

“IMPLEMENTING AND MONITORING A WEB SERVER (Using AWS)”



By:

AARAV RAJ SINGH

VAIBHAVI MANWAL

VANDIT JAIN

HARSH GUPTA

Overview



Our project is about '**implementing and monitoring a webserver**'. In this project, first we will be implementing a webserver, i.e. **a virtual instance**. To actually have a compute environment to run our workloads, we are launching our instances on amazon web services (AWS) using Amazon **Elastic compute cloud (EC2)** services. We then will be connecting our instance to remote server. Once we get the remote access, we will be hosting our website on virtual instance. we then will be monitoring our web server using CloudWatch and **simple notification service (SNS)**. We will be triggering the alarm.

AWS SERVICES USED:

1. **EC2 (ELASTIC CLOUD COMPUTING):** EC2 is a Web Service that enables you to Launch and manage LINUX/UNIX and WINDOWS server instances in Amazon's Data center.



(AMAZON EC2)

2. **SNS (SIMPLE NOTIFICATION SERVICE):** Amazon Simple Notification Service (SNS) is a highly available, durable, secure, fully managed pub/sub messaging service that enables you to decouple microservices, distributed systems, and serverless applications. Amazon SNS provides topics for high-throughput, push-based, many-to-many messaging.



(AMAZON SNS)

3. **CLOUDWATCH:** Amazon CloudWatch is a monitoring and observability service built for DevOps engineers, developers, site reliability engineers (SREs), and IT managers.

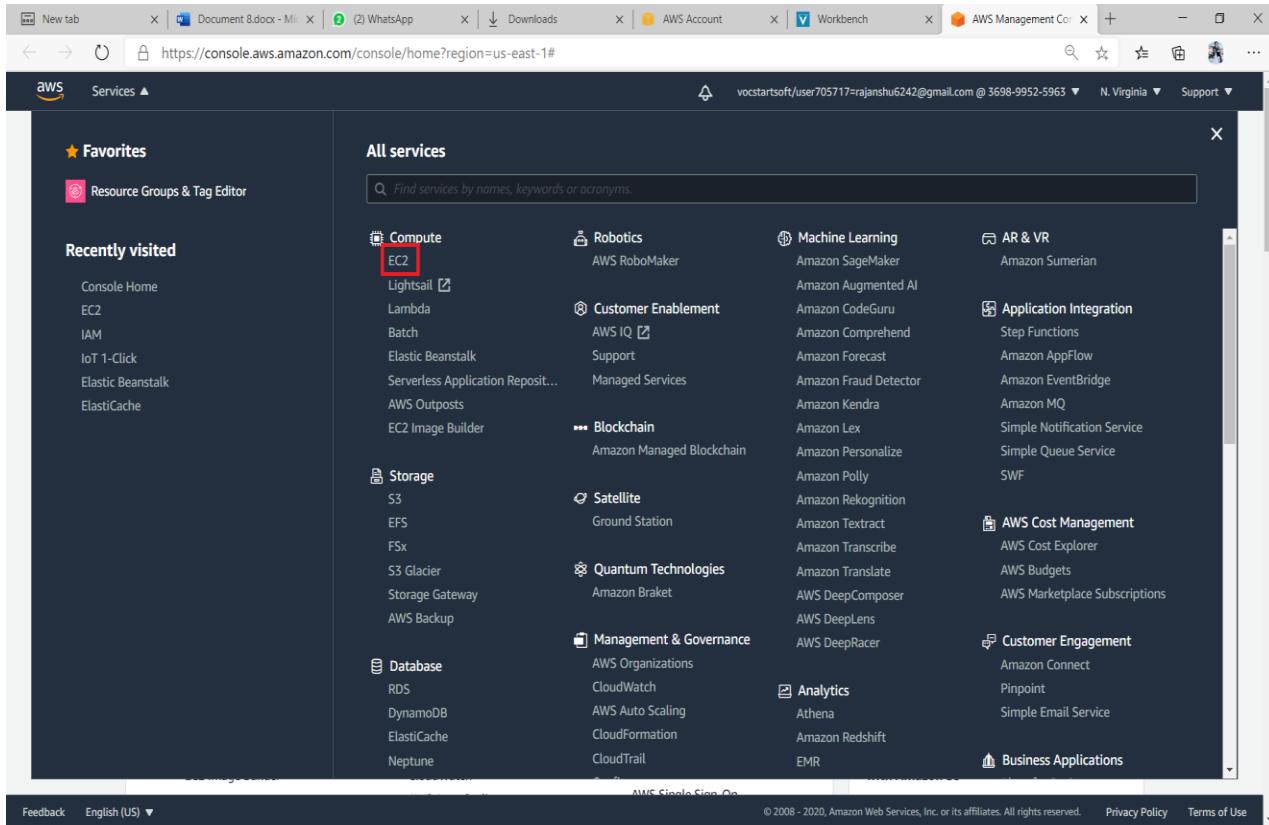


(CLOUDWATCH)

“Creating an EC2 instance and Implementing web server on it “

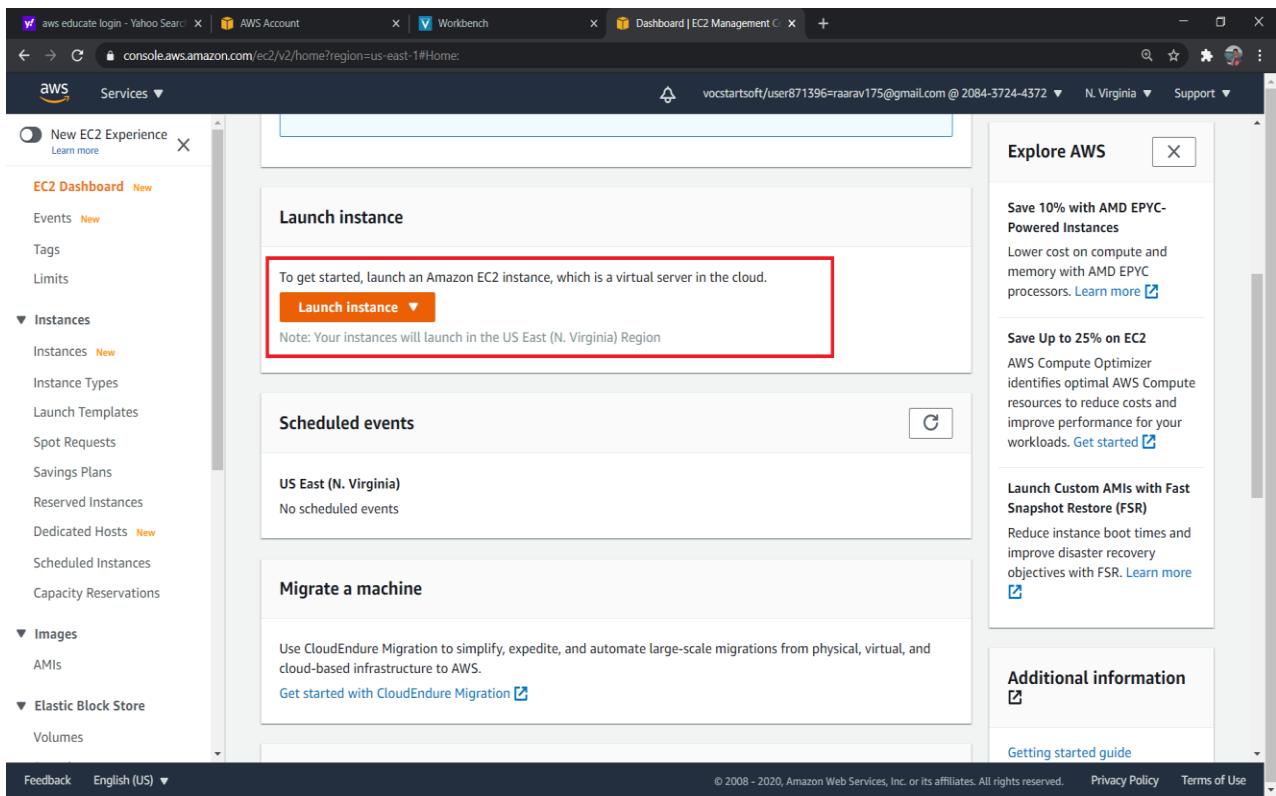


STEP 1: Login to AWS account and click on ‘EC2’ under All Services section.



The screenshot shows the AWS Management Console interface. In the top navigation bar, there are several tabs: New tab, Document 8.docx - Microsoft Word, (2) WhatsApp, Downloads, AWS Account, Workbench, and AWS Management Con. Below the tabs, the URL https://console.aws.amazon.com/console/home?region=us-east-1# is displayed. The main content area has a dark background. On the left, there's a sidebar with sections for Favorites (Resource Groups & Tag Editor), Recently visited (Console Home, EC2, IAM, IoT 1-Click, Elastic Beanstalk, ElastiCache), Compute (with EC2 highlighted by a red box), Storage (S3, EFS, FSx, S3 Glacier, Storage Gateway, AWS Backup), Database (RDS, DynamoDB, ElastiCache, Neptune), and a collapsed section for Lambda, Batch, and Elastic Beanstalk. The right side shows the 'All services' grid, which includes categories like Robotics, Machine Learning, AR & VR, Application Integration, etc., each with a list of specific AWS services. A search bar at the top of the grid allows users to find services by name or keyword.

STEP 2: Click on ‘Launch Instance’ and from the dropdown launch it.



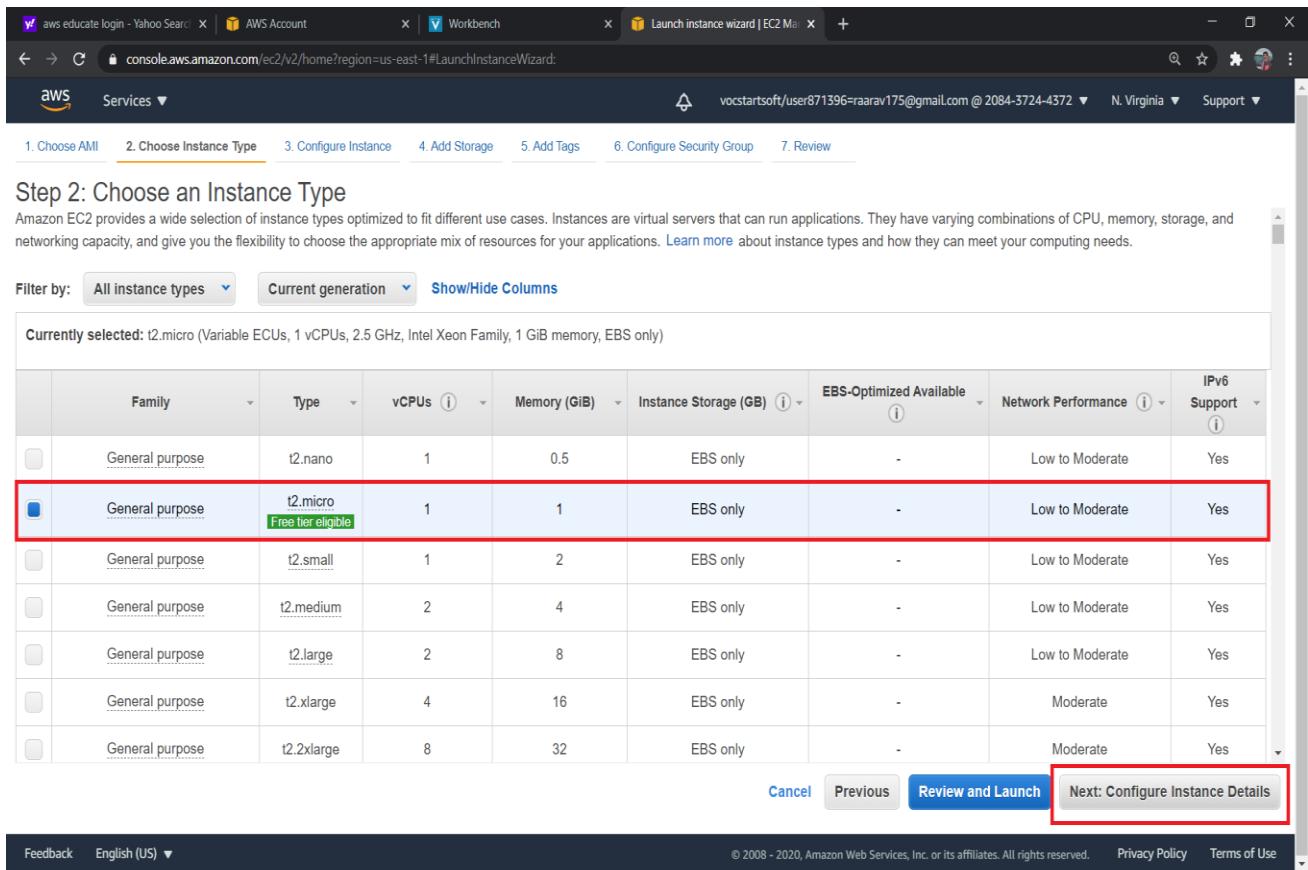
STEP 3: Select an Amazon Machine Image (AMI) of 'Microsoft Windows Server 2019 Base'.

The screenshot shows the AWS Launch Instance Wizard Step 1: Choose an Amazon Machine Image (AMI). The page title is "Step 1: Choose an Amazon Machine Image (AMI)". There are seven tabs at the top: 1. Choose AMI (selected), 2. Choose Instance Type, 3. Configure Instance, 4. Add Storage, 5. Add Tags, 6. Configure Security Group, and 7. Review. A "Launch a database using RDS" button is visible. The main content area lists several AMIs:

- Ubuntu Server 18.04 LTS (HVM), SSD Volume Type** - ami-0817d428a6fb68645 (64-bit x86) / ami-0f2b11fdc1647918 (64-bit Arm)
Free tier eligible
Ubuntu Server 18.04 LTS (HVM), EBS General Purpose (SSD) Volume Type. Support available from Canonical (<http://www.ubuntu.com/cloud/services>).
Root device type: ebs Virtualization type: hvm ENA Enabled: Yes
Select button (radio button selected for 64-bit (x86))
64-bit (x86)
64-bit (Arm)
- Microsoft Windows Server 2019 Base** - ami-0eb7fbcc77e5e6ec6
Windows
Free tier eligible
Microsoft Windows 2019 Datacenter edition. [English]
Root device type: ebs Virtualization type: hvm ENA Enabled: Yes
Select button (radio button selected for 64-bit (x86))
64-bit (x86)
- Deep Learning AMI (Ubuntu 18.04) Version 34.0** - ami-06a25ee8966373068
Free tier eligible
MXNet-1.6.0, TensorFlow-2.3.0, 2.1.0 & 1.15.3, PyTorch-1.4.0 & 1.6.0, Neuron, & others. NVIDIA CUDA, cuDNN, NCCL, Intel MKL-DNN, Docker, NVIDIA-Docker & EFA support. For fully managed experience, check: <https://aws.amazon.com/sagemaker>
Root device type: ebs Virtualization type: hvm ENA Enabled: Yes
Select button (radio button selected for 64-bit (x86))
64-bit (x86)
- Deep Learning AMI (Ubuntu 16.04) Version 34.0** - ami-0b2671fd508a3a2c3
Free tier eligible
MXNet-1.6.0, TensorFlow-2.3.0, 2.1.0 & 1.15.3, PyTorch-1.4.0 & 1.6.0, EI, Neuron, & others. NVIDIA CUDA, cuDNN, NCCL, Intel MKL-DNN, Docker, NVIDIA-Docker & EFA support. For fully managed experience, check: <https://aws.amazon.com/sagemaker>
Root device type: ebs Virtualization type: hvm ENA Enabled: Yes
Select button (radio button selected for 64-bit (x86))
64-bit (x86)

At the bottom, there are links for Feedback, English (US) ▾, © 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved., Privacy Policy, Terms of Use, and a dropdown menu.

