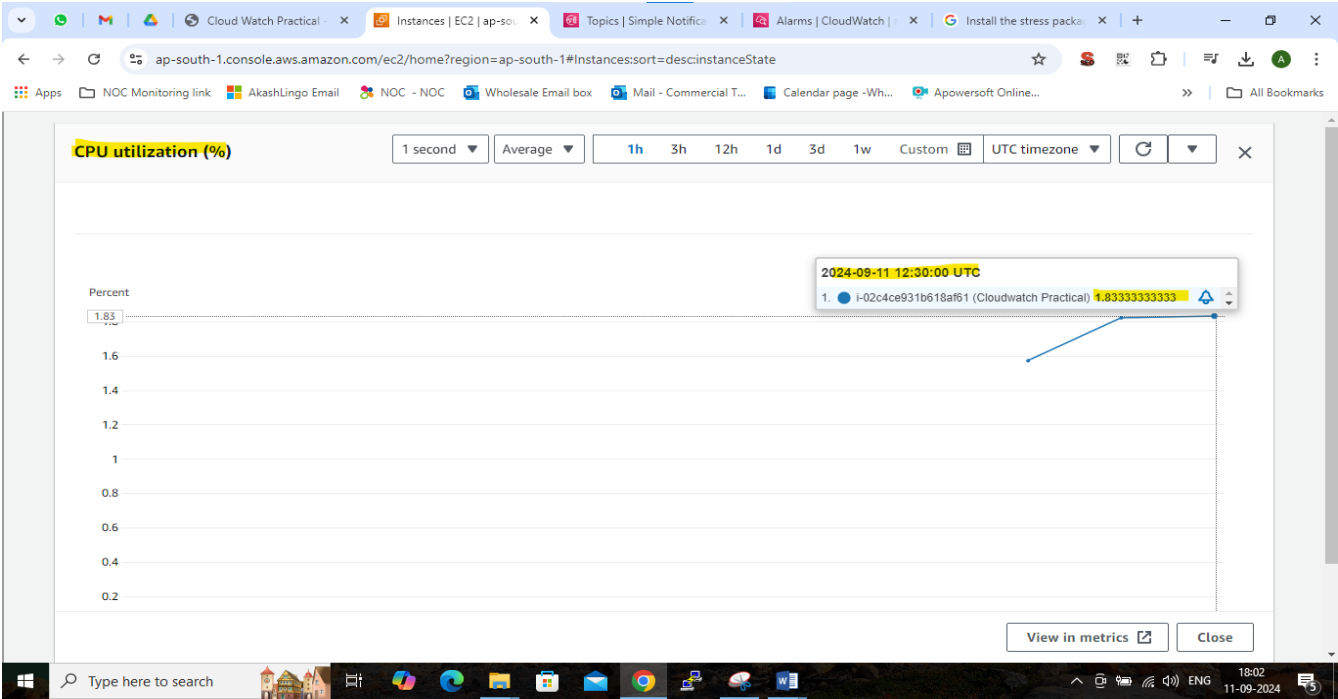
### a) Initially no alarm for new instance:

The screenshot shows the 'Alarm details for i-02c4ce931b618af61' dialog. The dialog indicates that the instance has no associated alarms.

### b) Monitor all below parameter for instance:

The screenshot shows the 'Monitoring' page for instance 'i-02c4ce931b618af61'. The page displays various metrics including CPU utilization, network in/out, and network packets in/out.

c) Current CPU Utilization | By default, basic monitoring is unable for 5 min frequency



2) Create a topic by using SNS (Simple Notification Service)

a) Create Topic

The screenshot shows the AWS Management Console for creating a new SNS topic. The 'Create topic' wizard is displayed, with the 'Standard' topic type selected. The topic name is 'MySNSTopic1'. The 'Display name' is also 'MySNSTopic1'. The 'Standard' topic type is selected, which provides best-effort message ordering, at-least once message delivery, high throughput (up to 300 publishes/second), and supports subscription protocols: SQS, Lambda, HTTP, SMS, email, and mobile application endpoints. The 'Name' field is 'MySNSTopic1' and the 'Display name' field is 'MySNSTopic1'. The 'Maximum 256 characters. Can include alphanumeric characters, hyphens (-) and underscores (\_).' and 'Maximum 100 characters.' constraints are shown. Below the wizard, the 'MySNSTopic1' topic details are shown, including the Name, Display name, ARN, Topic owner, and Type (Standard). The 'Subscriptions' section shows 0 subscriptions, with buttons for 'Edit', 'Delete', 'Request confirmation', 'Confirm subscription', and 'Create subscription'.

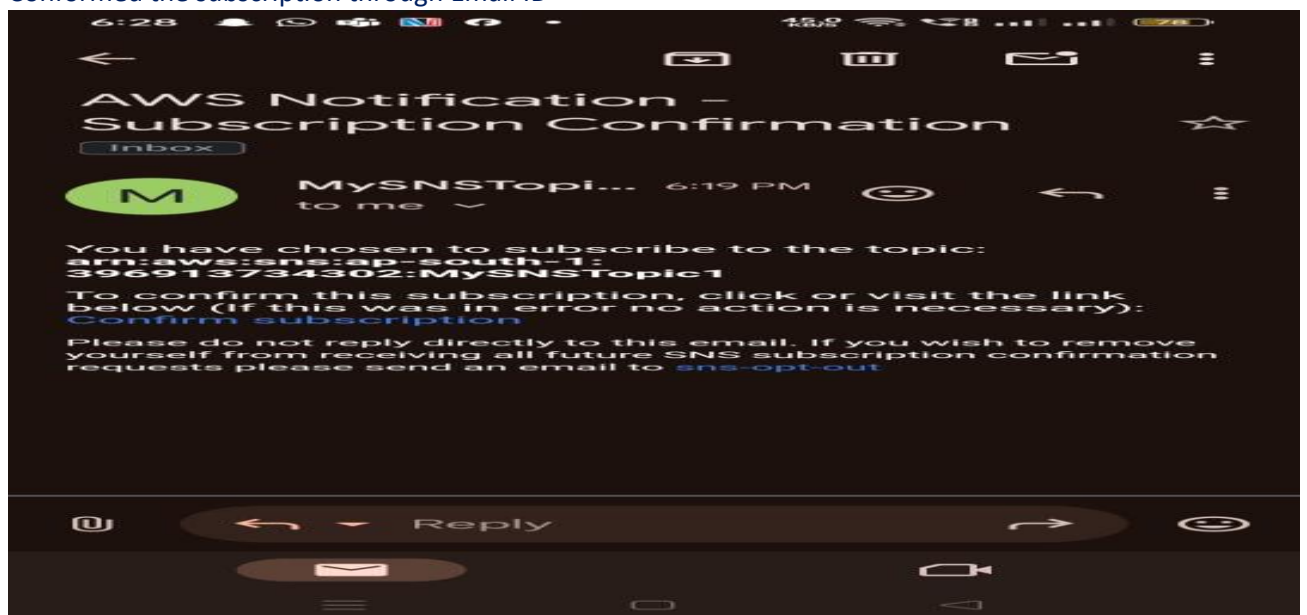
## b) Create subscription

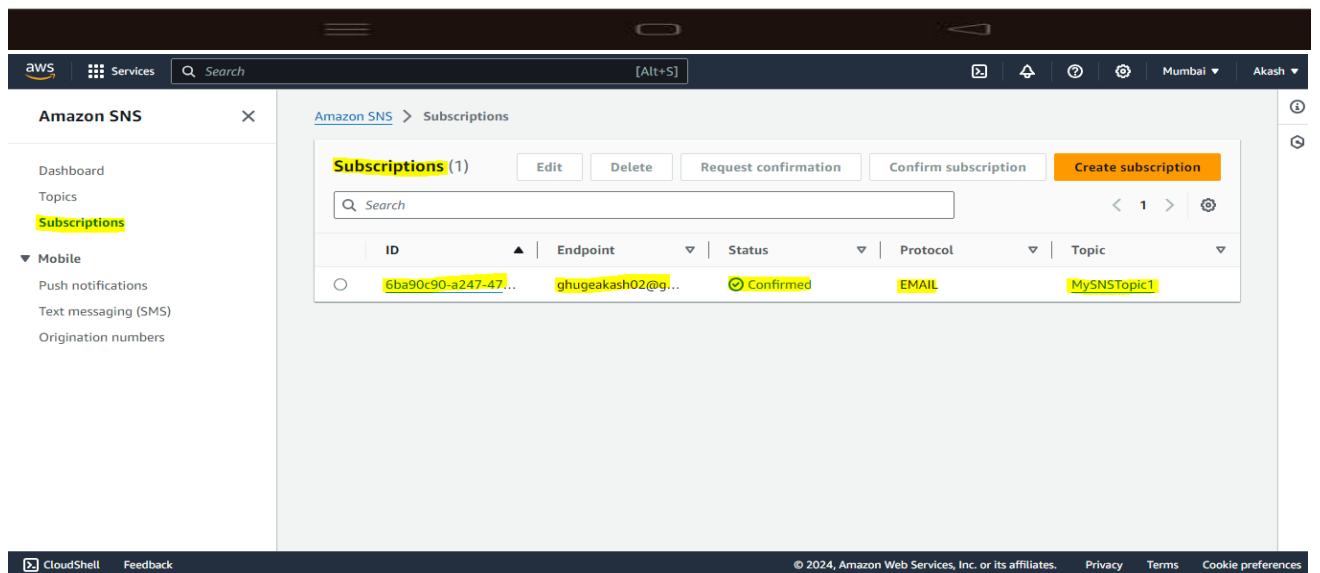
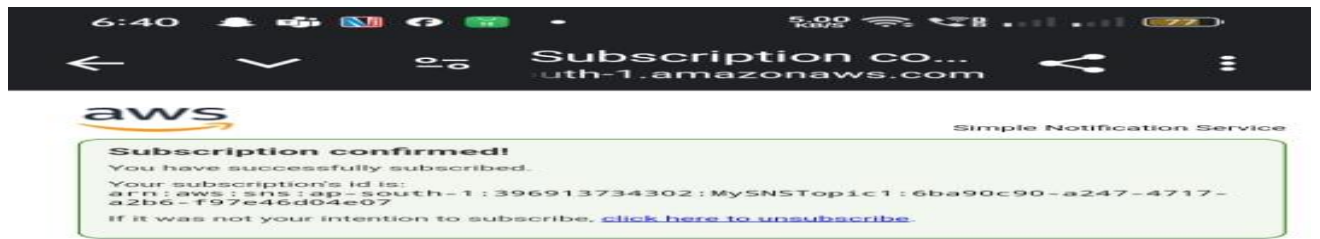
The screenshot shows the 'Create subscription' page in the AWS console. The 'Topic ARN' field is populated with 'arn:aws:sns:ap-south-1:396913734302:MySNSTopic1'. The 'Protocol' dropdown is set to 'Email'. The 'Endpoint' field contains 'ghugeakash02@gmail.com'. A blue information box states: 'After your subscription is created, you must confirm it. Info'. At the bottom, there is a link for 'Subscription filter policy - optional Info'.

