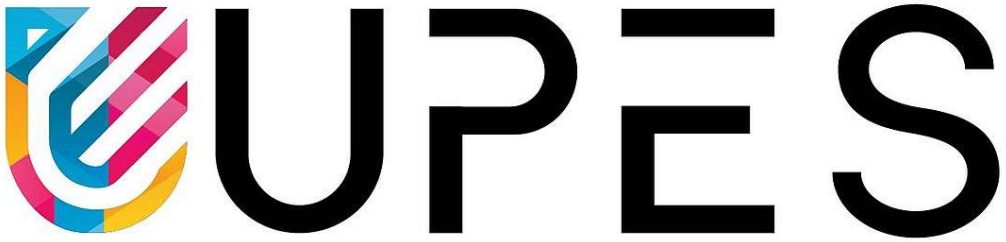


CLOUD SECURITY & MANAGEMENT LAB

Name: Harsh Ranjan

SAP ID: 500097019

Roll no: R2142211262

Batch :B7

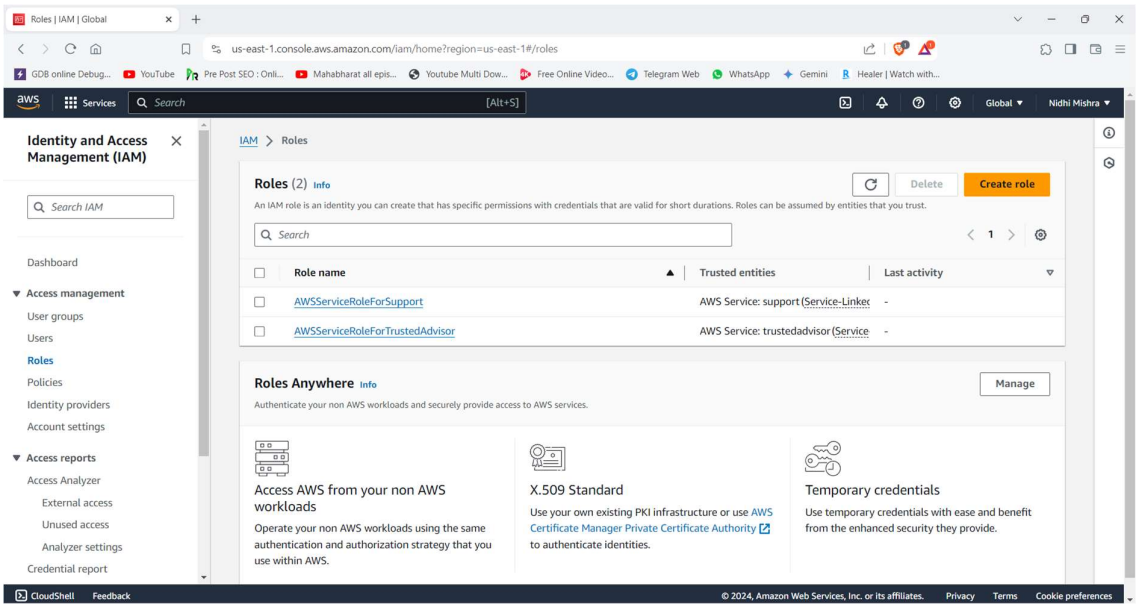
SUBMISSION TO:

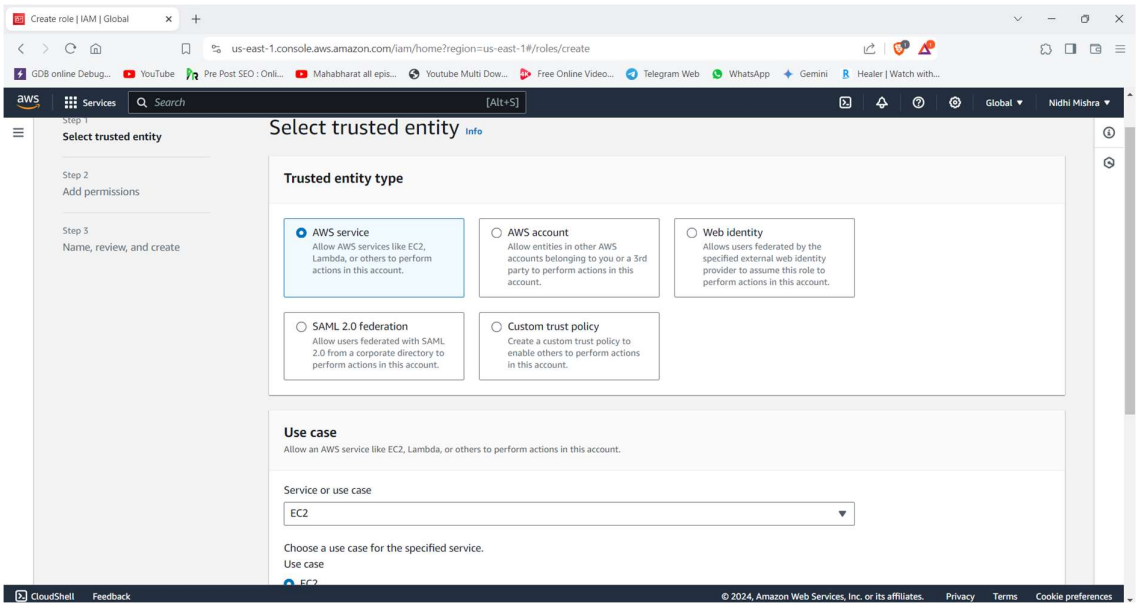
Ms. Avita Katal

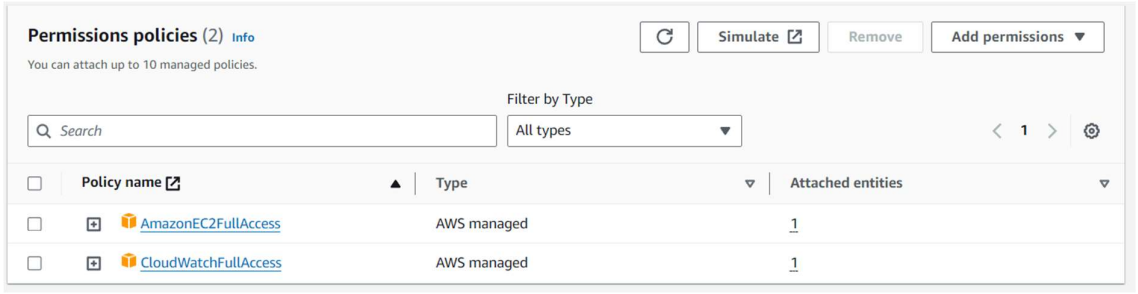
Experiment 6 a)-Cloud Monitoring Tool Cloudwatch integrated with EC2

STEP 1: Firstly, go to IAM and create a role for ec2 instance for these following 2 access controls:

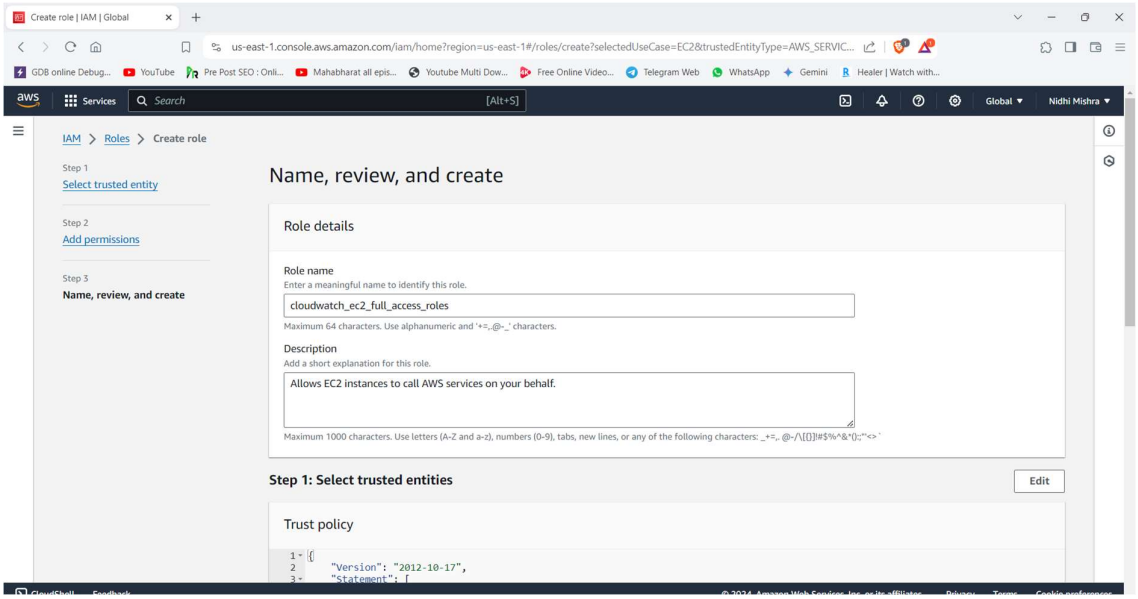
- Ec2 Full Access
- CloudWatch Full Access