STEP 4: Choose a ‘t2. micro’ instance which have 1 vCPU and have IPv6 Support and Click on “Next: Configure Instance Details”



The screenshot shows the AWS Launch Instance Wizard at Step 2: Choose an Instance Type. The user has selected the t2.micro instance type, which is highlighted with a red box. The t2.micro instance is described as having 1 vCPU, 1 GiB memory, and EBS only storage. It is marked as 'Free tier eligible'. The 'Next: Configure Instance Details' button is also highlighted with a red box.

	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance	IPv6 Support
[unchecked]	General purpose	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
<input checked="" type="checkbox"/>	General purpose	t2.micro <small>Free tier eligible</small>	1	1	EBS only	-	Low to Moderate	Yes
[unchecked]	General purpose	t2.small	1	2	EBS only	-	Low to Moderate	Yes
[unchecked]	General purpose	t2.medium	2	4	EBS only	-	Low to Moderate	Yes
[unchecked]	General purpose	t2.large	2	8	EBS only	-	Low to Moderate	Yes
[unchecked]	General purpose	t2.xlarge	4	16	EBS only	-	Moderate	Yes
[unchecked]	General purpose	t2.2xlarge	8	32	EBS only	-	Moderate	Yes

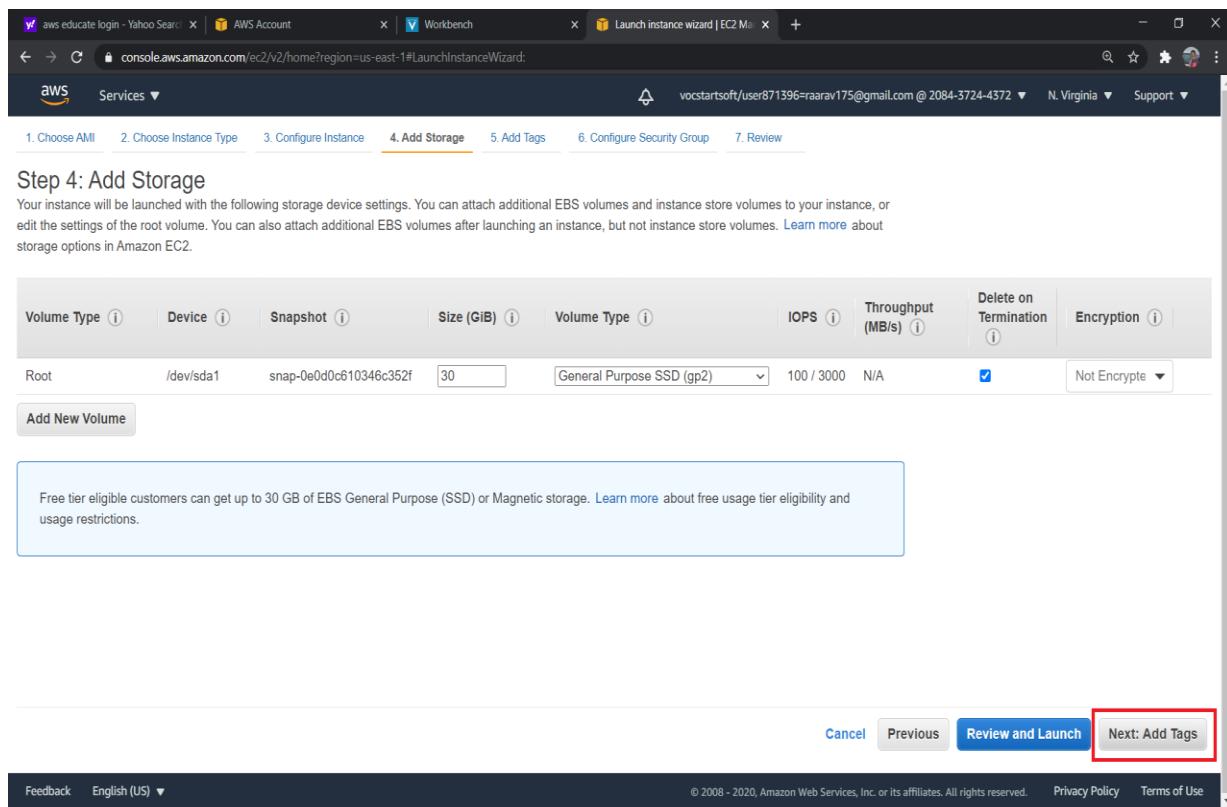
STEP 5: Keep Everything as it is and Click on “Next: Add Storage”.

The screenshot shows the AWS Launch Instance Wizard at Step 3: Configure Instance Details. The page is titled "Step 3: Configure Instance Details" and includes a sub-instruction: "Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot instances to take advantage of the lower pricing, assign an access management role to the instance, and more." Below this, there are several configuration sections:

- Number of instances:** Set to 1. There is a link "Launch into Auto Scaling Group".
- Purchasing option:** A checkbox for "Request Spot instances" is present.
- Network:** Set to "vpc-eb53a596 (default)". There are links to "Create new VPC" and "Create new subnet".
- Subnet:** Set to "No preference (default subnet in any Availability Zone)". There is a link to "Create new subnet".
- Auto-assign Public IP:** Set to "Use subnet setting (Enable)".
- Placement group:** A checkbox for "Add instance to placement group" is present.
- Capacity Reservation:** Set to "Open".
- Domain join directory:** Set to "No directory". There is a link to "Create new directory".
- IAM role:** Set to "None". There is a link to "Create new IAM role".
- Shutdown behavior:** Set to "Stop".
- Stop - Hibernate behavior:** A checkbox for "Enable hibernation as an additional stop behavior" is present.

At the bottom right, there are four buttons: "Cancel", "Previous", "Review and Launch" (in blue), and "Next: Add Storage" (which is highlighted with a red box).

STEP 6: Don't make any changes and Click on “Next: Add Tags”.



STEP 7: Don't make any Changes and click on “Next: Configure Security Group”

The screenshot shows the AWS Launch Instance Wizard at Step 5: Add Tags. The browser window has tabs for 'aws educate login - Yahoo Search', 'AWS Account', 'Workbench', and 'Launch instance wizard | EC2 Mail'. The main content area shows the 'Step 5: Add Tags' section. It includes fields for 'Key' (128 characters maximum) and 'Value' (256 characters maximum). A note says 'This resource currently has no tags'. Below it, instructions say 'Choose the Add tag button or click to add a Name tag.' and 'Make sure your IAM policy includes permissions to create tags.' A 'Add Tag' button is available. At the bottom, there are buttons for 'Cancel', 'Previous', 'Review and Launch', and 'Next: Configure Security Group' (which is highlighted with a red box).

STEP 8: Click on Add Rule and Choose HTTP. Now Click on “Review and Launch”.

Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. [Learn more](#) about Amazon EC2 security groups.

Assign a security group: Create a new security group Select an existing security group

Security group name:

Description:

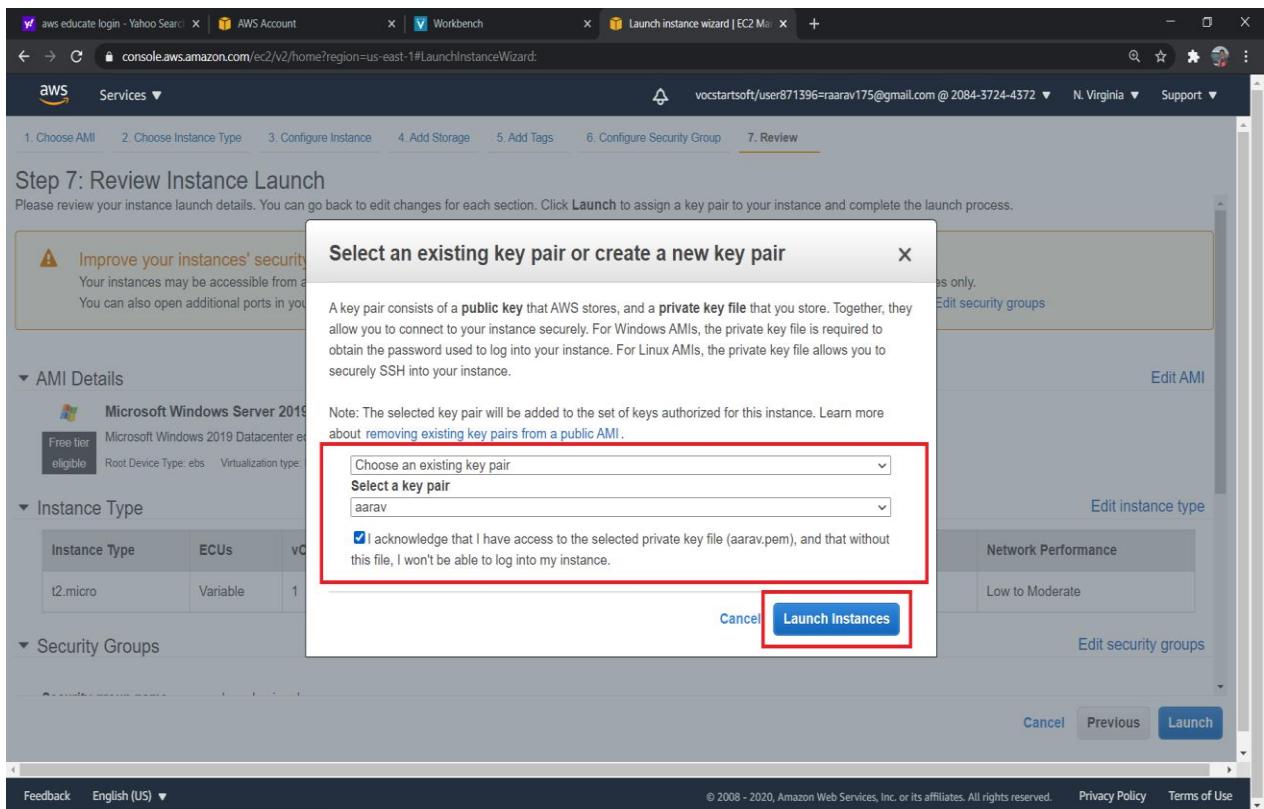
Type	Protocol	Port Range	Source	Description
RDP	TCP	3389	Custom 0.0.0.0/0	e.g. SSH for Admin Desktop
HTTP	TCP	80	Custom 0.0.0.0/0, ::/0	e.g. SSH for Admin Desktop

Add Rule

Warning
Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

Cancel Previous **Review and Launch**

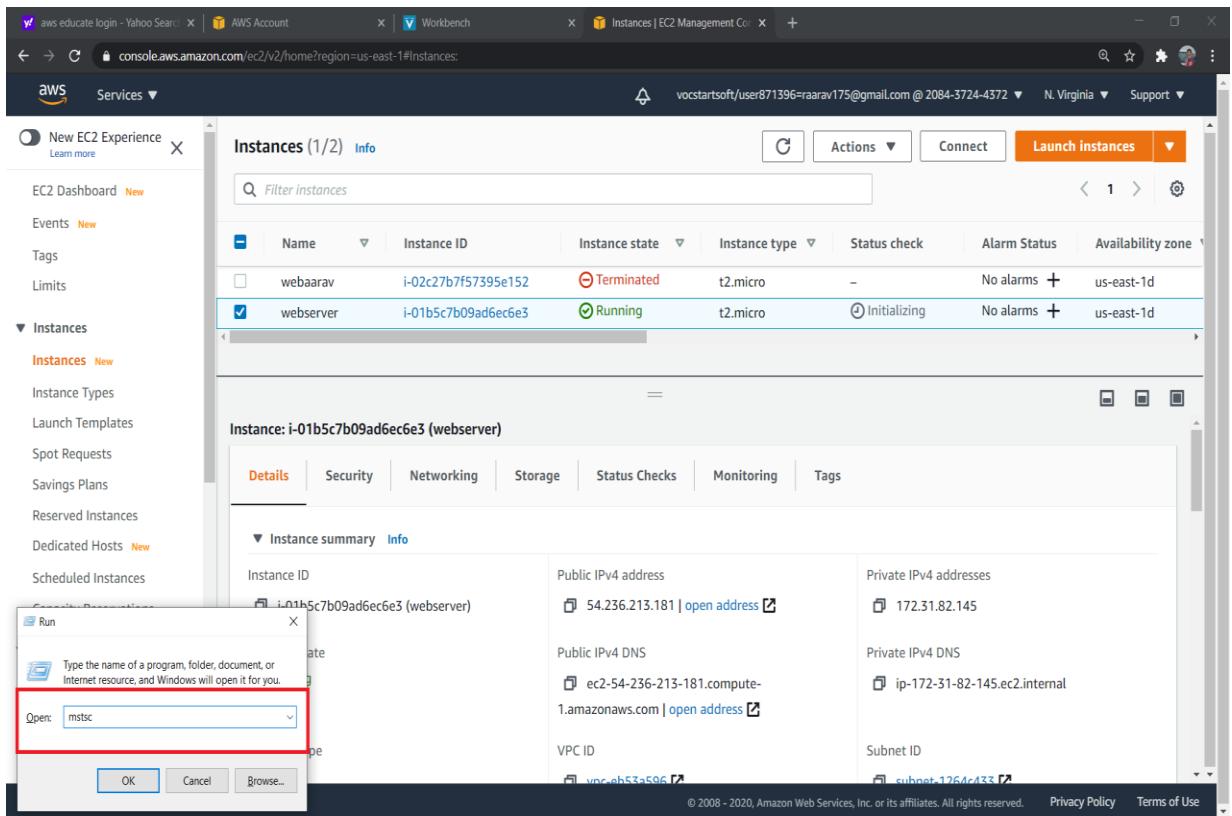
STEP 9: Select “Choose an existing key pair” or “Create a new pair”, check that box and click on “Launch Instance”.