The screenshot shows the 'Subscriptions' page in the AWS console. A green banner at the top indicates 'Subscription deleted successfully.' The table below lists the subscription details:

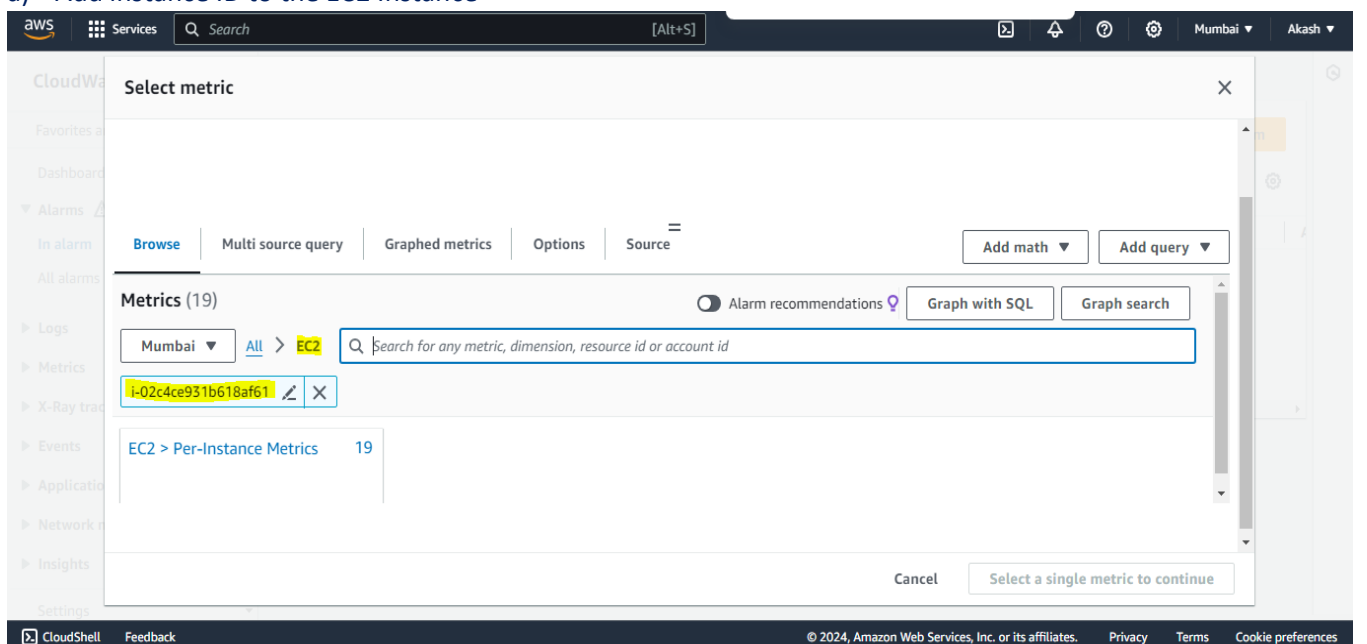
	ID	Endpoint	Status	Protocol	Topic
<input type="radio"/>	Pending confirmation	ghugeakash02@gm...	Pending confirmation	EMAIL	<a href="#">MySNSTopic1</a>

## c) Confirmed the subscription through Email ID





- 3) Create alarm by using cloud watch service – For CPU Utilization
  - a) Add instance ID to the EC2 Instance



## b) Select Metric - CPU Utilization

The screenshot shows the 'Select metric' dialog in the AWS CloudWatch console. The 'Browse' tab is selected, displaying a list of metrics for the instance i-02c4ce931b618... The 'CPUUtilization' metric is highlighted. The 'Source' is 'Cloudwatch Practical'. The 'Add math' and 'Add query' buttons are visible at the top right of the dialog.

<input type="checkbox"/>	Cloudwatch Practical	i-02c4ce931b618...	CPU surplus credit balance	No alarms
<input type="checkbox"/>	Cloudwatch Practical	i-02c4ce931b618...	CPU credit usage	No alarms
<input type="checkbox"/>	Cloudwatch Practical	i-02c4ce931b618...	CPU surplus credits charged	No alarms
<input checked="" type="checkbox"/>	Cloudwatch Practical	i-02c4ce931b618...	CPU utilization	No alarms
<input type="checkbox"/>	Cloudwatch Practical	i-02c4ce931b618...	Status check failed system	No alarms

## c) Create rule

The screenshot shows the 'Specify metric and conditions' dialog in the AWS CloudWatch console. The 'Metric' section displays a graph of CPU utilization over time, with a threshold line at 50. The 'Conditions' section shows the 'Greater/Equal' threshold type, with a threshold value of 50.

**Metric**

Graph: This alarm will trigger when the blue line goes above the red line for 1 datapoints within 2 minutes.

Percent: 50, 25.0, 1.57

Namespace: AWS/EC2

Metric name: CPUUtilization

InstanceId: i-02c4ce931b618a61

Instance name: Cloudwatch Practical

Statistic: Average

Period: 2 minutes

**Conditions**

Threshold type: Static

Whenever CPUUtilization is... Define the alarm condition.

Greater > threshold, Greater/Equal >= threshold, Lower/Equal <= threshold, Lower < threshold

Then... Define the threshold value. 50

## d) Add Notification ID

The screenshot shows the 'Notification' dialog in the AWS CloudWatch console. The 'Alarm state trigger' section shows the 'In alarm' state selected. The 'Send a notification to the following SNS topic' section shows the 'Select an existing SNS topic' option selected, with 'MySNSTopic1' chosen. The 'Email (endpoints)' section shows the email address 'ghugeakash02@gmail.com'.

**Notification**

Alarm state trigger: Define the alarm state that will trigger this action.

☒ In alarm: The metric or expression is outside of the defined threshold.

☐ OK: The metric or expression is within the defined threshold.

☐ Insufficient data: The alarm has just started or not enough data is available.

Send a notification to the following SNS topic: Define the SNS (Simple Notification Service) topic that will receive the notification.

☒ Select an existing SNS topic

☐ Create new topic

☐ Use topic ARN to notify other accounts

Send a notification to...: MySNSTopic1

Only topics belonging to this account are listed here. All persons and applications subscribed to the selected topic will receive notifications.

Email (endpoints): ghugeakash02@gmail.com - View in SNS Console

e) Alarm condition: CPUUtilization >= 50 for 1 datapoints within 2 minutes

The screenshot displays the AWS CloudWatch Alarms console. The left sidebar shows the navigation menu with options like Dashboards, Alarms, Logs, Metrics, X-Ray traces, Events, Application Signals, Network monitoring, and Insights. The main panel shows a list of alarms. One alarm is selected: 'CPU\_Utilization\_i-02c4ce931b618af61'. The alarm's state is 'OK', and its last state update is '2024-09-11 13:35:38'. The conditions are 'CPUUtilization >= 50 for 1 datapoints within 2 minutes'. Below the list, a graph shows the CPU Utilization percentage over time. The graph has a red line at 50% and a blue line representing the actual CPU Utilization. The blue line is below the red line, indicating that the alarm is not currently active. A tooltip shows the current value: '2024-09-11 13:24 UTC CPUUtilization 1.9366666667'.

4) Connected with Instance to increase the CPU Utilization

a) Connect with Instance

The screenshot displays the AWS EC2 Instances console. The left sidebar shows the navigation menu with options like EC2 Dashboard, EC2 Global View, Events, Instances, Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity, and Reservations. The main panel shows a list of instances. One instance is selected: 'Cloudwatch Pr... i-02c4ce931b618af61'. The instance's state is 'Running', and its status check is '2/2 checks passed'. Below the list, the 'Monitoring' tab is selected, showing the 'CPU utilization (%)' metric. The graph shows the CPU utilization percentage over time. The graph has a red line at 50% and a blue line representing the actual CPU Utilization. The blue line is below the red line, indicating that the alarm is not currently active. A tooltip shows the current value: '2024-09-11 13:24 UTC CPUUtilization 1.9366666667'.

b) #Top command is used to check the current CPU utilization of the instance/Server in Linux

#sudo su - -> Interring root user

#top -> To check the current CPU utilization of the instance/Server in Linux

```
Tasks: 105 total, 1 running, 104 sleeping, 0 stopped, 0 zombie
%Cpu(s): 1.0 us, 2.3 sy, 0.0 ni, 96.3 id, 0.0 wa, 0.0 hi, 0.3 si, 0.0 st
MiB Mem : 949.5 total, 595.3 free, 129.0 used, 225.2 buff/cache
MiB Swap: 0.0 total, 0.0 free, 0.0 used, 683.1 avail Mem

  PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM    TIME+  COMMAND
    1 root        20   0 105192 16500 10164 S   0.0   1.7   0:00.75 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
    5 root        0 -20     0     0     0 I   0.0   0.0   0:00.00 slab_flushwq
    6 root        0 -20     0     0     0 I   0.0   0.0   0:00.00 netns
    8 root        0 -20     0     0     0 I   0.0   0.0   0:00.00 kworker/0:0H-events_highpri
   10 root        0 -20     0     0     0 I   0.0   0.0   0:00.00 mm_percpu_wq
   11 root        20   0     0     0     0 I   0.0   0.0   0:00.00 rcu_tasks_kthread
   12 root        20   0     0     0     0 I   0.0   0.0   0:00.00 rcu_tasks_rude_kthread
   13 root        20   0     0     0     0 I   0.0   0.0   0:00.00 rcu_tasks_trace_kthread
   14 root        20   0     0     0     0 S   0.0   0.0   0:00.13 ksoftirqd/0
   15 root        20   0     0     0     0 I   0.0   0.0   0:00.04 rcu_preempt
   16 root        rt    0     0     0     0 S   0.0   0.0   0:00.02 migration/0
   18 root        20   0     0     0     0 S   0.0   0.0   0:00.00 cpuhp/0
[root@ip-172-31-10-46 ~]# yum install stress
```

i-02c4ce931b618af61 (Cloudwatch Practical)