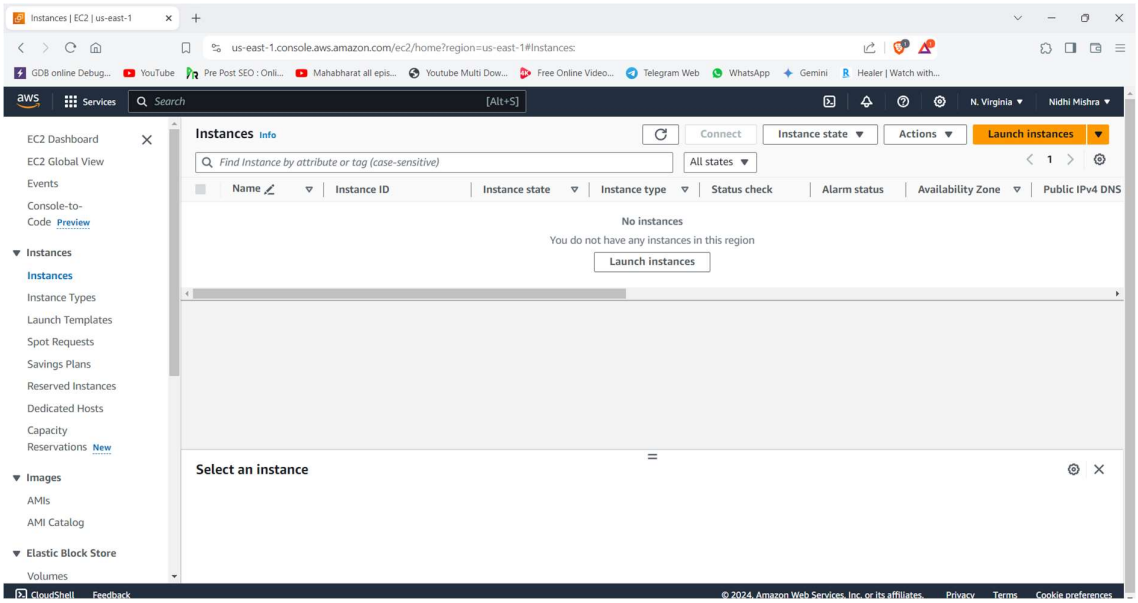


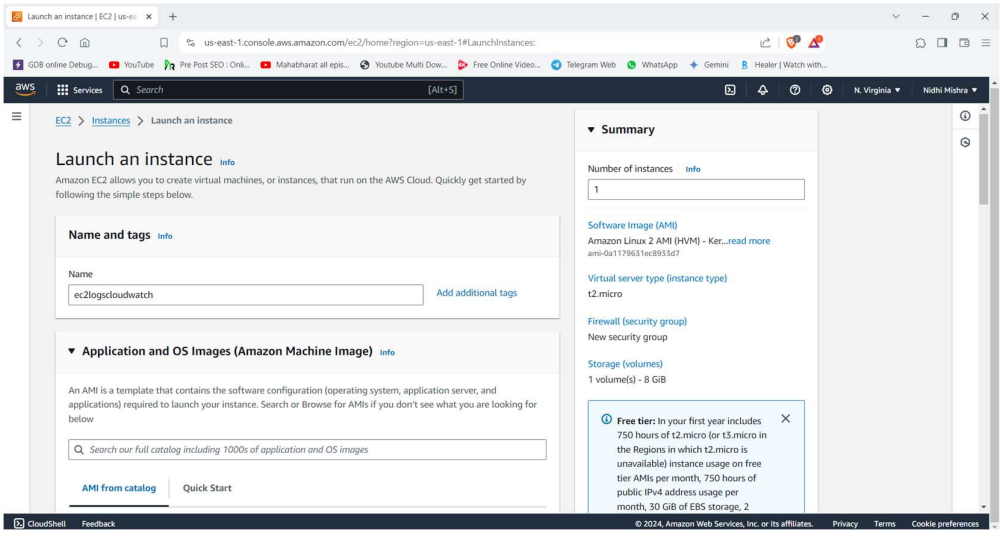


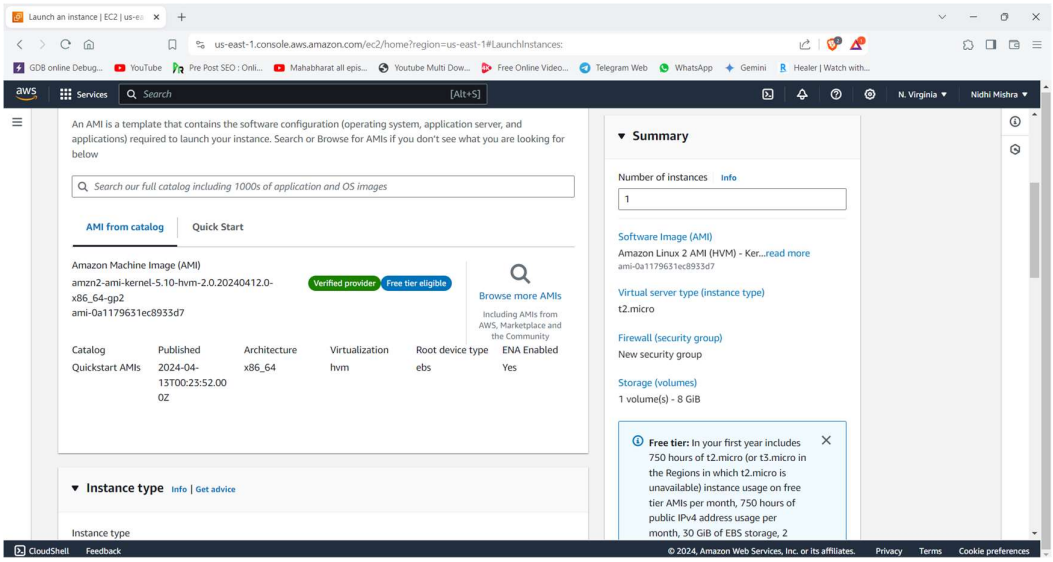
STEP 2 : Assign the name of the role.