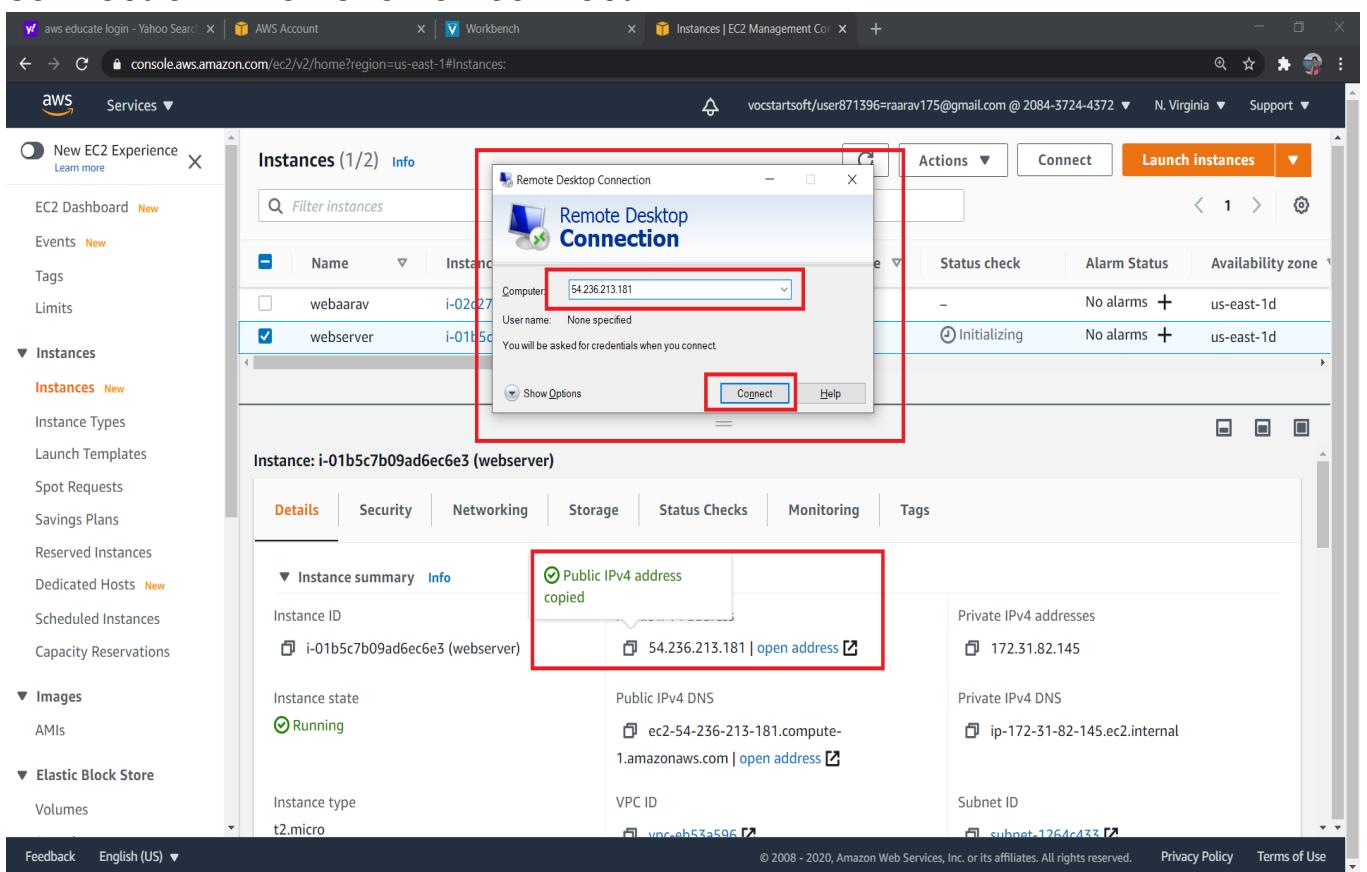
STEP 10: check your instance box and name it accordingly. Copy the Public IPv4 address.

The screenshot shows the AWS EC2 Management Console interface. On the left, there's a navigation sidebar with options like EC2 Dashboard, Events, Tags, Limits, Instances (selected), and various launch and reservation options. The main area displays a table of instances. One instance, named 'webserver' with the ID i-01b5c7b09ad6ec6e3, is selected and its details are expanded. The 'Details' tab is active, showing the instance summary. The 'Public IPv4 address' field is highlighted with a red box and contains the value '54.236.213.181'. Other visible fields include Instance ID (i-01b5c7b09ad6ec6e3), Instance state (Running), Instance type (t2.micro), Public IPv4 DNS (ec2-54-236-213-181.compute-1.amazonaws.com), VPC ID (vpc-0b53a506), and Subnet ID (subnet-0264c133).

STEP 11: press ‘win+R’ key (open Run terminal) and type ‘mstsc’ and hit OK.



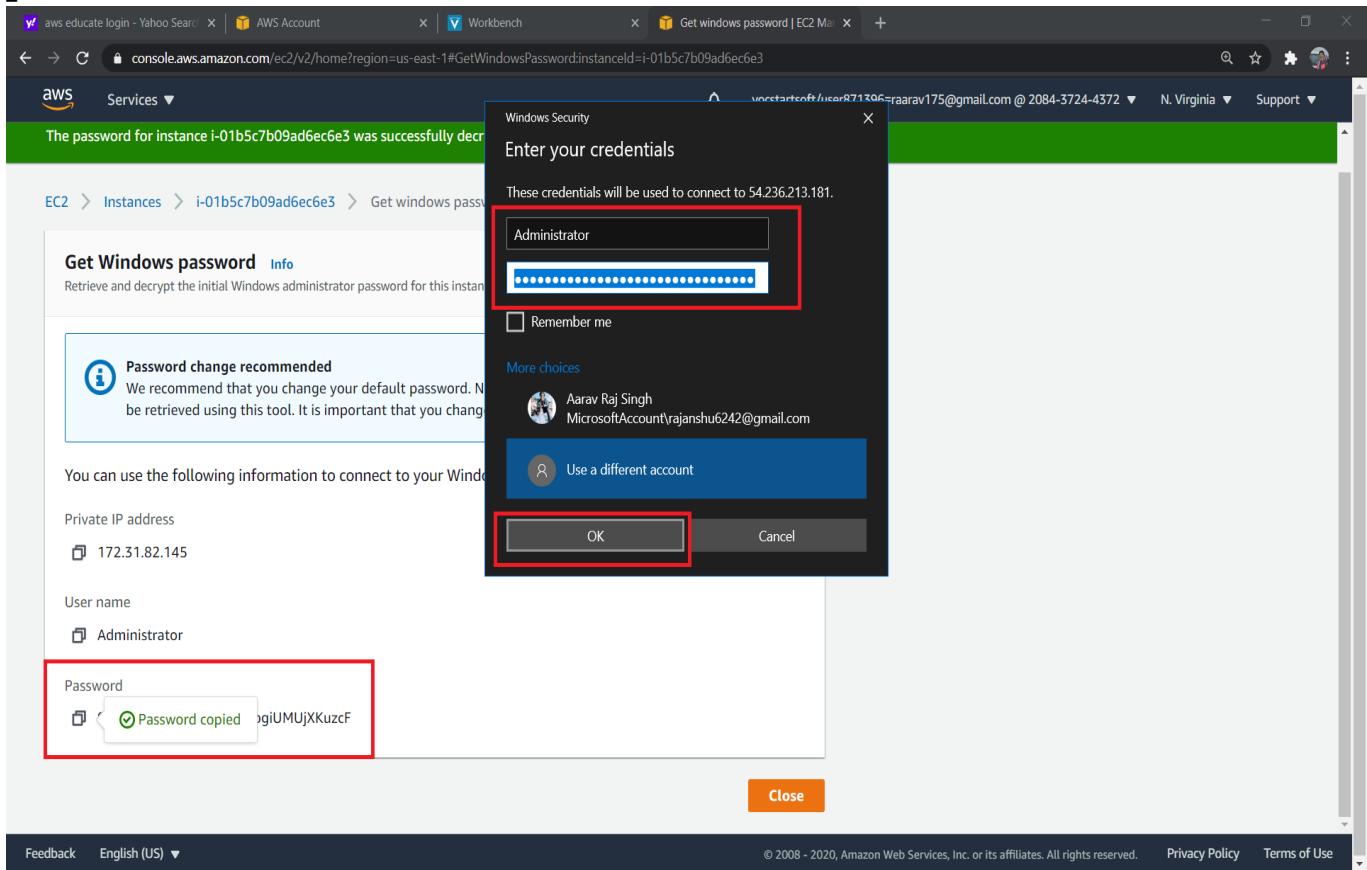
STEP 12: Copy ipv4 of the instance and then paste it in remote connection. Then click on connect.



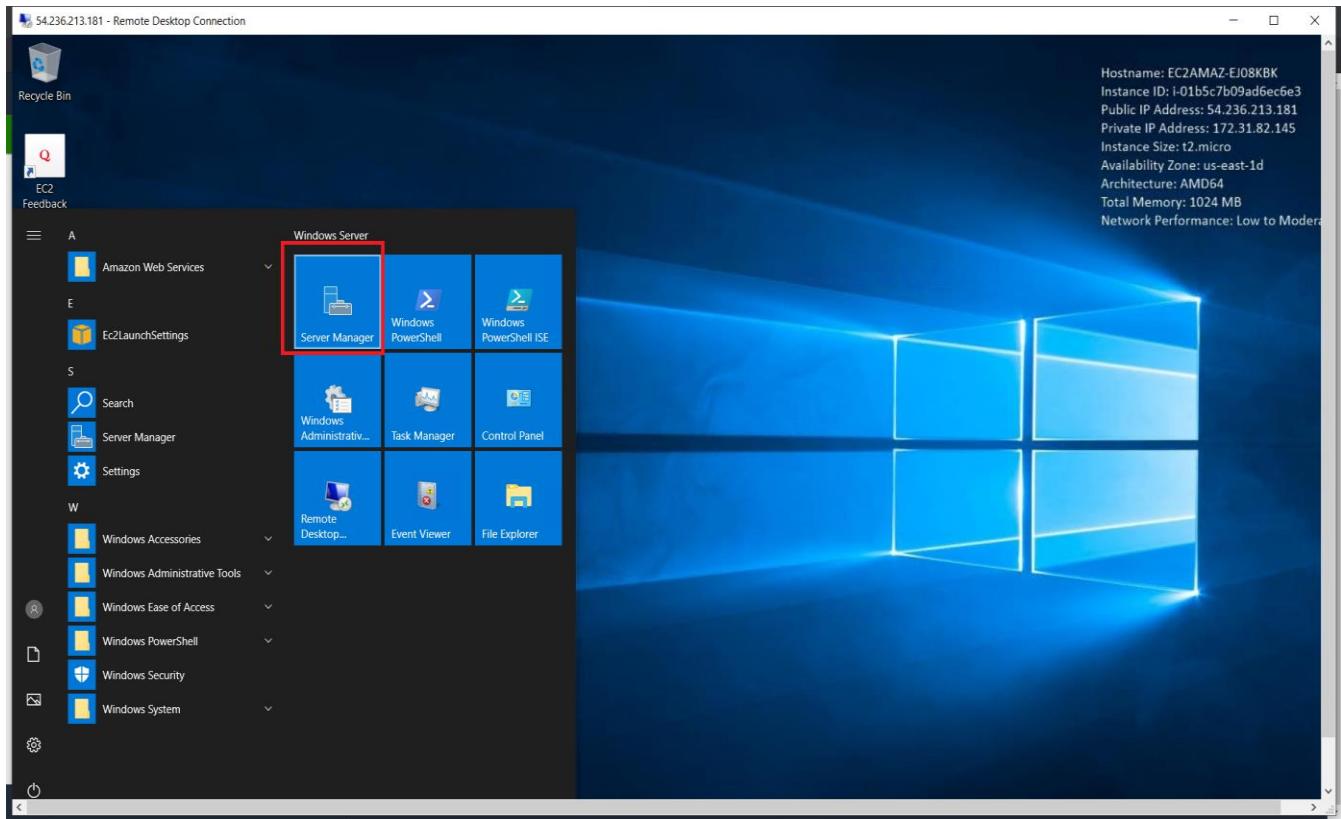
STEP 13: Go to your instance window and then click on actions. Get password from there.

The screenshot shows the AWS EC2 Instances Management Console. On the left, the navigation pane is visible with sections like New EC2 Experience, EC2 Dashboard, Events, Tags, Limits, Instances (selected), Images, and Elastic Block Store. The main area displays a table of instances with two entries: 'webaarav' (terminated) and 'webserver' (running). The 'Actions' button in the top right is highlighted with a red box. A dropdown menu is open next to it, also highlighted with a red box, showing options like View details, Get Windows password, Create template from instance, Launch more like this, Manage tags, Instance state, Instance settings, Networking, Image, and Monitoring. The 'Get Windows password' option is specifically highlighted with a red box. Below the table, the details for the 'webserver' instance are shown, including its ID, state, type, and network information.

STEP 14: Go to your remote connection and enter your username as “Administrator” and copy the password from get password then paste it.



STEP 15: After you got the remote access, click on ‘server manager’ in start bar on Your Virtual Machine.

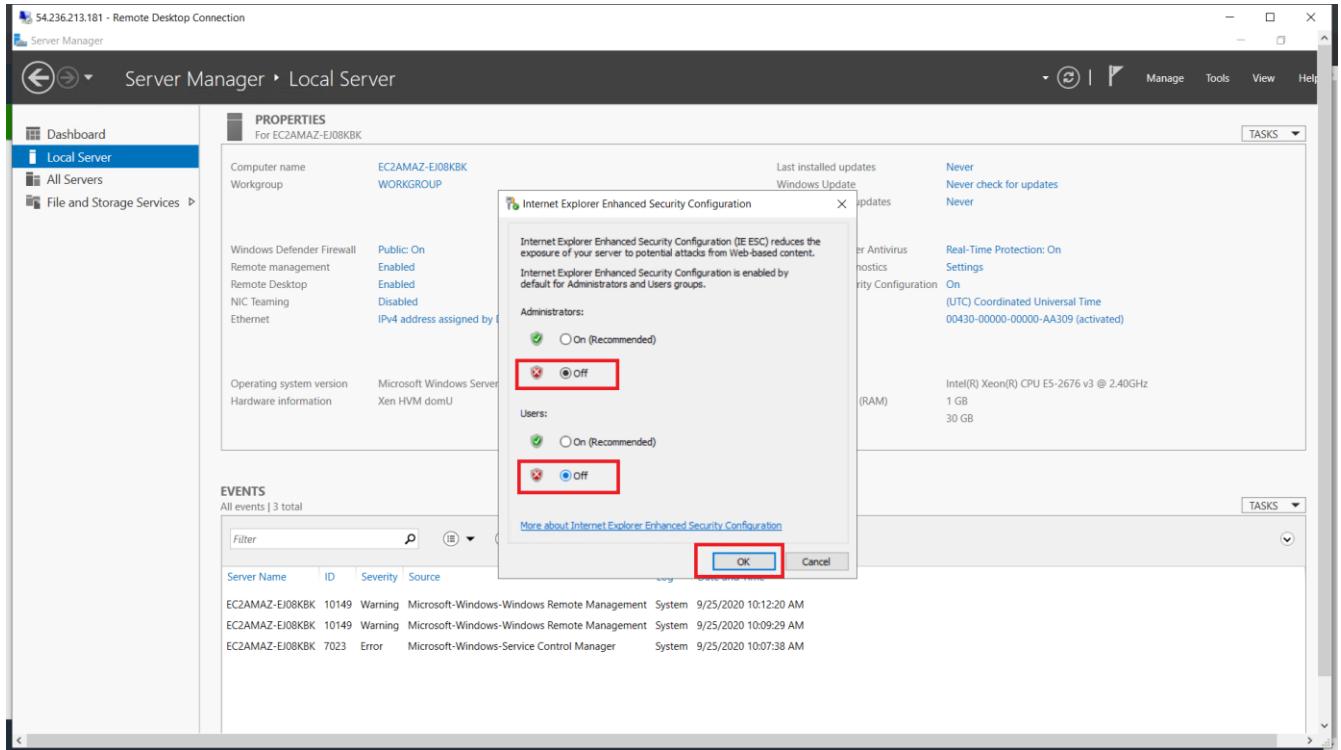


STEP 16: click on ‘Local Server’ then open IE enhanced security configuration.