PublicIPs: 3.110.47.167 PrivateIPs: 172.31.10.46

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

c) #yum install stress -y :-> To Increase the server utilization

```
aws Services Q Search [Alt+S] Mumbai Akash
18 root 20 0 0 0 0 0 0.0 0.0 0:00.00 cpuhp/0
[root@ip-172-31-10-46 ~]# yum install stress -y
Last metadata expiration check: 1:33:56 ago on Wed Sep 11 12:19:53 2024.
Dependencies resolved.

Package Architecture Version Repository Size
Installing:
stress x86_64 1.0.7-2.amzn2023.0.1 amazonlinux 34 k

Transaction Summary
Install 1 Package

Total download size: 34 k
Installed size: 68 k
Downloading Packages:
stress-1.0.7-2.amzn2023.0.1.x86_64.rpm 522 kB/s | 34 kB 00:00
Total 317 kB/s | 34 kB 00:00
Running transaction check
Transaction check succeeded.
```

i-02c4ce931b618af61 (Cloudwatch Practical)

PublicIPs: 3.110.47.167 PrivateIPs: 172.31.10.46

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

#stress - -cpu 20 - -timeout 1000

```
aws Services Q Search [Alt+S] Mumbai Akash
Installed size: 68 k
Downloading Packages:
stress-1.0.7-2.amzn2023.0.1.x86_64.rpm 522 kB/s | 34 kB 00:00
Total 317 kB/s | 34 kB 00:00
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
Preparing : stress-1.0.7-2.amzn2023.0.1.x86_64 1/1
Installing : stress-1.0.7-2.amzn2023.0.1.x86_64 1/1
Running scriptlet: stress-1.0.7-2.amzn2023.0.1.x86_64 1/1
Verifying : stress-1.0.7-2.amzn2023.0.1.x86_64 1/1
Installed:
stress-1.0.7-2.amzn2023.0.1.x86_64
Complete!
[root@ip-172-31-10-46 ~]# stress --cpu 20 --timeout 1000
stress: info: [28779] dispatching hogs: 20 cpu, 0 io, 0 vm, 0 hdd
```

i-02c4ce931b618af61 (Cloudwatch Practical)

PublicIPs: 3.110.47.167 PrivateIPs: 172.31.10.46

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Services

Search

[Alt+S]

top - 14:04:49 up 1:45, 4 users, load average: 19.98, 14.86, 7.09

Tasks: 132 total, 21 running, 111 sleeping, 0 stopped, 0 zombie

%Cpu(s): 99.3 us, 0.0 sy, 0.0 ni, 0.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.7 st

Mem Mem : 949.5 total, 516.3 free, 144.1 used, 289.2 buff/cache

Mem Swap: 0.0 total, 0.0 free, 0.0 used, 663.2 avail Mem

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
28780	root	20	0	3512	108	0	R	5.0	0.0	0:20.02	stress
28781	root	20	0	3512	108	0	R	5.0	0.0	0:20.02	stress
28782	root	20	0	3512	108	0	R	5.0	0.0	0:20.02	stress
28783	root	20	0	3512	108	0	R	5.0	0.0	0:20.02	stress
28784	root	20	0	3512	108	0	R	5.0	0.0	0:20.02	stress
28785	root	20	0	3512	108	0	R	5.0	0.0	0:20.02	stress
28786	root	20	0	3512	108	0	R	5.0	0.0	0:20.02	stress
28787	root	20	0	3512	108	0	R	5.0	0.0	0:20.01	stress
28788	root	20	0	3512	108	0	R	5.0	0.0	0:20.02	stress
28789	root	20	0	3512	108	0	R	5.0	0.0	0:20.01	stress
28790	root	20	0	3512	108	0	R	5.0	0.0	0:20.02	stress
28791	root	20	0	3512	108	0	R	5.0	0.0	0:20.02	stress
28792	root	20	0	3512	108	0	R	5.0	0.0	0:20.02	stress
28793	root	20	0	3512	108	0	R	5.0	0.0	0:20.01	stress
28794	root	20	0	3512	108	0	R	5.0	0.0	0:20.02	stress

i-02c4ce931b618af61 (Cloudwatch Practical)

PublicIPs: 3.110.47.167 PrivateIPs: 172.31.10.46

CloudShell Feedback

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

#stress - -cpu 52 - -timeout 1000

Services

Search

[Alt+S]

Total 317 kB/s | 34 kB 00:00

Running transaction check

Transaction check succeeded.

Running transaction test

Transaction test succeeded.

Running transaction

Preparing : 1/1

Installing : stress-1.0.7-2.amzn2023.0.1.x86\_64 1/1

Running scriptlet: stress-1.0.7-2.amzn2023.0.1.x86\_64 1/1

Verifying : stress-1.0.7-2.amzn2023.0.1.x86\_64 1/1

Installed:

stress-1.0.7-2.amzn2023.0.1.x86\_64

Complete!

[root@ip-172-31-10-46 ~]# stress --cpu 20 --timeout 1000

stress: info: [28779] dispatching hogs: 20 cpu, 0 io, 0 vm, 0 hdd

^C

[root@ip-172-31-10-46 ~]# stress --cpu 52 --timeout 1000

stress: info: [29550] dispatching hogs: 52 cpu, 0 io, 0 vm, 0 hdd

i-02c4ce931b618af61 (Cloudwatch Practical)

PublicIPs: 3.110.47.167 PrivateIPs: 172.31.10.46

CloudShell Feedback

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Services

Search

[Alt+S]

top - 14:11:33 up 1:52, 4 users, load average: 51.59, 36.40, 18.90

Tasks: 164 total, 53 running, 111 sleeping, 0 stopped, 0 zombie

%Cpu(s): 5.5 us, 0.0 sy, 0.0 ni, 0.0 id, 0.0 wa, 0.0 hi, 0.0 si, 94.5 st

Mem Mem : 949.5 total, 512.9 free, 147.4 used, 289.2 buff/cache

Mem Swap: 0.0 total, 0.0 free, 0.0 used, 659.9 avail Mem

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
29590	root	20	0	3512	108	0	R	3.8	0.0	0:05.05	stress
29592	root	20	0	3512	108	0	R	3.8	0.0	0:05.05	stress
29578	root	20	0	3512	108	0	R	3.5	0.0	0:05.05	stress
29553	root	20	0	3512	108	0	R	3.2	0.0	0:05.04	stress
29577	root	20	0	3512	108	0	R	3.2	0.0	0:05.06	stress
29580	root	20	0	3512	108	0	R	3.2	0.0	0:05.04	stress
29565	root	20	0	3512	108	0	R	2.9	0.0	0:05.03	stress
29566	root	20	0	3512	108	0	R	2.9	0.0	0:05.04	stress
29568	root	20	0	3512	108	0	R	2.9	0.0	0:05.04	stress
29556	root	20	0	3512	108	0	R	2.6	0.0	0:05.01	stress
29558	root	20	0	3512	108	0	R	2.6	0.0	0:05.03	stress
29559	root	20	0	3512	108	0	R	2.6	0.0	0:05.01	stress
29560	root	20	0	3512	108	0	R	2.6	0.0	0:05.05	stress
29561	root	20	0	3512	108	0	R	2.6	0.0	0:05.06	stress
29569	root	20	0	3512	108	0	R	2.6	0.0	0:05.02	stress

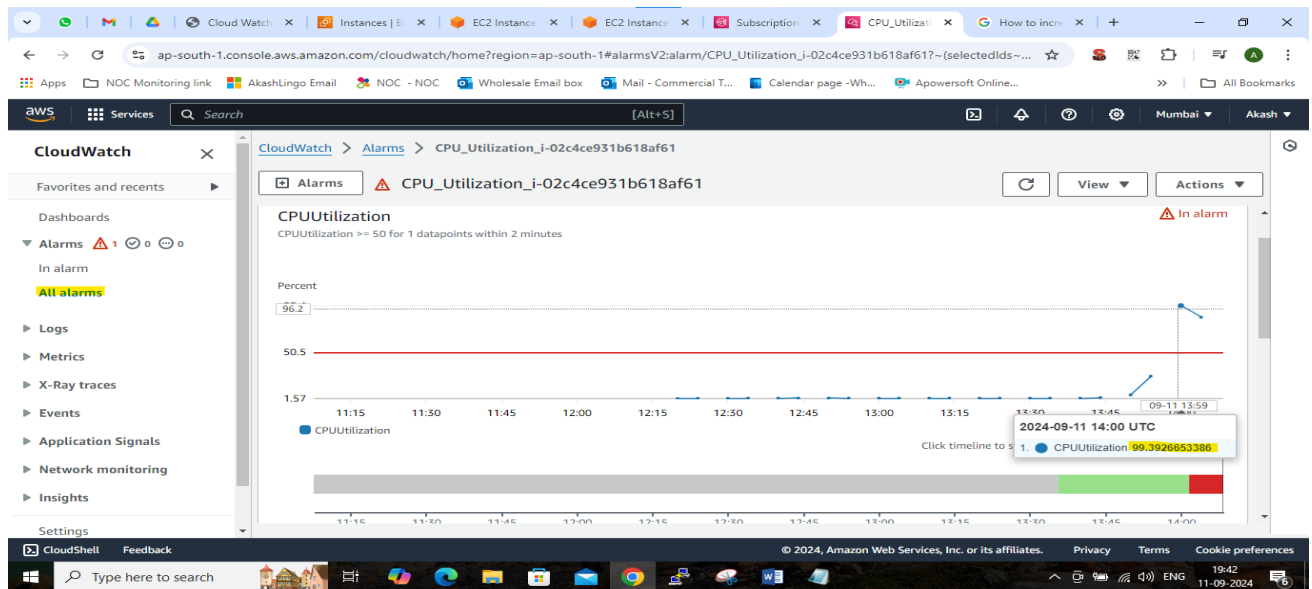
i-02c4ce931b618af61 (Cloudwatch Practical)