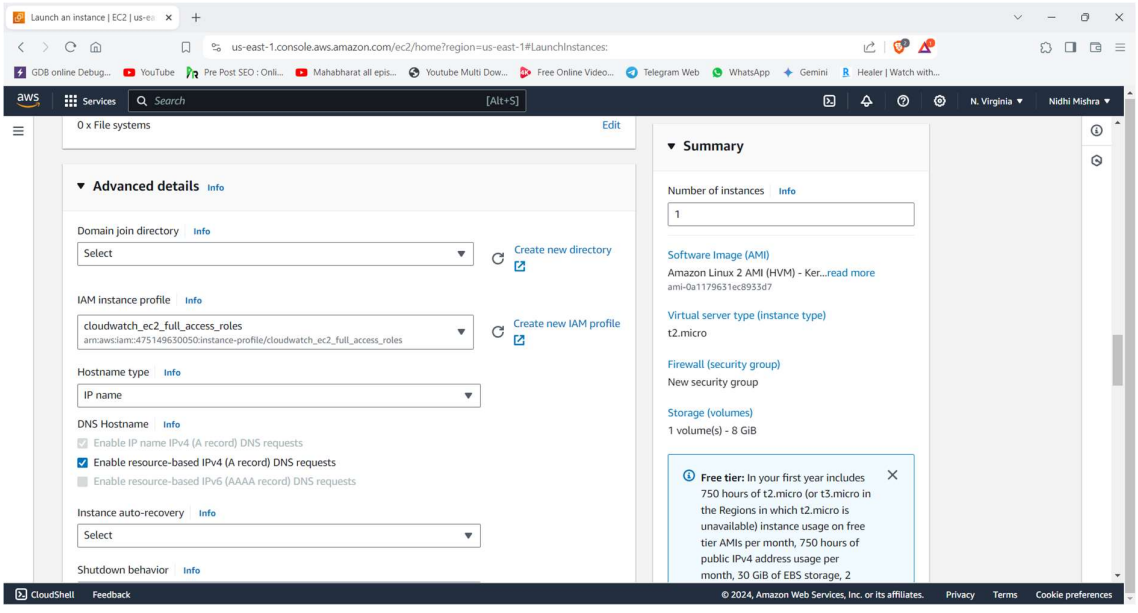


STEP 3: Now go to EC2 tab and launch a Amazon AMI 2 Machine and attach the created IAM role to that ec2 in advanced tab.

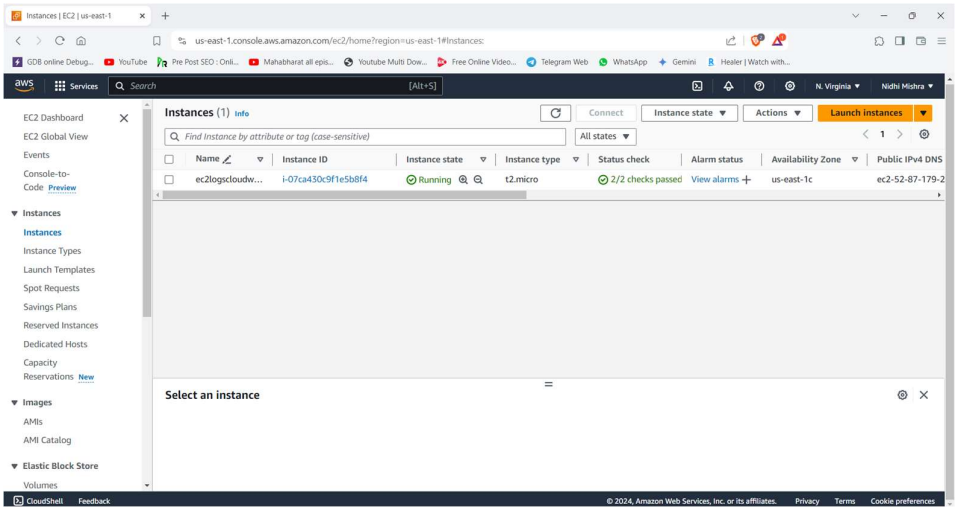



Launch an instance | EC2 | us-east-1
us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#LaunchInstances:
Services
Search
[Alt+S]
N. Virginia
Nidhi Mishra
EC2 > Instances > Launch an instance
Launch an instance Info
Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.
Name and tags Info
Name
ec2logscloudwatch
Add additional tags
Application and OS Images (Amazon Machine Image) Info
An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below
Search our full catalog including 1000s of application and OS images
AMI from catalog
Quick Start
Summary
Number of instances Info
1
Software Image (AMI)
Amazon Linux 2 AMI (HVM) - Ker...read more
ami-0a1179631ec8933d7
Virtual server type (instance type)
t2.micro
Firewall (security group)
New security group
Storage (volumes)
1 volume(s) - 8 GiB
Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