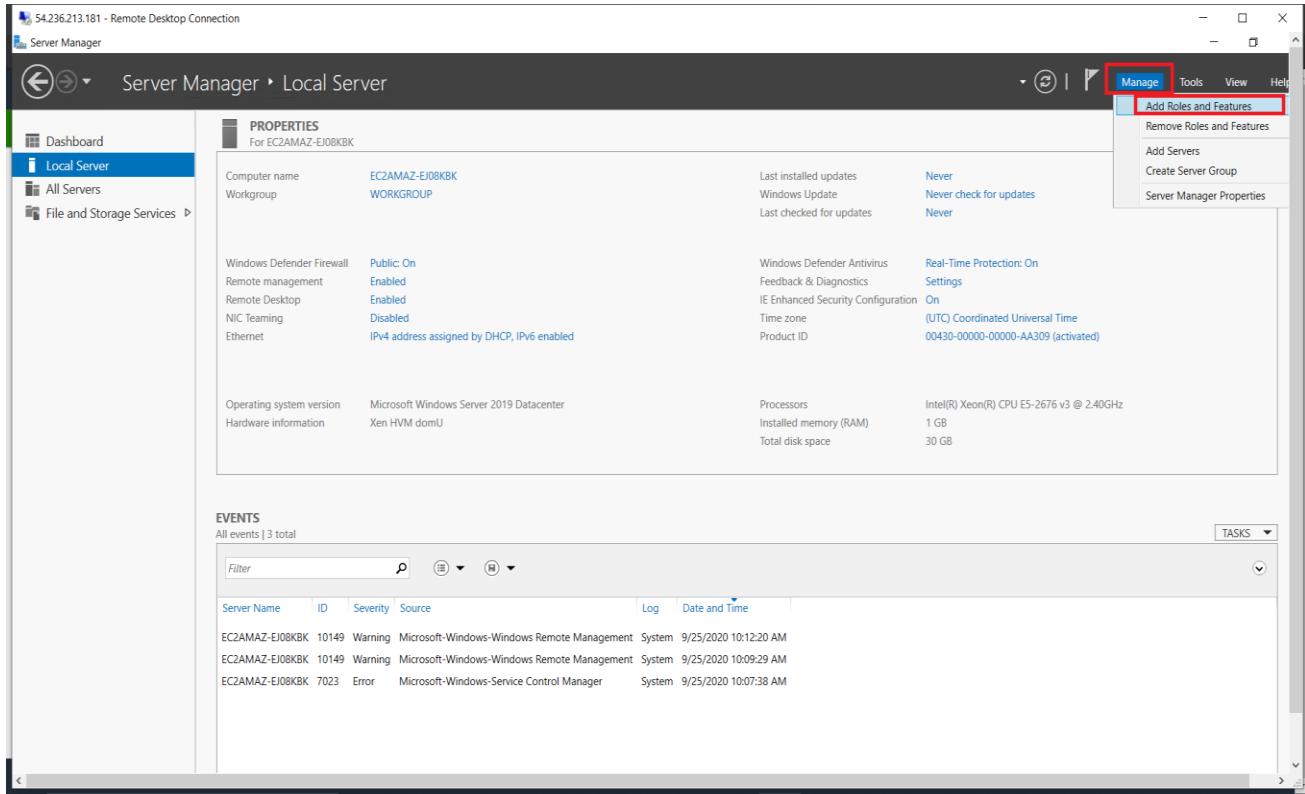
The screenshot shows the Windows Server Manager interface. On the left, a navigation pane has 'Local Server' selected, highlighted with a red box. The main area displays the 'PROPERTIES' for the local server, specifically for EC2AMAZ-EJ08KBK. In the 'Windows Defender Firewall' section, the 'IE Enhanced Security Configuration' row is highlighted with a red box. Below this, the 'EVENTS' section shows a table of system logs with three entries:

Server Name	ID	Severity	Source	Log	Date and Time
EC2AMAZ-EJ08KBK	10149	Warning	Microsoft-Windows-Windows Remote Management	System	9/25/2020 10:12:20 AM
EC2AMAZ-EJ08KBK	10149	Warning	Microsoft-Windows-Windows Remote Management	System	9/25/2020 10:09:29 AM
EC2AMAZ-EJ08KBK	7023	Error	Microsoft-Windows-Service Control Manager	System	9/25/2020 10:07:38 AM

STEP 17: Turn 'Off' both the settings and click 'OK'

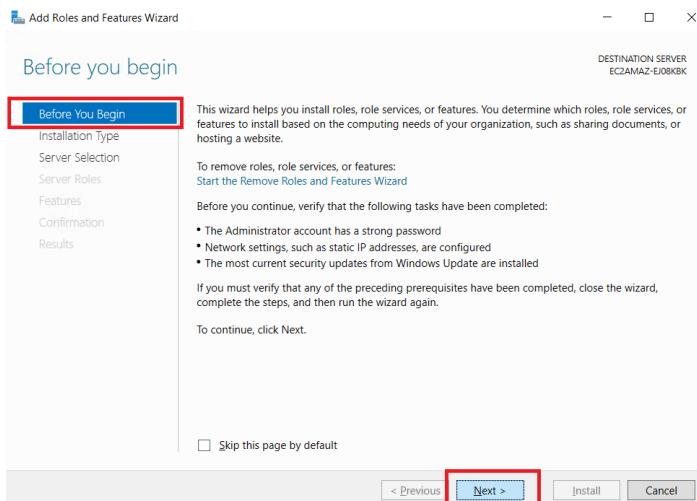


STEP 18: Click on manage and open add roles and features on Top-Right.

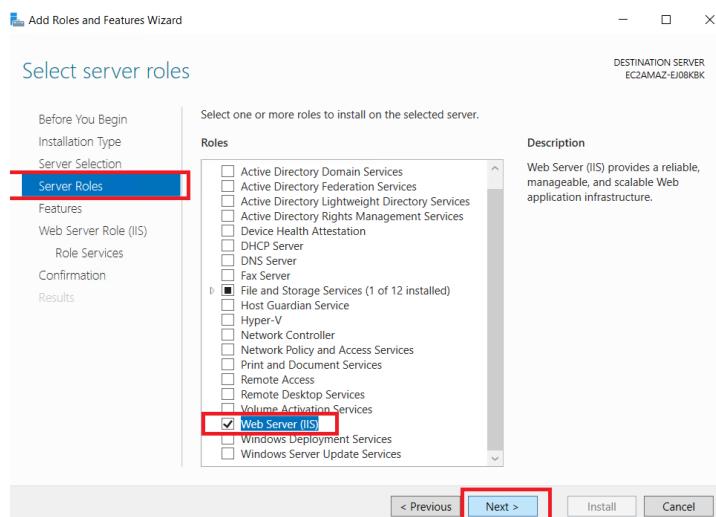


STEP 19: Install Web Server (IIS).

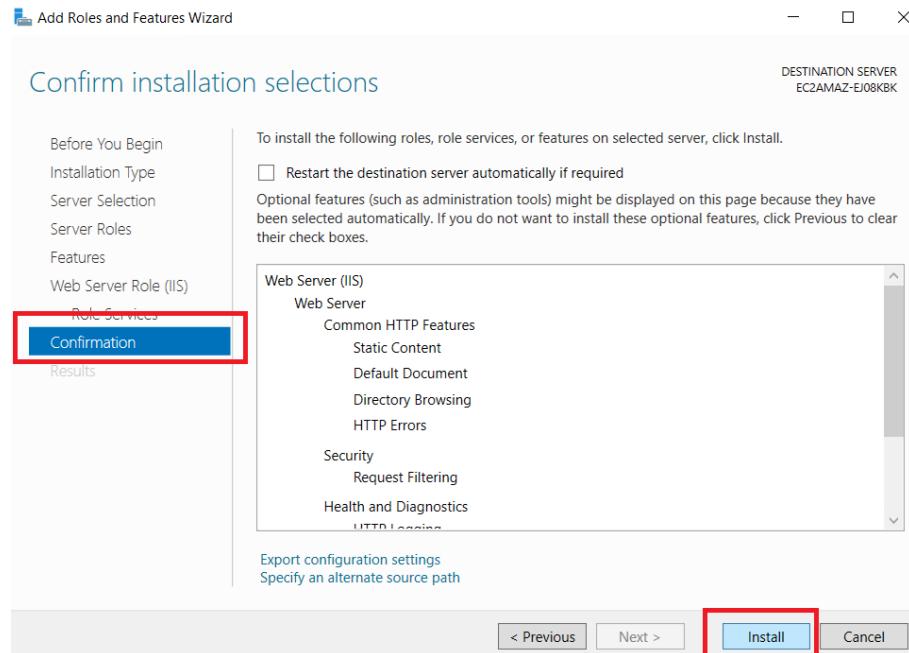
Sub-step I: click on next >>next>>next since no need to make any changes.



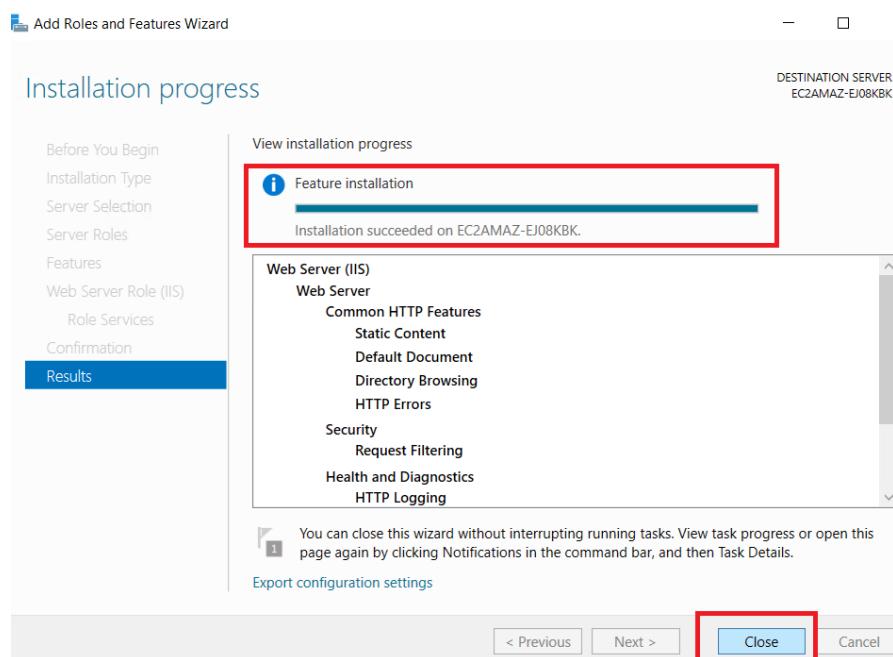
Sub-step II: click on web server (IIS) and add features, then click on next>>next.



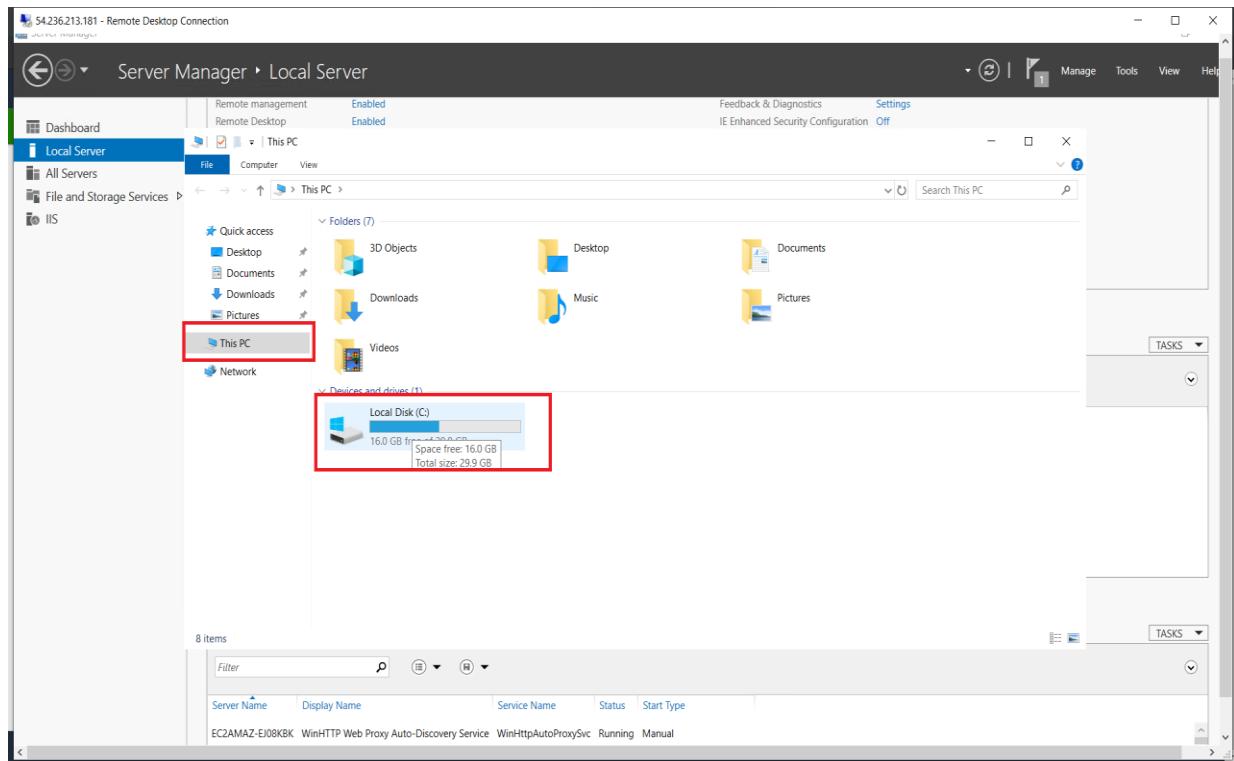
Sub-step III: On the confirmation step Click on install.



Sub-step IV: wait for installation completion and then click on close.