PublicIPs: 3.110.47.167 PrivateIPs: 172.31.10.46

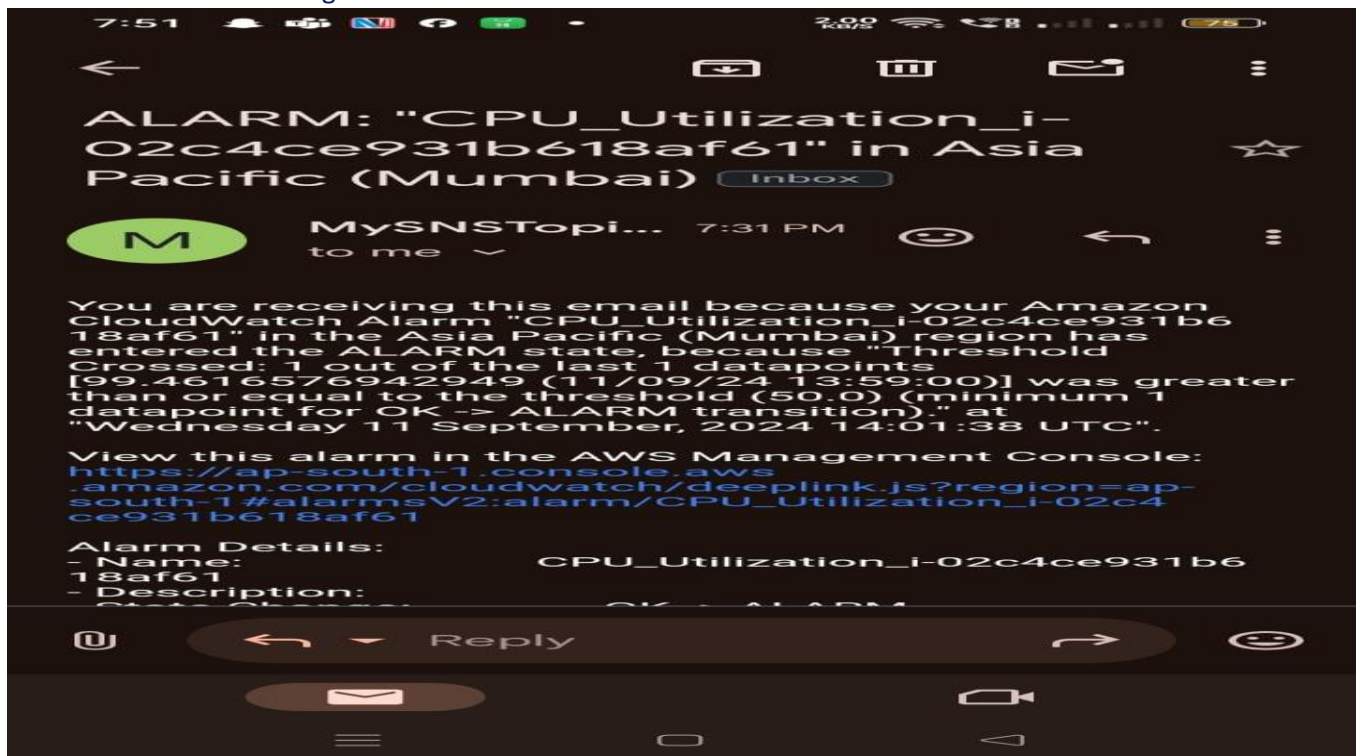
CloudShell Feedback

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

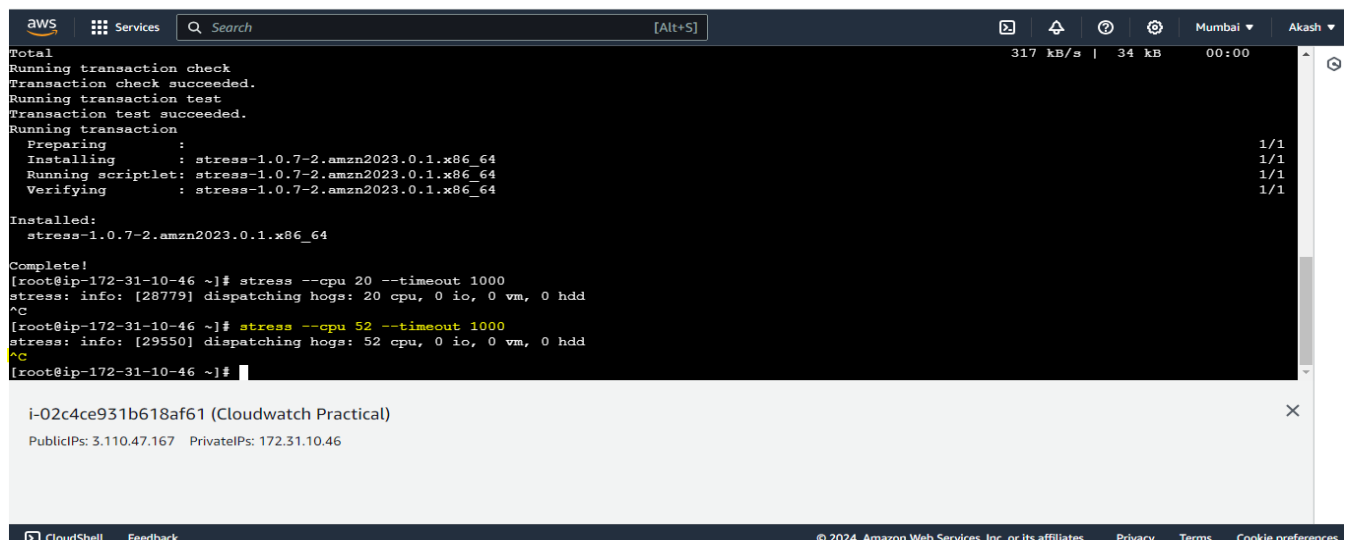




5) We received the alarm on given email ID



6) Control C – we canceled the load -> Reduces the CPU utilization



aws Services Search [Alt+S] Mumbai Akash

```
top - 14:18:42 up 1:59, 4 users, load average: 3.61, 26.81, 22.99
Tasks: 109 total, 1 running, 108 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.0 us, 0.0 sy, 0.0 ni, 99.0 id, 0.0 wa, 0.0 hi, 0.0 si, 1.0 st
MiB Mem : 949.5 total, 512.7 free, 147.6 used, 289.2 buff/cache
MiB Swap: 0.0 total, 0.0 free, 0.0 used. 659.7 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
29976	root	20	0	223908	3356	2704	R	0.3	0.3	0:00.02	top
1	root	20	0	105192	16540	10164	S	0.0	1.7	0:00.85	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
5	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	slub_flushwq
6	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	netns
8	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/0:0H-events_highpri
10	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	mm_percpu_wq
11	root	20	0	0	0	0	I	0.0	0.0	0:00.00	rcu_tasks_kthread
12	root	20	0	0	0	0	I	0.0	0.0	0:00.00	rcu_tasks_rude_kthread
13	root	20	0	0	0	0	I	0.0	0.0	0:00.00	rcu_tasks_trace_kthread
14	root	20	0	0	0	0	S	0.0	0.0	0:00.31	ksoftirqd/0
15	root	20	0	0	0	0	I	0.0	0.0	0:00.07	rcu_preempt
16	root	rt	0	0	0	0	S	0.0	0.0	0:00.03	migration/0

i-02c4ce931b618af61 (Cloudwatch Practical)

PublicIPs: 3.110.47.167 PrivateIPs: 172.31.10.46

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

ap-south-1.console.aws.amazon.com/cloudwatch/home?region=ap-south-1#alarmsV2:alarm/CPU\_Utilization\_i-02c4ce931b618af61?~(select...

CloudWatch Alarms CPU\_Utilization\_i-02c4ce931b618af61

CPUUtilization CPUUtilization >= 50 for 1 datapoints within 2 minutes

Percent 99.4 50.5 6.38

11:30 11:45 12:00 12:15 12:30 12:45 13:00 13:15 13:30 13:45 14:00 09-11 14:14 UTC

Click timeline to 1. CPUUtilization 6.36041555583

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Type here to search

## Terminate the Instance:

ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#instances:v=3:\$case=tags:true%5Cclient:false:\$regex=tags:false%5...

Instances (1/2) info

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IP
Cloudwatch Practical	i-02c4ce931b618af61	Running	t2.micro	2/2 checks passed	View alarms	ap-south-1b	ec2-3-1
EC2-Cloudwatch-Role	i-0a5ef07d936ea024a	Running	t2.micro	Initializing	View alarms	ap-south-1b	ec2-3-1

Actions: Launch instances, Stop instance, Reboot instance, Hibernate instance, Terminate instance

i-02c4ce931b618af61 (Cloudwatch Practical)

Details Status and alarms Monitoring Security Networking Storage Tags

Instance summary info

Instance ID: i-02c4ce931b618af61 (Cloudwatch Practical)

IPV4 address: 3.110.47.167

Hostname type: ip-172-31-10-46.ap-south-1.compute.internal

Private IPV4 address: 172.31.10.46

Private IPV4 DNS: ec2-3-110-47-167.ap-south-1.compute.amazonaws.com

Instance type: t2.micro

VPC ID: vpc-0a5ef07d

EC2 Dashboard

EC2 Global View

Events

Instances

Instance Types

Launch Templates

Spot Requests

Savings Plans

Reserved Instances

Dedicated Hosts

Capacity

Reservations

Images

AMIs

AMI Catalog