Launch an instance | EC2 | us-east-1
us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#LaunchInstances:
Services
Search
[Alt+S]
N. Virginia
Nidhi Mishra
An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below
Search our full catalog including 1000s of application and OS images
AMI from catalog
Quick Start
Amazon Machine Image (AMI)
amzn2-ami-kernel-5.10-hvm-2.0.20240412.0-x86_64-gp2
ami-0a1179631ec8933d7
Verified provider
Free tier eligible
Browse more AMIs
Including AMIs from AWS, Marketplace and the Community
Catalog
Quickstart AMIs
Published
2024-04-13T00:23:52.000Z
Architecture
x86_64
Virtualization
hvm
Root device type
ebs
ENA Enabled
Yes
Instance type Info | Get advice
Instance type
Summary
Number of instances Info
1
Software Image (AMI)
Amazon Linux 2 AMI (HVM) - Ker...read more
ami-0a1179631ec8933d7
Virtual server type (instance type)
t2.micro
Firewall (security group)
New security group
Storage (volumes)
1 volume(s) - 8 GiB
Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

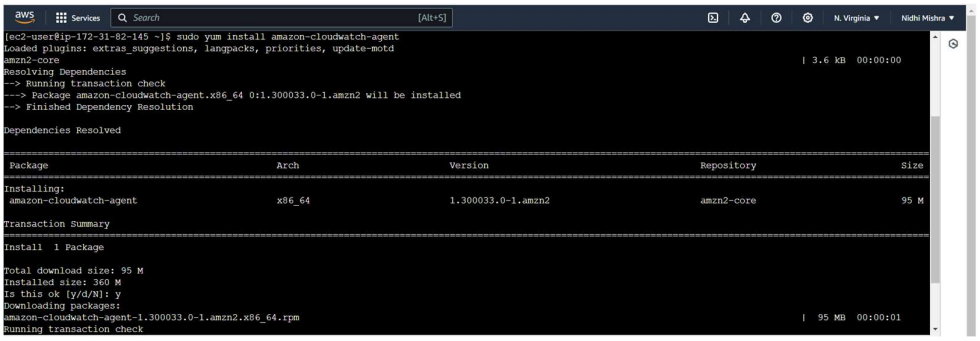

Launch an instance | EC2 | us-east-1
us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#LaunchInstances:
Services
Search
[Alt+S]
N. Virginia
Nidhi Mishra
0 x File systems
Edit
Advanced details Info
Domain join directory Info
Select
Create new directory
IAM instance profile Info
cloudwatch_ec2_full_access_roles
arn:aws:iam::475149630050:instance-profile/cloudwatch_ec2_full_access_roles
Create new IAM profile
Hostname type Info
IP name
DNS Hostname Info
Enable IP name IPv4 (A record) DNS requests
Enable resource-based IPv4 (A record) DNS requests
Enable resource-based IPv6 (AAAA record) DNS requests
Instance auto-recovery Info
Select
Shutdown behavior Info
Summary
Number of instances Info
1
Software Image (AMI)
Amazon Linux 2 AMI (HVM) - Ker...read more
ami-0a1179631ec8933d7
Virtual server type (instance type)
t2.micro
Firewall (security group)
New security group
Storage (volumes)
1 volume(s) - 8 GiB
Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

STEP 4: Now wait for the instance to get ready and then connect it.