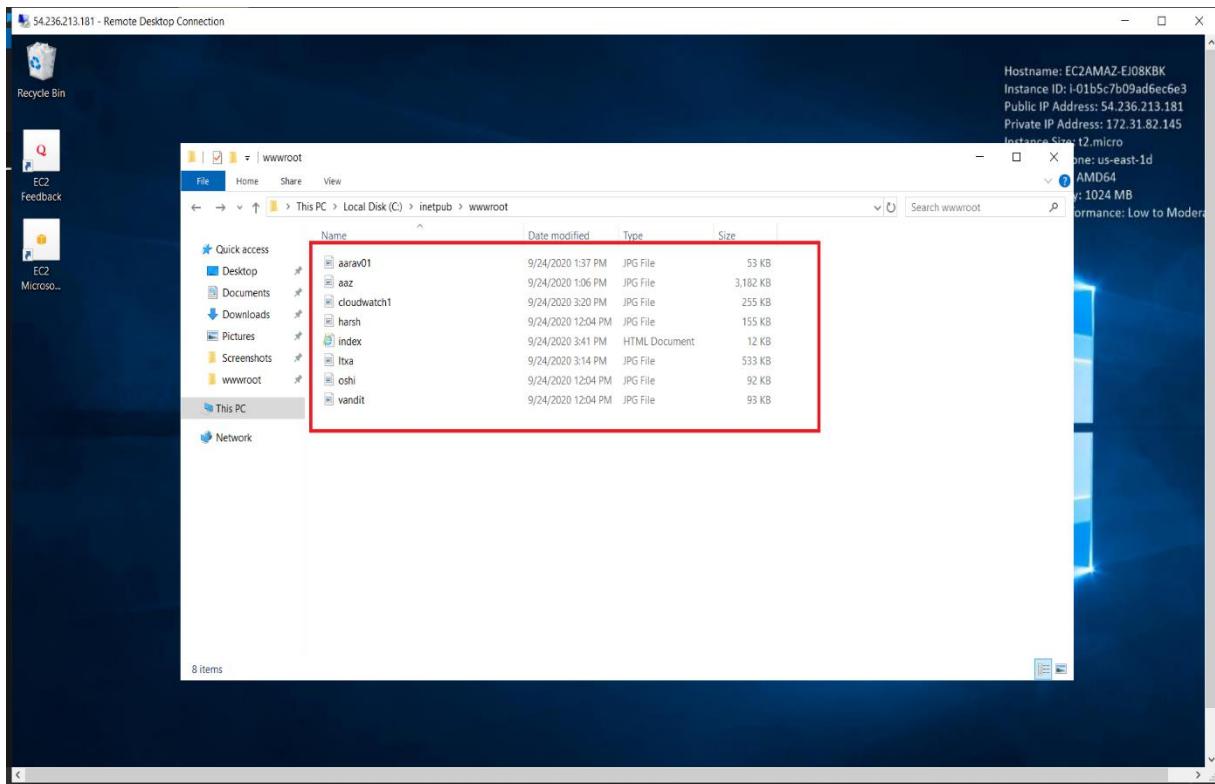


STEP 20: Click on ‘win + E’ key to open ‘This PC’. Click on C drive.



Step 21: click on inetpub folder then on wwwroot folder and paste all the content you have for website which is needed to be hosted.

'Delete the content which was already there.'



STEP 22: Website hosted



OUR TEAM

we love cloud

"None of us is as strong as all of us."

We believe that team work means more we and less me. We are implementing and monitoring Web Server through cloud computing using the market leader AWS (Amazon web Services). Cloud monitoring is the process of evaluating, monitoring, and managing cloud-based services, applications and infrastructure.



Aarav Raj Singh
(Logistic & Tech Lead)
Email:aresa.aarav@gmail.com



Vaibhav Manawal
(Tech Support & Documentation)
Email:shinmanwal@gmail.com



Vandit Jain
(Tech Member & Research)
Email:vanditjain11@gmail.com



Harsh Gupta
(Tech Support)
Email:harshc2303@gmail.com

SAMPLE CLOUD WATCH



CONTACT DETAILS

We respond within 48 hours!

- 📍 India
- 📞 Phone: +91 7319952159
- ✉️ Email: aresa.aarav@gmail.com

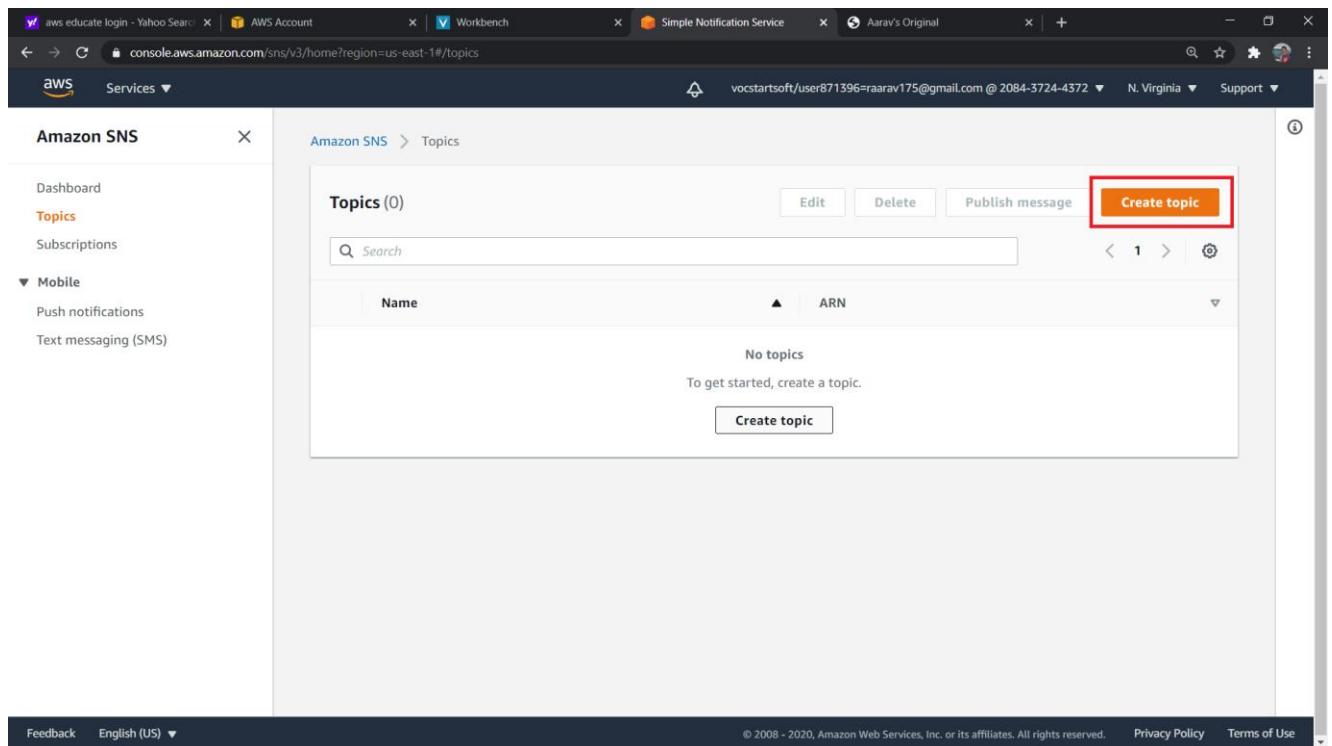


Now you can check your web server by opening the ipv4 address.

STEP 23: Go back to your AWS dashboard and search for SNS service under All Service section.

The screenshot shows the AWS Management Console with a search bar at the top containing 'SNS'. A red box highlights the search results, which include 'Simple Notification Service' and a description: 'SNS managed message topics for Pub/Sub'. The left sidebar shows 'Favorites' and 'Recently visited' sections. The main area displays a grid of AWS services categorized by color-coded boxes: EC2, Storage, Database, Customer Enablement, Blockchain, Satellite, Quantum Technologies, Management & Governance, Analytics, and various other services like SageMaker, Comprehend, Forecast, Fraud Detector, Kendra, Lex, Personalize, Polly, Rekognition, Transcribe, Translate, DeepComposer, DeepLens, DeepRacer, Organizations, CloudWatch, and Pinpoint. The 'Analytics' box is also highlighted with a red border.

STEP 24: click on create topic under topics in Amazon SNS.

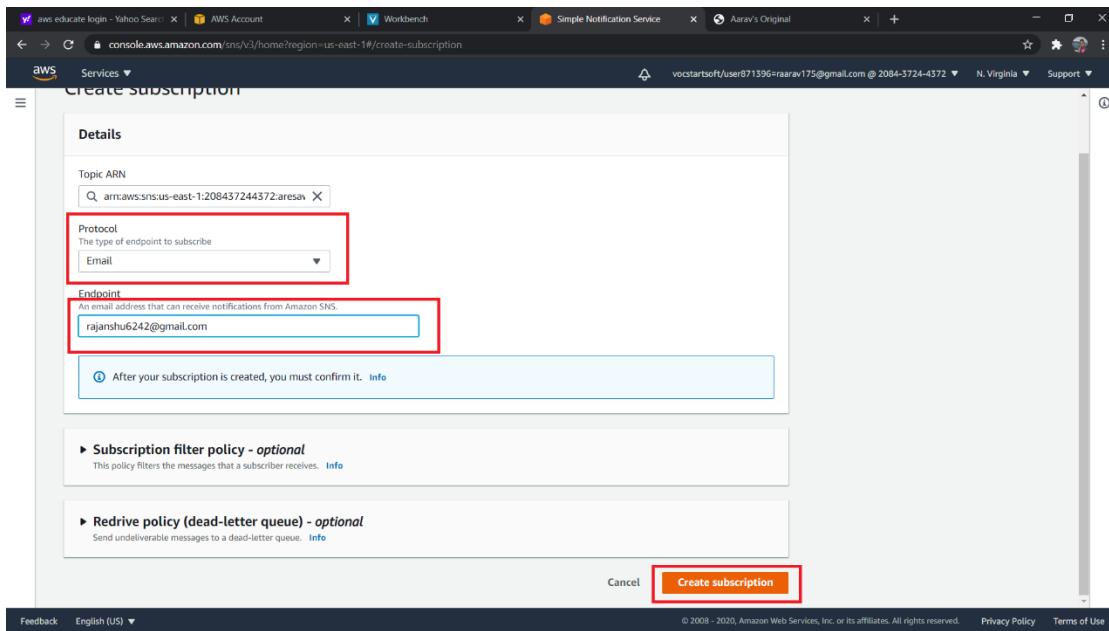
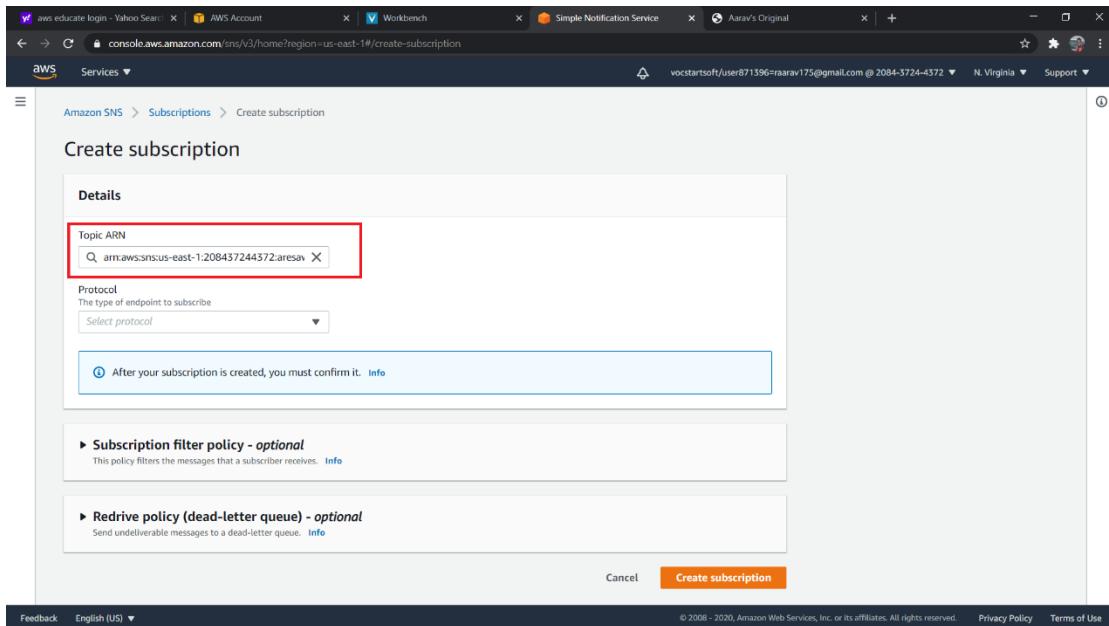


The screenshot shows the AWS SNS Topics page. On the left, there's a sidebar with links for Dashboard, Topics (which is currently selected and highlighted in orange), Subscriptions, Mobile (Push notifications and Text messaging (SMS)), Feedback, and Language (English (US)). The main content area has a header 'Topics (0)' with buttons for Edit, Delete, Publish message, and Create topic (which is highlighted with a red box). Below this is a search bar and a table with columns for Name and ARN. A message says 'No topics' and 'To get started, create a topic.' with a 'Create topic' button. At the bottom, there are links for Privacy Policy and Terms of Use, along with copyright information: '© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.'

STEP 25: Then successfully create your topic name and save it.

The screenshot shows the AWS SNS console in a browser window. The URL is `console.aws.amazon.com/sns/v3/home?region=us-east-1#/topic/arm:aws:sns:us-east-1:208437244372:aresaweb`. The top navigation bar includes tabs for 'aws educate login - Yahoo Search', 'AWS Account', 'Workbench', 'Simple Notification Service', and 'Aarav's Original'. The main content area has a green header bar with the message: 'Topic aresaweb created successfully. You can create subscriptions and send messages to them from this topic.' Below this, the topic details for 'aresaweb' are shown. The 'Details' section includes fields for 'Name' (aresaweb), 'Display name' (Aresa), 'ARN' (arn:aws:sns:us-east-1:208437244372:aresaweb), and 'Topic owner' (208437244372). A red box highlights the ARN field. At the bottom, there are tabs for 'Subscriptions', 'Access policy', 'Delivery retry policy (HTTP/S)', 'Delivery status logging', 'Encryption', and 'Tags'. The 'Subscriptions' tab is selected, showing '(0)' and a 'Create subscription' button. A red box highlights the 'Create subscription' button. The left sidebar shows navigation links: 'Dashboard', 'Topics' (which is selected and highlighted in orange), 'Subscriptions', 'Mobile' (with 'Push notifications' and 'Text messaging (SMS)'), and 'Services' dropdown.

STEP 26: Now click on subscription and paste your ARN (Amazon Resource Number) and enter EMAIL as protocol and your email id in Endpoint and click on create subscription.



STEP 27: subscription created successfully. But the confirmation is still pending.

The screenshot shows the AWS SNS console with a successful subscription message highlighted in a red box:

Subscription to aresaweb created successfully.
The ARN of the subscription is arn:aws:sns:us-east-1:208437244372:aresaweb:cd354f69-7654-4a41-a70c-cd8aec63239b.

The main details of the subscription are:

- ARN:** arn:aws:sns:us-east-1:208437244372:aresaweb:cd354f69-7654-4a41-a70c-cd8aec63239b
- Status:** Pending confirmation (highlighted in red)
- Protocol:** EMAIL
- Endpoint:** rajanshu6242@gmail.com
- Topic:** aresaweb

Below the details, there are tabs for "Subscription filter policy" (highlighted in red) and "Redrive policy (dead-letter queue)".