Elastic Block Store

Volumes

Snapshots

Lifecycle Manager

Network & Security

Security Groups

Elastic IPs

Placement Groups

Key Pairs

Network Interfaces

Services

Search

[Alt+S]

Mumbai

Akash

Successfully initiated termination (deletion) of i-02c4ce931b618af61

Instances (1/2) info

Last updated less than a minute ago

Connect

Instance state

Actions

Launch instances

Find instance by attribute or tag (case-sensitive)

All states

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP	IPv
<input checked="" type="checkbox"/>	Cloudwatch Practical	i-02c4ce931b618af61	Terminated	t2.micro	--	View alarms	ap-south-1b	--	--	--	--
<input type="checkbox"/>	EC2-Cloudwatch-Role	i-0e3ef07d93ea4024a	Running	t2.micro	2/2 checks passed	View alarms	ap-south-1b	ec2-5-110-45-6.ap-sout...	3.110.45.6	--	--

i-02c4ce931b618af61 (Cloudwatch Practical)

Details | Status and alarms | Monitoring | Security | Networking | Storage | Tags

Instance summary info

Instance ID

i-02c4ce931b618af61 (Cloudwatch Practical)

IPv6 address

--

Hostname type

--

Answer private resource DNS name

--

Auto-assigned IP address

--

Public IPv4 address

--

Instance state

Terminated

Instance type

t2.micro

VPC ID

--

Private IPv4 addresses

--

Public IPv4 DNS

--

Elastic IP addresses

--

AWS Compute Optimizer finding

--

CloudShell

Feedback

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

## Practical:2

We have going to monitor the Disk, CPU, Memory utilization of the server

1. Create EC2 IAM Role (Cloud watch full access & SSM Full Access)
2. Create Instance and attached the role
3. Bootstrapping – Cloud watch agent

### 1) Create EC2 IAM Role (Cloud watch full access & SSM Full Access) - EC2-Cloudwatch-Role

The screenshot displays the AWS IAM console interface. The top navigation bar shows 'IAM' > 'Roles' > 'Create role'. The main content area is titled 'Select trusted entity' and shows the 'Trusted entity type' as 'AWS service'. The 'Use case' is 'EC2'. The 'Service or use case' is 'EC2'. The 'Use case' dropdown is set to 'EC2'. The 'Permissions policies' section shows two policies attached: 'AmazonSSMFullAccess' and 'CloudWatchFullAccess'. The 'Summary' section shows the role name 'EC2-Cloudwatch-Role', creation date 'September 12, 2024, 04:54 (UTC+05:30)', ARN 'arn:aws:iam::396913734302:role/EC2-Cloudwatch-Role', and instance profile ARN 'arn:aws:iam::396913734302:instance-profile/EC2-Cloudwatch-Role'.

### 2) Create Instance and attached the role

#### a) To push the IAM Role - EC2-Cloudwatch-Role

The screenshot displays the AWS EC2 console interface. The 'Advanced details' section shows the 'IAM instance profile' set to 'EC2-Cloudwatch-Role'. The 'Summary' section shows the 'Number of instances' as 1, 'Software Image (AMI)' as 'Amazon Linux 2 Kernel 5.10 AMI...', 'Virtual server type (instance type)' as 't2.micro', 'Firewall (security group)' as 'New security group', and 'Storage (volumes)' as '1 volume(s) - 8 GiB'. The 'Launch instance' button is visible at the bottom right.

b) Bootstraps creep command

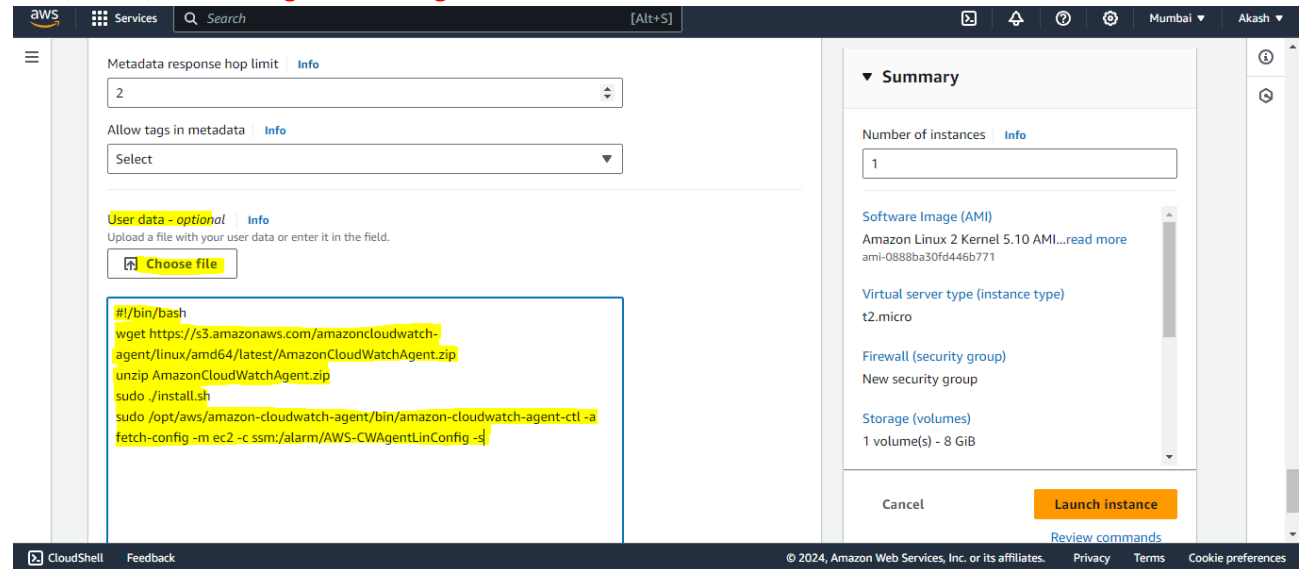
```
#!/bin/bash
```

```
wget https://s3.amazonaws.com/amazoncloudwatch-agent/linux/amd64/latest/AmazonCloudWatchAgent.zip
```

```
unzip AmazonCloudWatchAgent.zip
```

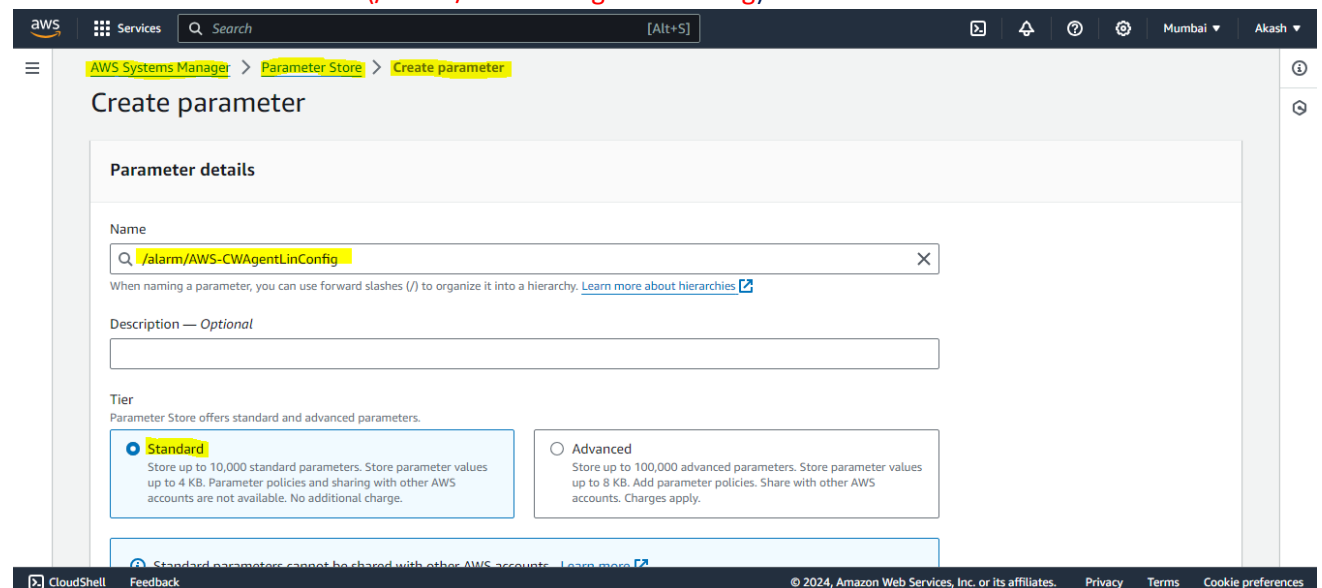
```
sudo ./install.sh
```