STEP 28: Go to your Gmail and click on confirm subscription.

The screenshot shows a Gmail inbox with several tabs at the top: 'aws educate login - Yahoo S...', 'AWS Account', 'Workbench', 'Simple Notification Service', 'Aarav's Original', and 'AWS Notification - Subscript...'. The 'Inbox' tab is selected, showing 50 messages. A single message from 'Aresa <no-reply@sns.amazonaws.com>' is highlighted with a red box. The subject of the message is 'AWS Notification - Subscription Confirmation'. The message body contains the following text:

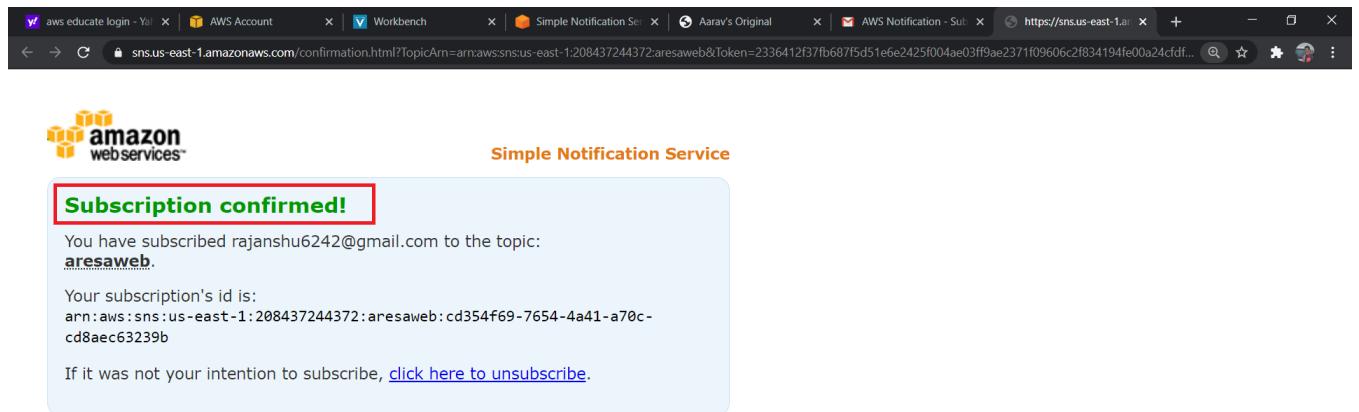
You have chosen to subscribe to the topic:
arn:aws:sns:us-east-1:208437244372:aresaweb

To confirm this subscription, click or visit the link below (If this was in error no action is necessary):
[Confirm subscription](https://sns.us-east-1.amazonaws.com/confirmation.html?TopicArn=arn:aws:sns:us-east-1:208437244372:aresaweb)

Please do not reply directly to this email. If you wish to remove yourself from receiving all future SNS subscription confirmation requests please send an email to [sns-opt-out](#)

At the bottom of the message are 'Reply' and 'Forward' buttons. The URL <https://sns.us-east-1.amazonaws.com/confirmation.html?TopicArn=arn:aws:sns:us-east-1:208437244372:aresaweb> is visible in the browser's address bar.

STEP 29: Then you got the message “Subscription Confirmed”. As shown below



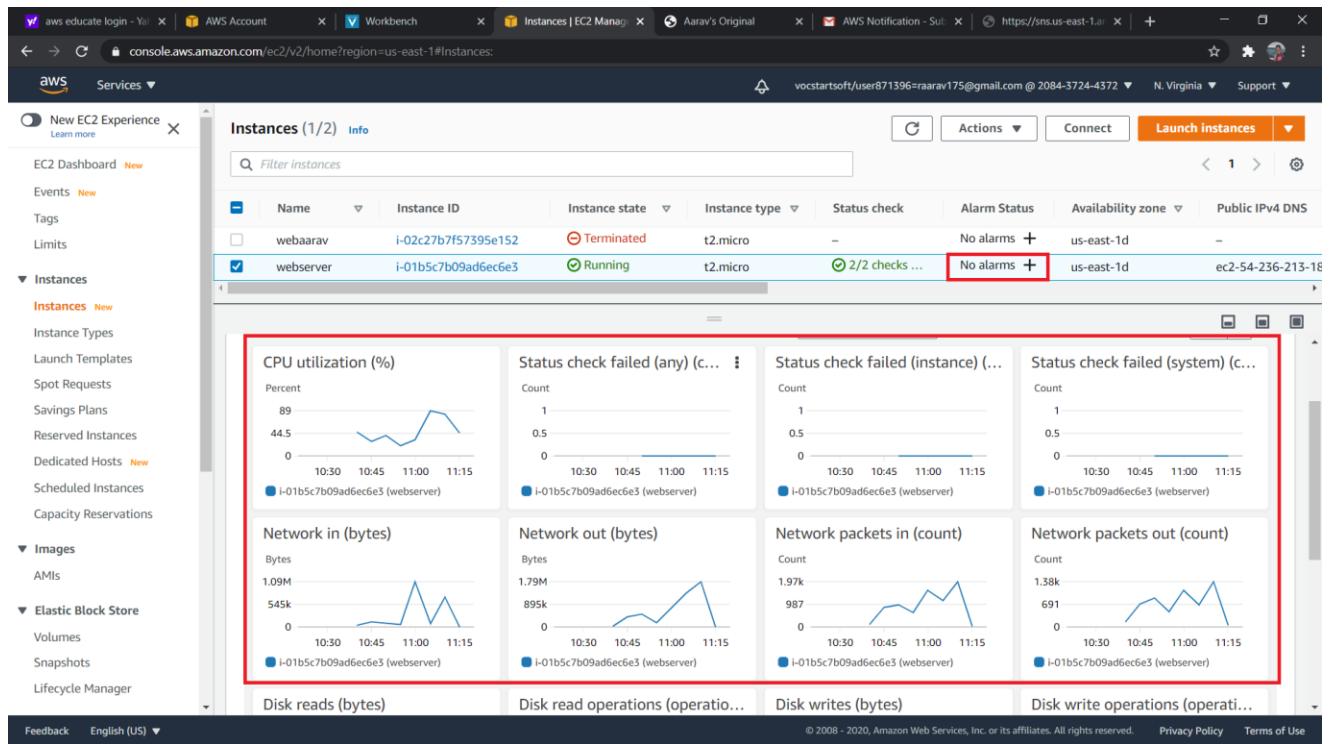
STEP 30: Go to your Ec2 dashboard and click on monitoring.

The screenshot shows the AWS EC2 Instances page. On the left, there's a navigation sidebar with links like EC2 Dashboard, Events, Tags, Limits, Instances (selected), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts (New), Scheduled Instances, Capacity Reservations, Images, AMIs, and Elastic Block Store. The main area displays a table of instances. One instance, named 'webserver' with ID i-01b5c7b09ad6ec6e3, is selected and shown in more detail. The 'Monitoring' tab in the instance details section is highlighted with a red box. The instance summary table includes columns for Instance ID, Instance state, Instance type, Public IPv4 address, Private IPv4 addresses, Public IPv4 DNS, Private IPv4 DNS, VPC ID, and Subnet ID.

Name	Instance ID	Instance state	Instance type	Status check	Alarm Status	Availability zone	Public IPv4 DNS
webaarav	i-02c27b7f57395e152	Terminated	t2.micro	-	No alarms +	us-east-1d	-
webserver	i-01b5c7b09ad6ec6e3	Running	t2.micro	2/2 checks ...	No alarms +	us-east-1d	ec2-54-236-213-18

Instance summary	
Instance ID	i-01b5c7b09ad6ec6e3 (webserver)
Instance state	Running
Instance type	t2.micro
VPC ID	vpc-eb53a596
Public IPv4 address	54.236.213.181 open address
Private IPv4 addresses	172.31.82.145
Public IPv4 DNS	ec2-54-236-213-181.compute-1.amazonaws.com open address
Private IPv4 DNS	ip-172-31-82-145.ec2.internal
Subnet ID	subnet-1264c433

STEP 31: No alarm is created for this instance till now, and the CPU utilization is also very less.(Refer to image given below)



STEP 32: Search for cloud watch under All service section.

The screenshot shows the AWS CloudWatch search results in the All services section. A red box highlights the search bar and the 'CloudWatch' result. The search bar contains the text 'cloudwatch'. Below it, the 'CloudWatch' result is listed under 'Monitor Resources and Applications'. The page also displays other services categorized by type, such as Storage, Database, and Analytics.

All services

CloudWatch

Monitor Resources and Applications

Amazon EventBridge

Serverless event bus that connects application data from your own apps, SaaS, and AWS services

Lambda Customer Enablement Amazon CodeGuru

Batch AWS IQ Amazon Comprehend

Elastic Beanstalk Support Amazon Forecast

Serverless Application Repository Managed Services Amazon Fraud Detector

AWS Outposts Blockchain Amazon Kendra

EC2 Image Builder Amazon Managed Blockchain Amazon Lex

Storage Satellite Amazon Personalize

S3 Amazon Rekognition

EFS Ground Station Amazon Transcribe

FSx Amazon Braket Amazon Translate

S3 Glacier Management & Governance AWS DeepComposer

Storage Gateway AWS Organizations AWS DeepLens

AWS Backup CloudWatch AWS DeepRacer

Database AWS Auto Scaling Analytics

RDS CloudFormation Athena

DynamoDB CloudTrail Amazon Redshift

ElastiCache EMR

Neptune

Customer Engagement

Amazon Connect

Pinpoint

Simple Email Service

Business Applications

AWS Cost Explorer

AWS Budgets

AWS Marketplace Subscriptions

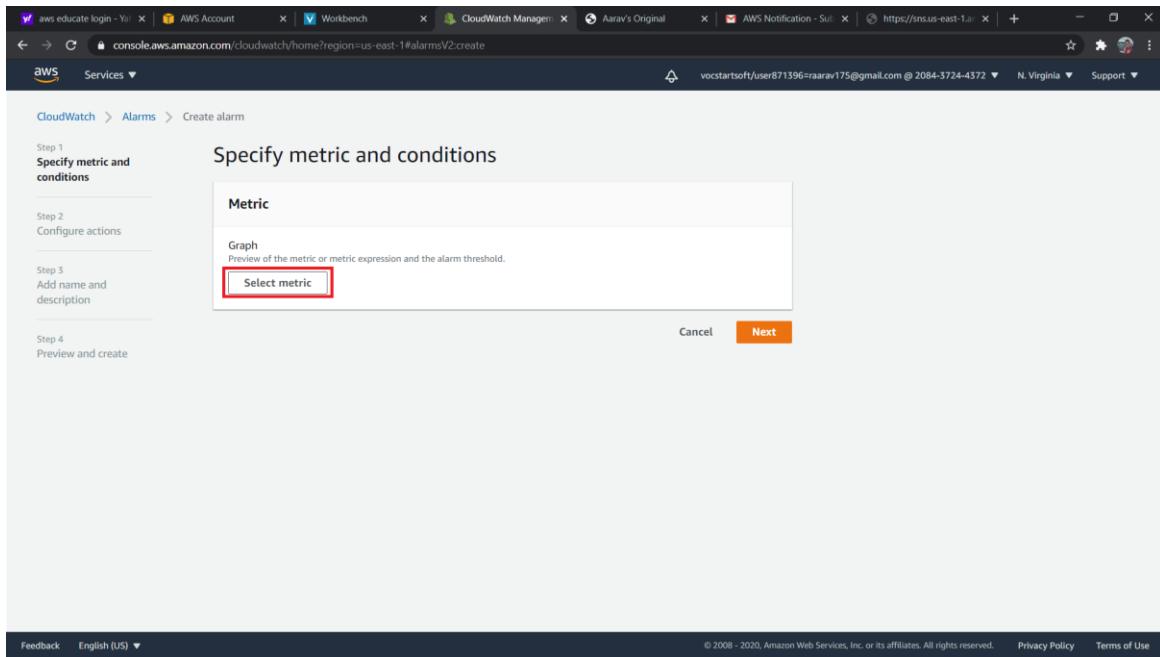
© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Privacy Policy Terms of Use

STEP 33: click on Alarm and then click create alarm.

The screenshot shows the AWS CloudWatch Alarms page. On the left, there is a navigation sidebar with various options like CloudWatch, Dashboards, Alarms, INSUFFICIENT, OK, Billing, Logs, Log groups, Insights, Metrics, Events, Rules, Event Buses, ServiceLens, Service Map, Traces, Container Insights, Resources, Performance Monitoring, Synthetics, Canaries, Contributor Insights, Settings, Favorites, and Add a dashboard. The 'Alarms' option is selected and highlighted with a red box. On the right, the main content area displays the 'Alarms (0)' section. It includes a search bar, filter buttons for 'In alarm', 'Any type', and 'Actions', and a prominent orange 'Create alarm' button which is also highlighted with a red box. Below these are columns for Name, State, Last state update, Conditions, and Actions. A message at the bottom of the table says 'Loading alarms'.

STEP 34: Click on 'Select metric'.

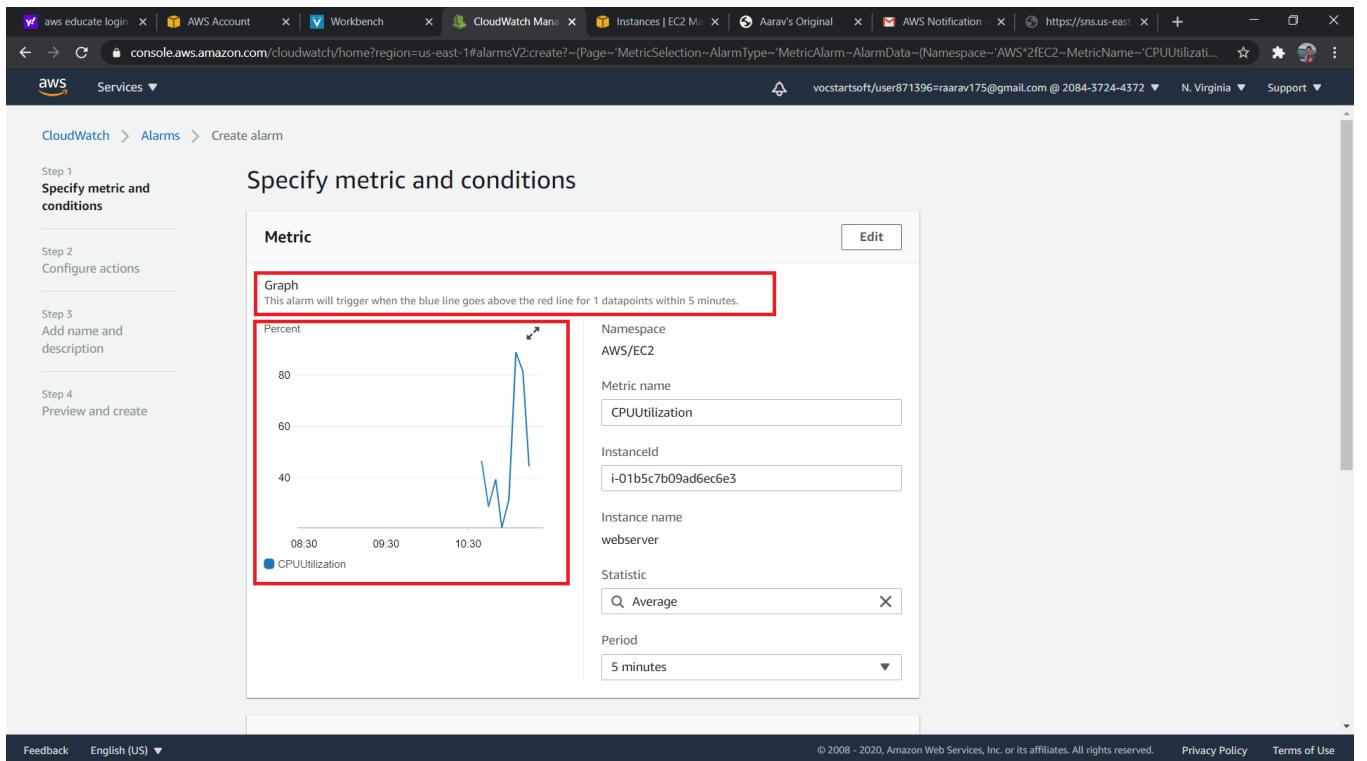


STEP 35: Choose EC2>> Per Instance Metrics and carefully select the metric name corresponding to your instance name.

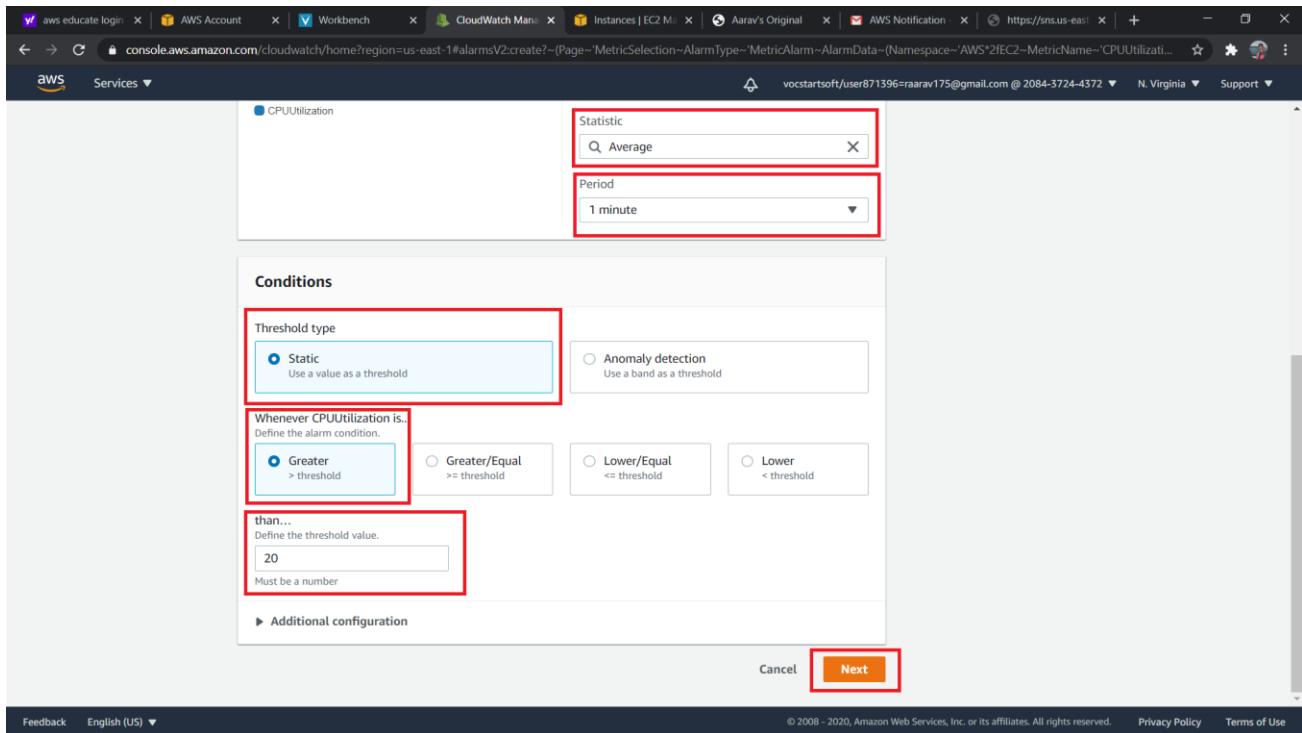
The screenshot shows the AWS CloudWatch Metrics selection interface. At the top, there's a navigation bar with tabs like 'All metrics', 'Graphed metrics (1)', 'Graph options', and 'Source'. Below the navigation bar, there's a search bar and a breadcrumb trail: 'All > EC2 > Per-Instance Metrics'. A table lists metrics for various instances. One row for 'webserver' is selected, indicated by a red box around the checkbox and the row. The selected row shows the metric 'CPUUtilization'. At the bottom right of the interface, there are 'Cancel' and 'Select metric' buttons, with the 'Select metric' button also highlighted by a red box.

Instance Name (51)	InstanceId	Metric Name
webaarav	i-02c27b7f157395e152	CPUSurplusCreditsCharged
webserver	i-01b5c7b09ad6ec6e3	NetworkPacketsIn
webserver	i-01b5c7b09ad6ec6e3	NetworkPacketsOut
<input checked="" type="checkbox"/> webserver	i-01b5c7b09ad6ec6e3	CPUUtilization
webserver	i-01b5c7b09ad6ec6e3	NetworkIn
webserver	i-01b5c7b09ad6ec6e3	NetworkOut
webserver	i-01b5c7b09ad6ec6e3	DiskReadBytes
webserver	i-01b5c7b09ad6ec6e3	DiskWriteBytes
webserver	i-01b5c7b09ad6ec6e3	DiskReadOps

STEP 36: watch out for the graph and the threshold will trigger an alarm only when blue line will exceed red line.



STEP 37: Select the static condition and put the threshold as eg.20(our case), watch out for time period as well and click on “Next”.



STEP 38: Select where to send notification and choose your topic.

The screenshot shows the AWS CloudWatch Metrics V2 Create Alarm interface at Step 2: Configure actions. The 'Notification' section is the primary focus, with three trigger options: 'In alarm' (selected), 'OK', and 'Insufficient data'. Below this, the 'Select an SNS topic' section is highlighted, showing three options: 'Select an existing SNS topic' (selected), 'Create new topic', and 'Use topic ARN'. A search bar contains the text 'aresaweb'. An email endpoint 'rajanshu6242@gmail.com' is listed with a link to 'View in SNS Console'. At the bottom, there's an 'Add notification' button and an 'Auto Scaling action' section.

STEP 39: Add Alarm Name and description.

The screenshot shows the AWS CloudWatch Metrics Alarm creation wizard. The current step is "Step 3: Add name and description". The page title is "Add name and description". A red box highlights the "Alarm name" input field, which contains the value "ALERT!". Below it is an optional "Alarm description" field containing the text "Warning: Threshold is crossing 20 mark.". At the bottom right are "Cancel", "Previous", and "Next" buttons, with "Next" being highlighted by a red box. On the left sidebar, the steps are listed: Step 1 (Specify metric and conditions), Step 2 (Configure actions), Step 3 (Add name and description, currently selected), and Step 4 (Preview and create).

STEP 40: Look at Conditions and configure actions, review alarm and click on create alarm.

The screenshot shows the AWS CloudWatch Metrics V2 Create Alarm wizard. The current step is Step 2: Configure actions. The interface is divided into three main sections:

- Step 1: Set conditions** (Top section):
 - Threshold type: Static
 - Whenever CPUUtilization is Greater (>) than... 20
 - Additional configuration (link)
- Step 2: Configure actions** (Middle section):
 - Actions
 - Notification: When In alarm, send a notification to "aresaweb"
- Step 3: Add name and description** (Bottom section):
 - Name and description
 - Name: ALERT!
 - Description: Warning: Threshold is crossing 20 mark.

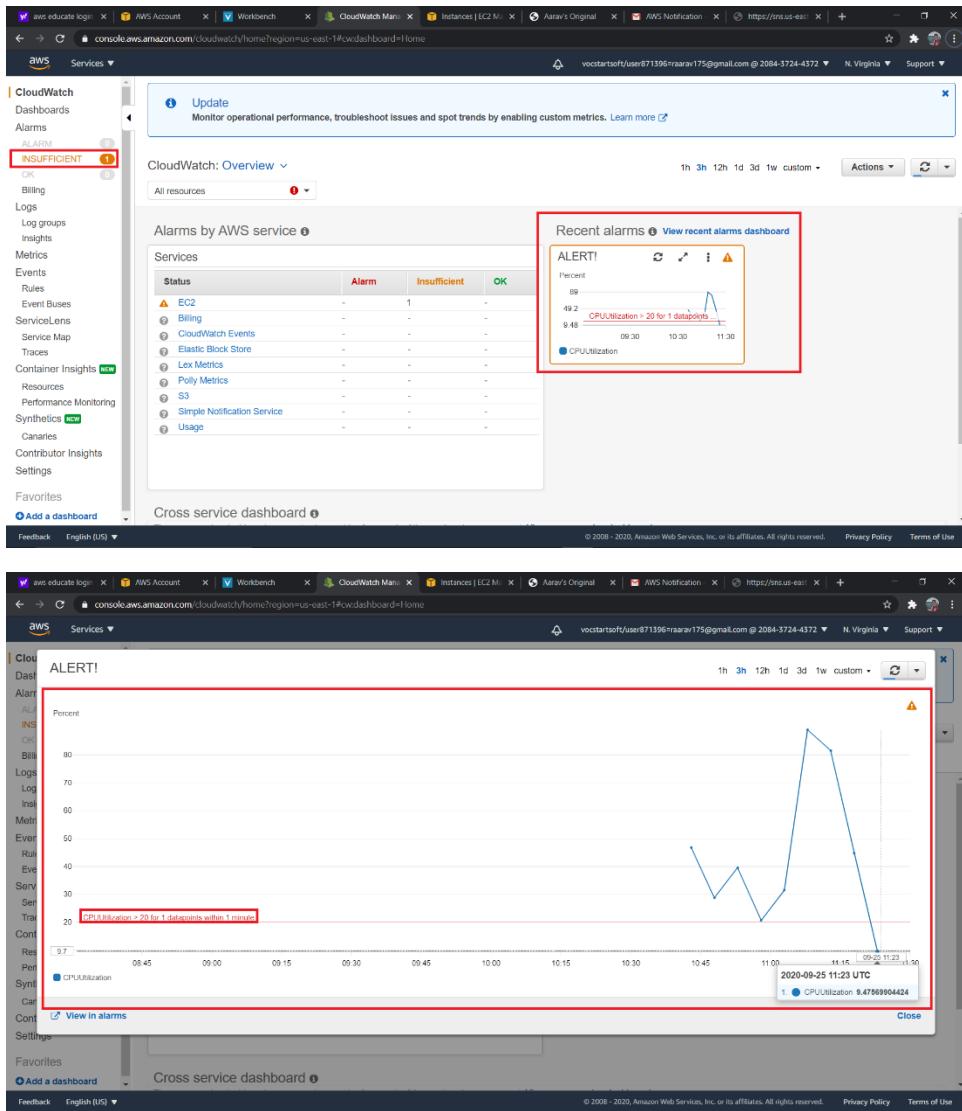
A red box highlights the "Conditions" and "Actions" sections of Step 2. A second red box highlights the "Name and description" section of Step 3. The "Create alarm" button is highlighted with a red box at the bottom right of the "Name and description" section.

STEP 41: Alarm Alert has been successfully created and click on INSUFFICIENT or Dashboard on left side.

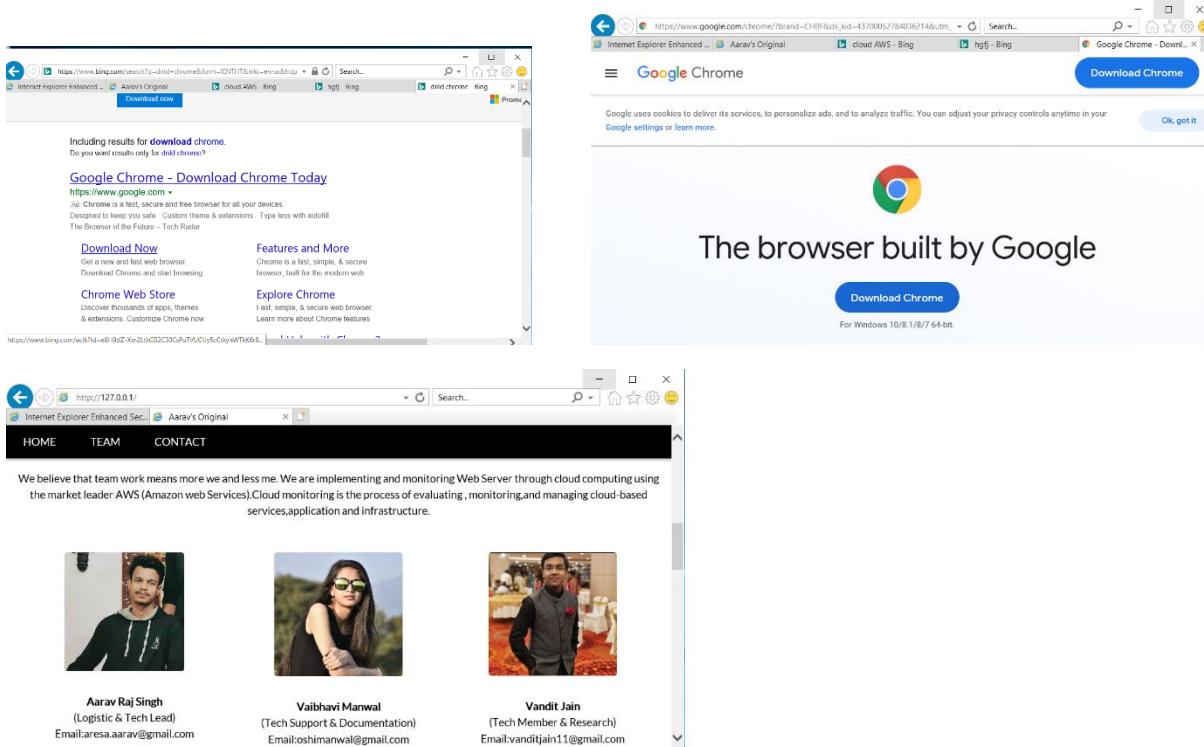
The screenshot shows the AWS CloudWatch Alarms interface. A green success message at the top states "Successfully created alarm ALERT!". On the left sidebar, under the "ALARMS" section, the "INSUFFICIENT" link is highlighted with a red box. The main content area displays a table titled "Alarms (1/1)". The table has columns for Name, State, Last state update, Conditions, and Action. One row is shown, corresponding to the newly created alarm "ALERT!". The "Conditions" column for this row indicates "CPUUtilization > 20 for 1 datapoints within 1 minute". The entire row for "ALERT!" is also highlighted with a red box.