```
sudo /opt/aws/amazon-cloudwatch-agent/bin/amazon-cloudwatch-agent-ctl -a fetch-config -m ec2 -c ssm:/alarm/AWS-CWAgentLinConfig -s
```



c) Create **SSM (AWS Systems Manager)** Config Parameter

Value for the SSM Parameter (`/alarm/AWS-CWAgentLinConfig`):



d) Valu

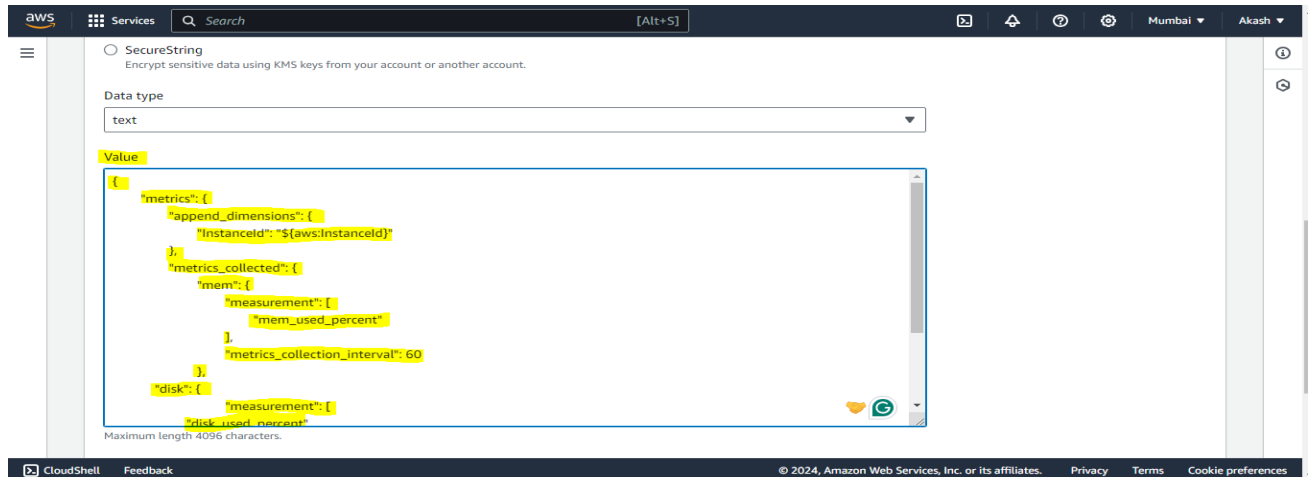
```
{
  "metrics": {
    "append_dimensions": {
      "InstanceId": "${aws:InstanceId}"
    },
    "metrics_collected": {
      "mem": {
        "measurement": [
          "mem_used_percent"
        ],

```

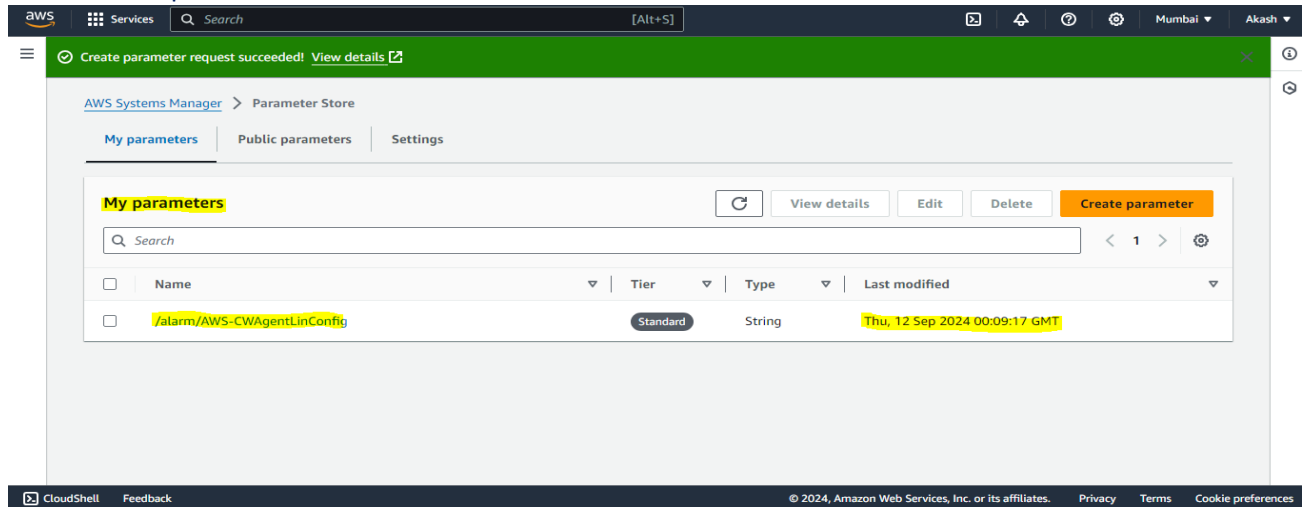
```

    "metrics_collection_interval": 60
  },
  "disk": {
    "measurement": [
      "disk_used_percent"
    ],
    "metrics_collection_interval": 60
  }
}