Name	State	Last state update	Conditions	Action
ALERT!	Insufficient data	2020-09-25 17:00:43	CPUUtilization > 20 for 1 datapoints within 1 minute	1 action

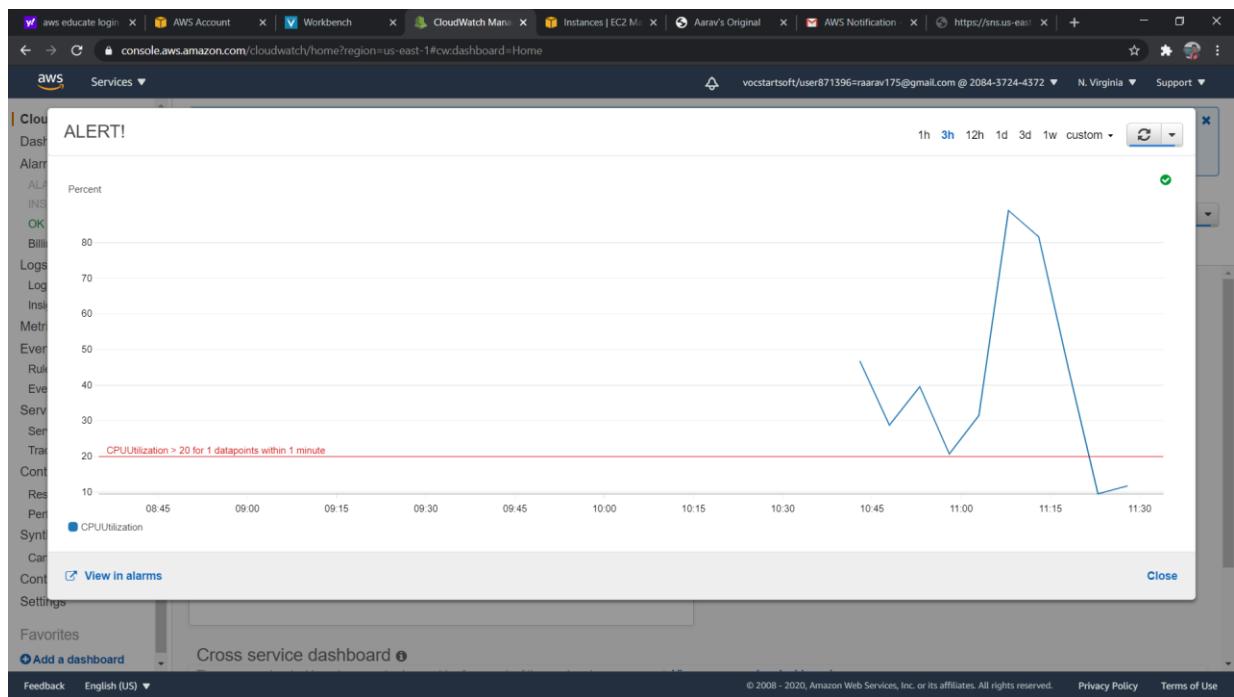
STEP 42: Enlarge the recent alarm and notice that the threshold is on 0, and to trigger alarm alert we need to make threshold reach 20(i.e. above red line).



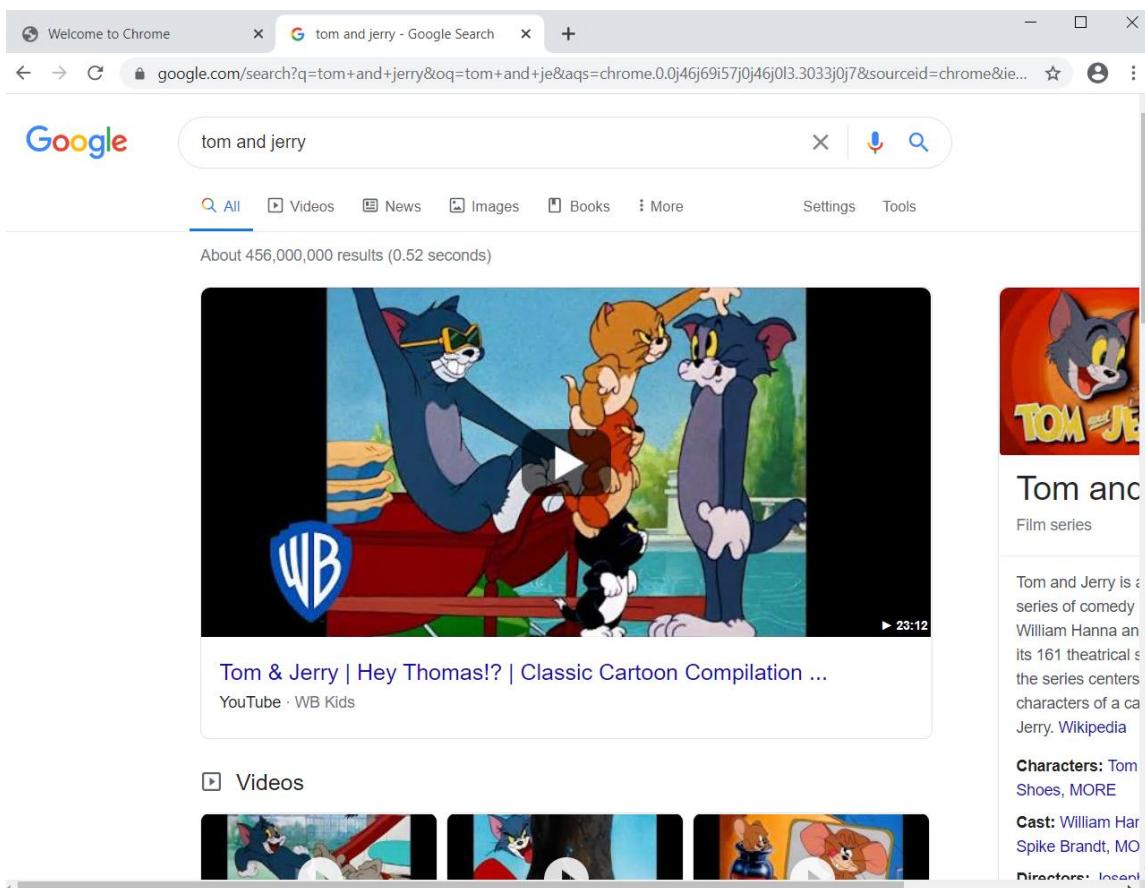
STEP 43: To increase threshold open virtual machine and run a few applications or search in browsers or download something to increase threshold of web server (i.e. virtual machine).



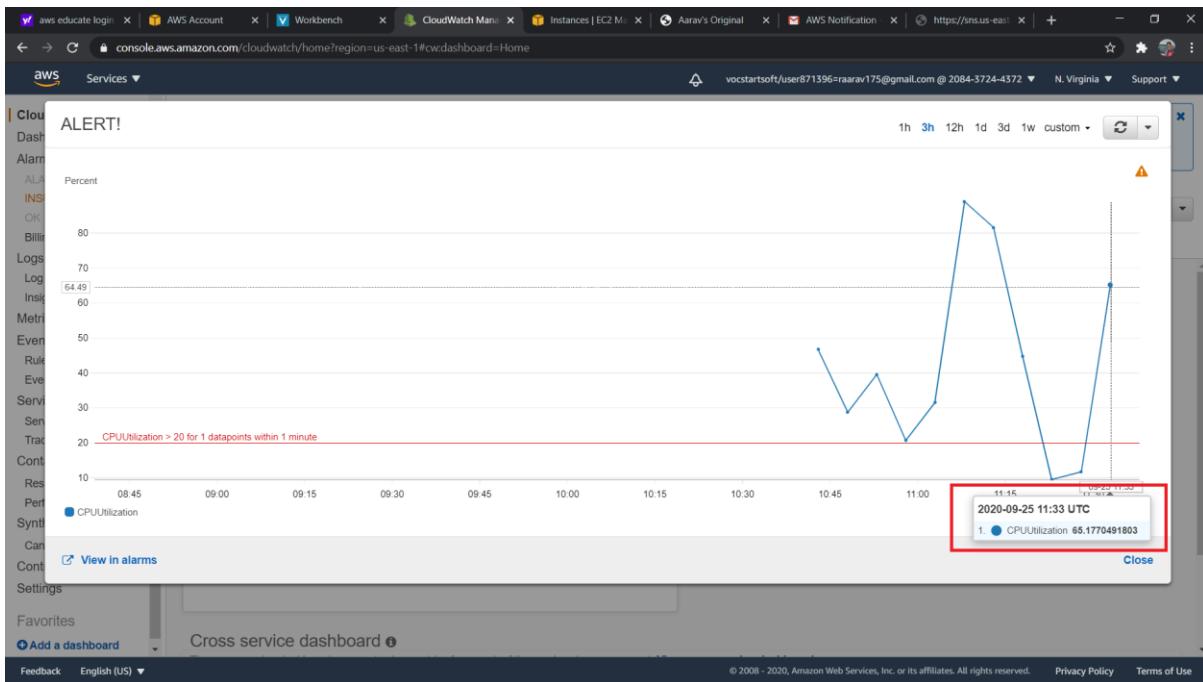
STEP 44: We can see threshold rising a little on dashboard of AWS CloudWatch.



STEP 45: play a video to increase the threshold on our web server (virtual machine) or create traffic on website to trigger the alarm.



STEP 46: since the threshold is increased (>20) and the blue line exceeds the red, we created the condition to trigger the alarm.



STEP 47: And finally, we got the alarm in the form of mail which was created with name 'Aresa'.

The screenshot shows a Gmail inbox with several tabs at the top: 'aws educate login', 'AWS Account', 'Workbench', 'CloudWatch Mane', 'Instances | EC2 M...', 'Aarav's Original', 'ALARM: "ALERT!" | https://sns.us-eas...', and a blank tab. The 'ALARM: "ALERT!" | https://sns.us-eas...' tab is active. The inbox list shows an email from 'Aresa <no-reply@sns.amazonaws.com>' with the subject 'ALARM: "ALERT!" in US East (N. Virginia)'. The email body contains the following text:

You are receiving this email because your Amazon CloudWatch Alarm "ALERT!" in the US East (N. Virginia) region has entered the ALARM state, because "Threshold Crossed: 1 out of the last 1 datapoints [65.17704918032788 (25/09/20 11:33:00)] was greater than the threshold (20.0) (minimum 1 datapoint for OK -> ALARM transition)" at "Friday 25 September, 2020 11:39:28 UTC".

View this alarm in the AWS Management Console:
<https://us-east-1.console.aws.amazon.com/cloudwatch/home?region=us-east-1#s=Alarms&alarm=ALERT!>

Alarm Details:

- Name: ALERT!
- Description: Warning: Threshold is crossing 20 mark.
- State Change: INSUFFICIENT_DATA -> ALARM
- Reason for State Change: Threshold Crossed: 1 out of the last 1 datapoints [65.17704918032788 (25/09/20 11:33:00)] was greater than the threshold (20.0) (minimum 1 datapoint for OK -> ALARM transition).
- Timestamp: Friday 25 September, 2020 11:39:28 UTC
- AWS Account: 208437244372
- Alarm Arn: arn:aws:cloudwatch:us-east-1:208437244372:alarm:ALERT!

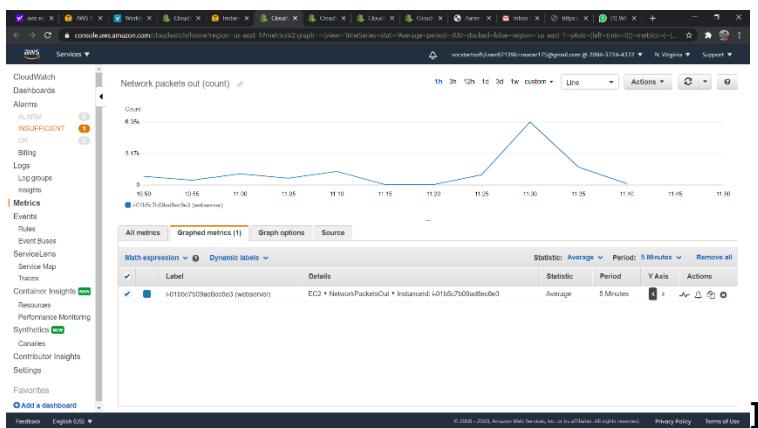
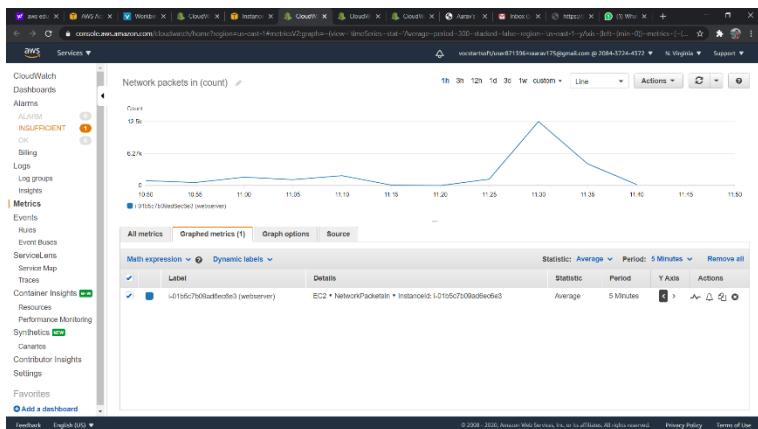
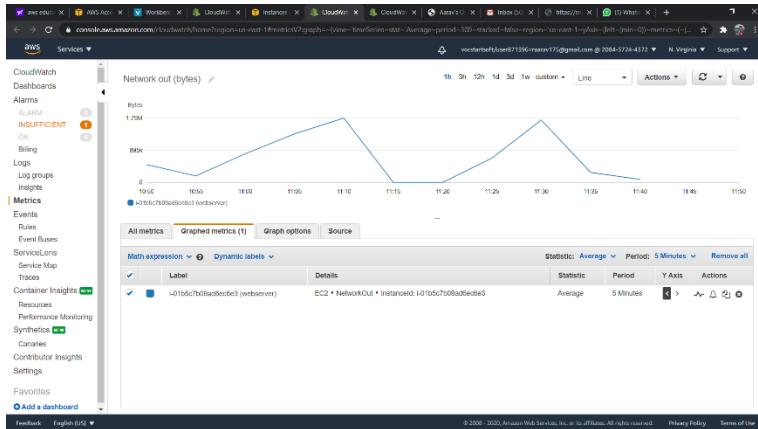
Threshold:

- The alarm is in the ALARM state when the metric is GreaterThanThreshold 20.0 for 60 seconds.

Monitored Metric:

- MetricNamespace: AWS/EC2
- MetricName: CPUUtilization
- Dimensions: [InstanceId = i-01b5c7b09ad6ec6e3]
- Period: 60 seconds
- Statistic: Average
- Unit: not specified
- TreatMissingData: missing

STEP 48: We can see other plots as well.



Etc.



HENCE, WE **SUCCESSFULLY** IMPLEMENTED AND MONITORED THE WEB SERVER ON AWS (USING EC2, SNS and CLOUDWATCH).