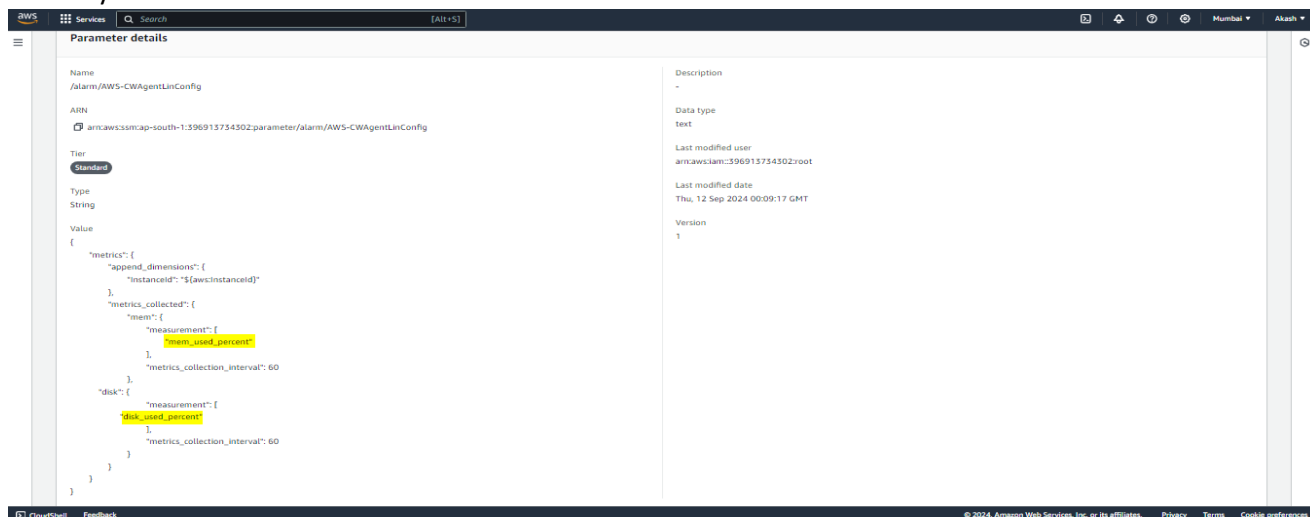
```



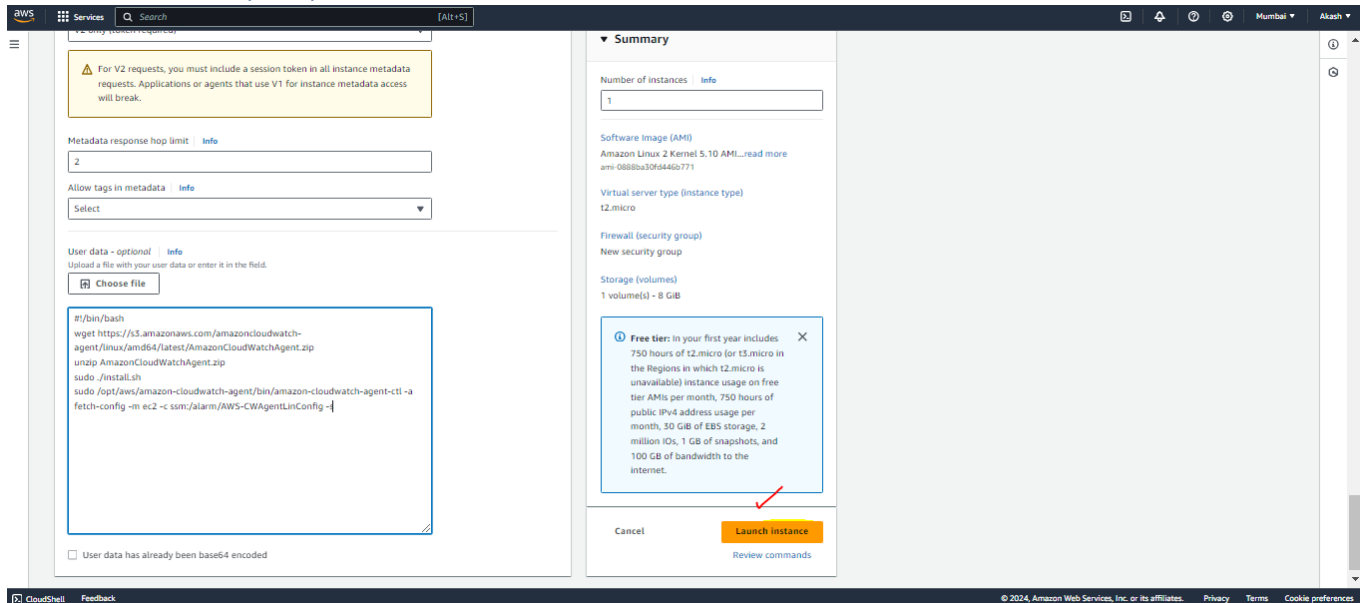
## e) New Created parameter Details



## Memory & Disk Monitor

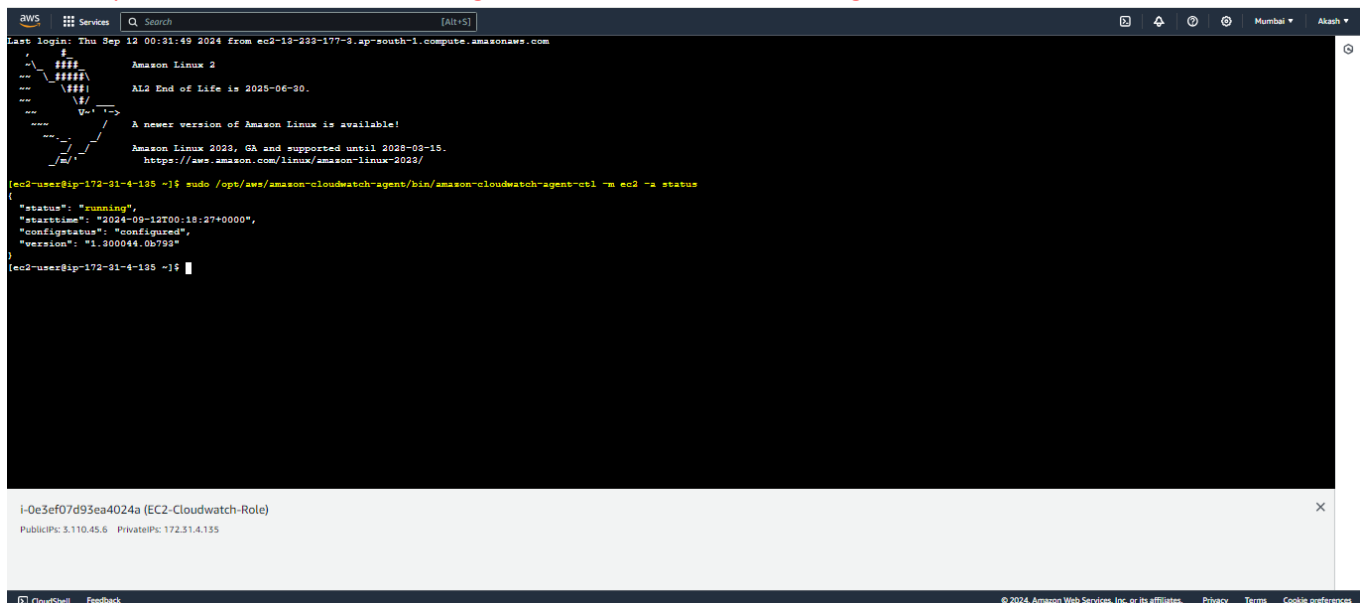


### 3) Launch the bootstrap script



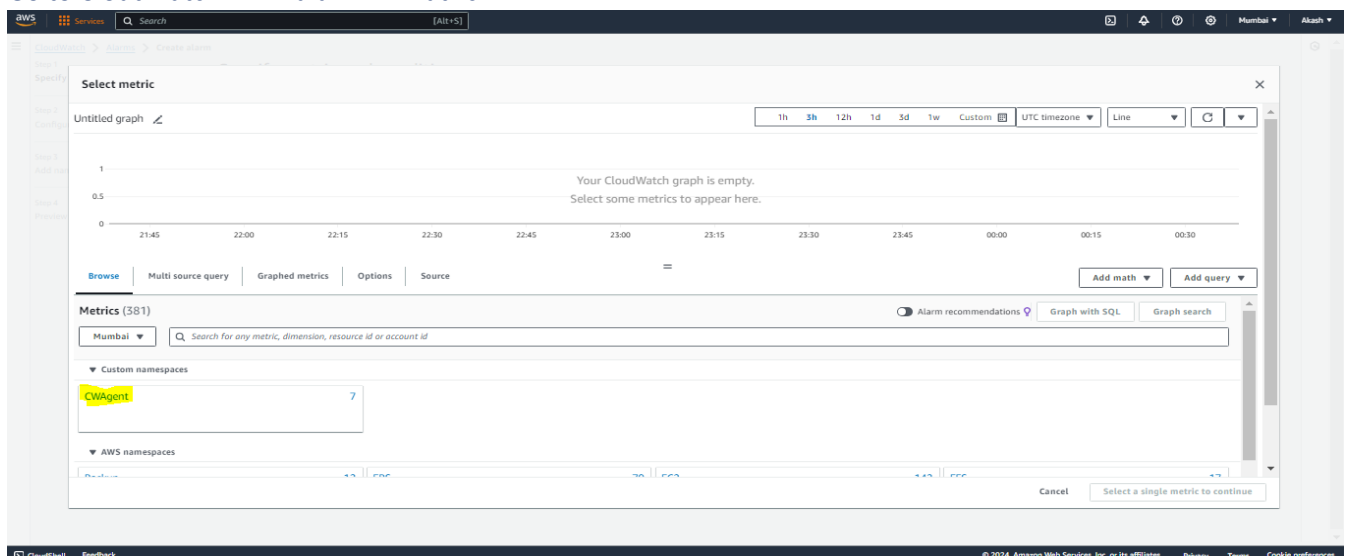
### 4) Check if EC2 Instance has CWAgent Installed or not: via Instance Connect

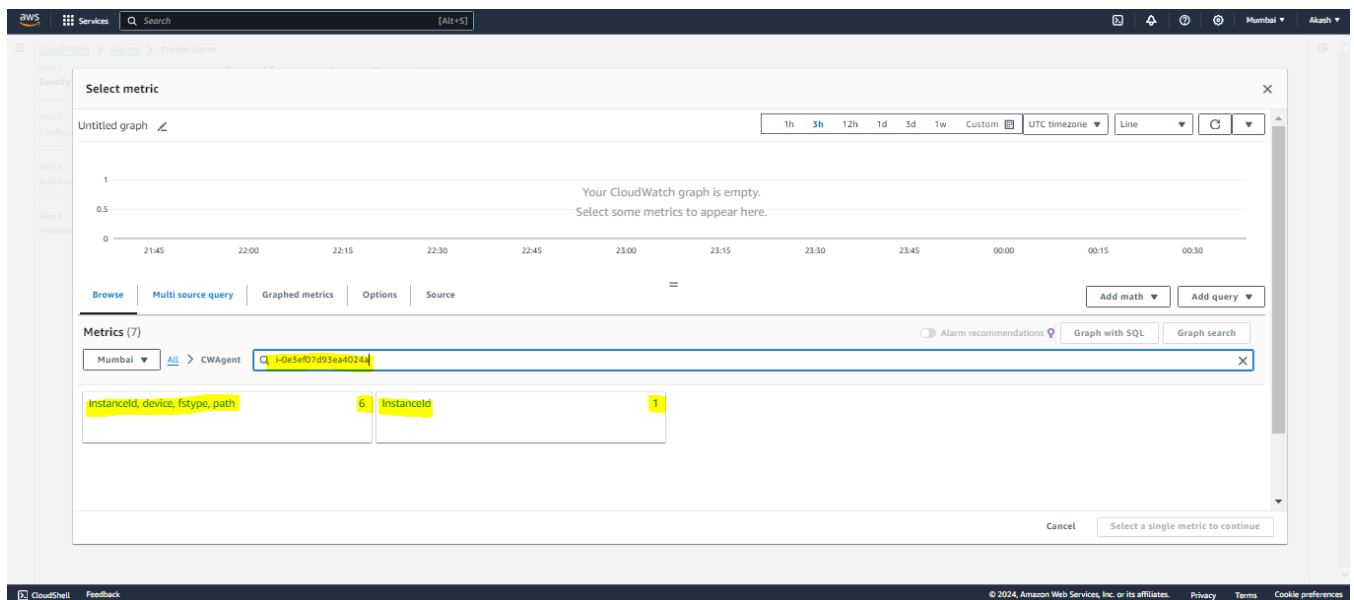
#**sudo /opt/aws/amazon-cloudwatch-agent/bin/amazon-cloudwatch-agent-ctl -m ec2 -a status**



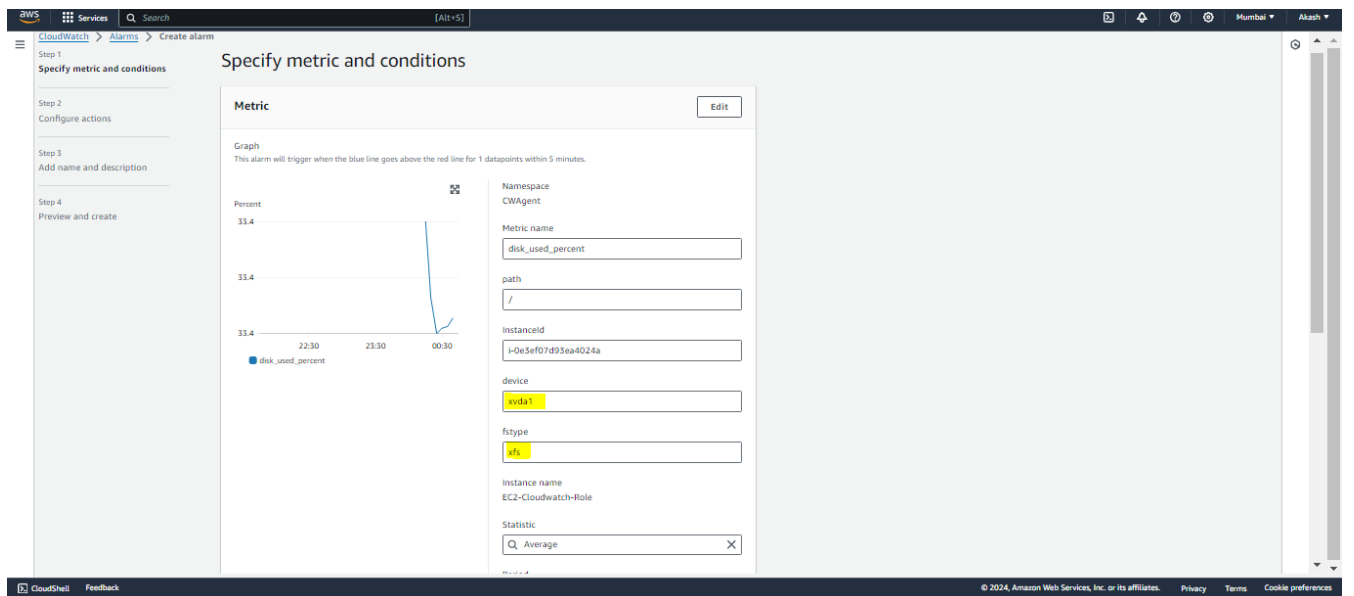
### 5) Take instance ID - i-0e3ef07d93ea4024a

Go to Cloudwatch -> All alarm -> matric

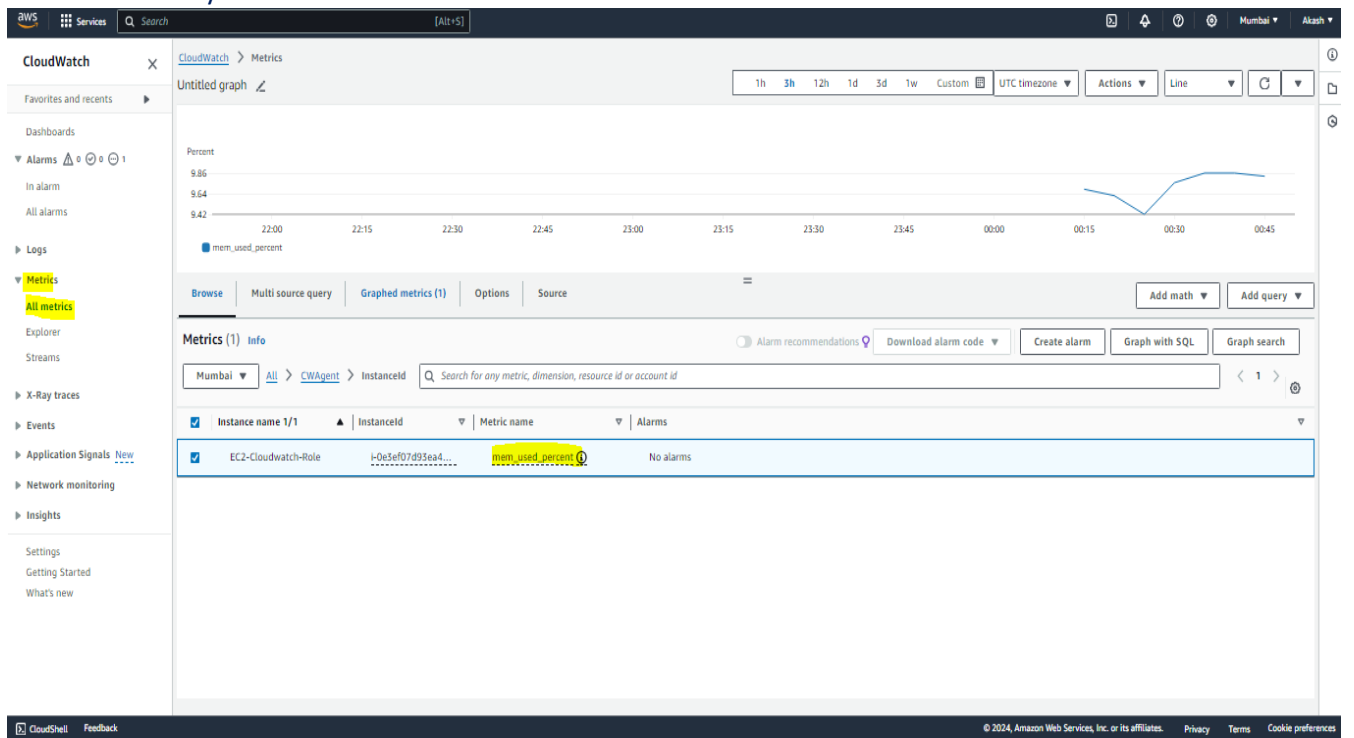




## Selected volume

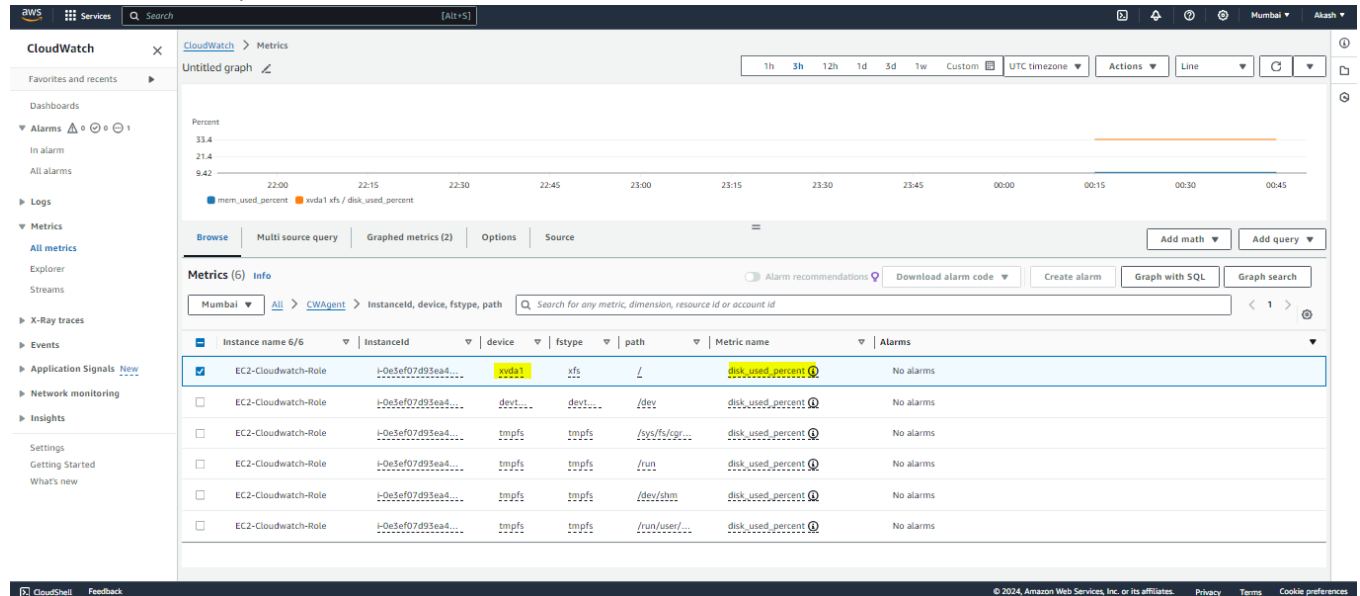


## Selected Memory

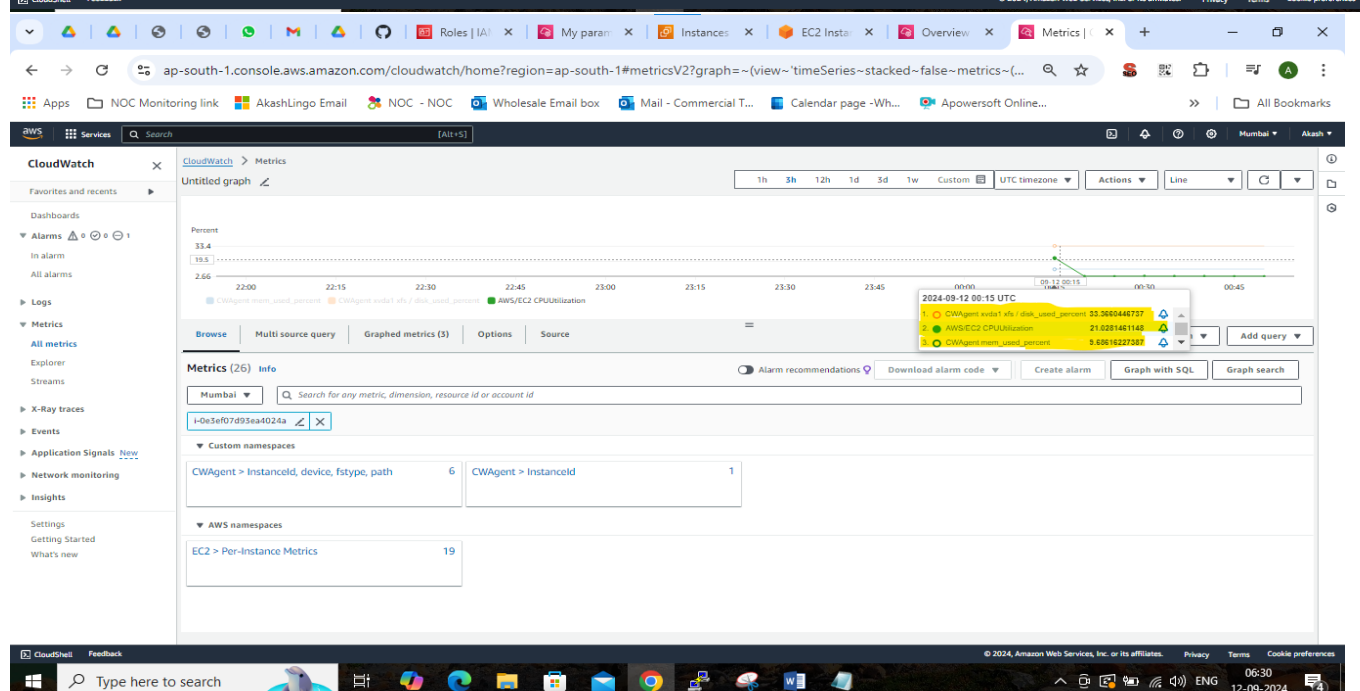
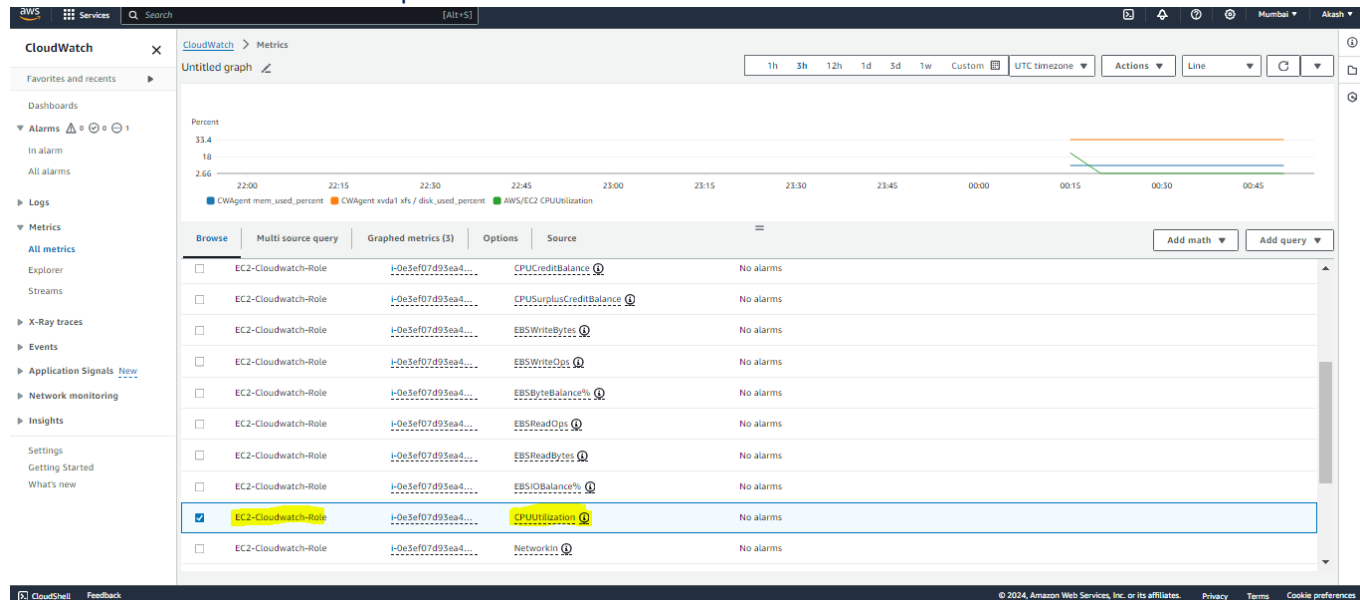




## Selected Disk used parameter



## Take CPU Utilization with the help of instance ID - i-0e3ef07d93ea4024a



## 6) Create Dashboard