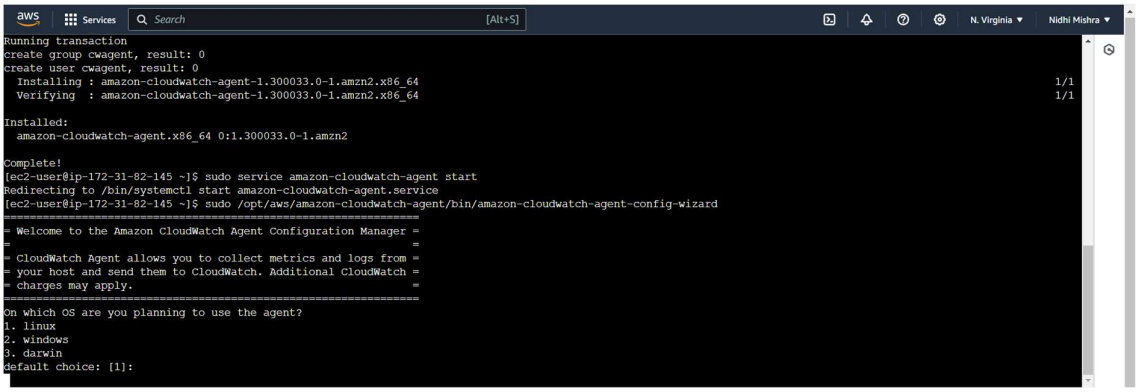
STEP 5: Use the following command to install CloudWatch agent in it.

- sudo yum install amazon-cloudwatch-agent

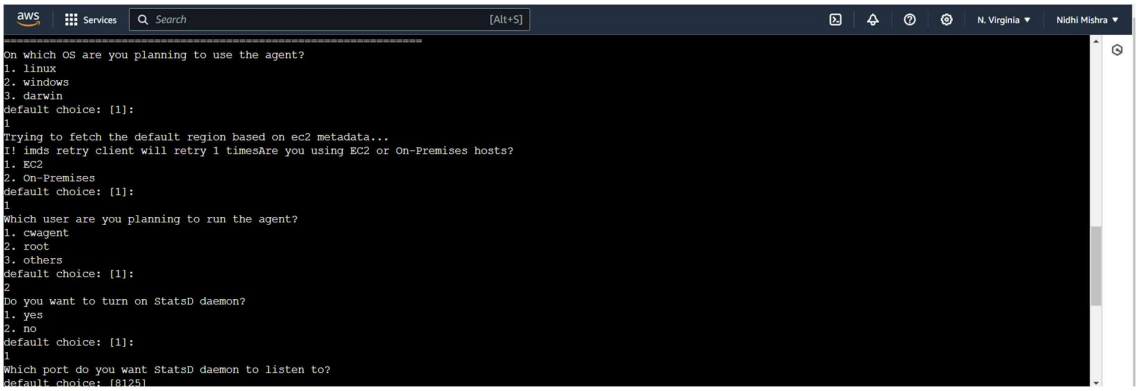


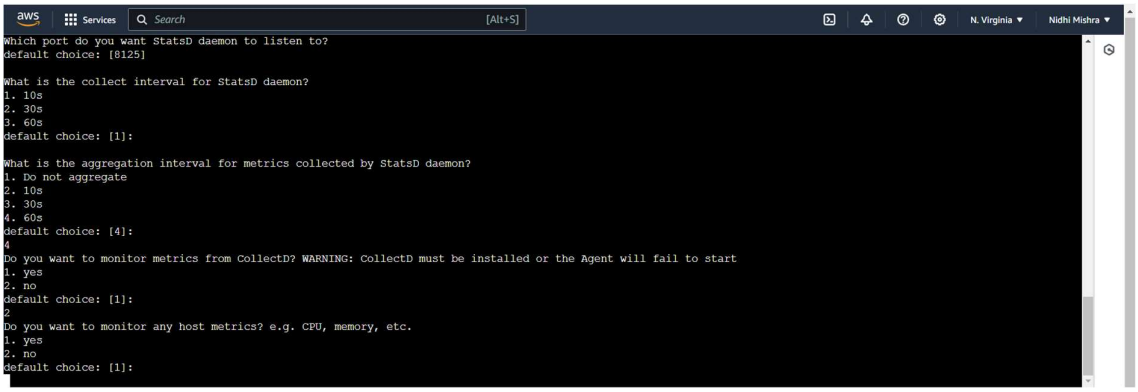
STEP 6: Now open the CloudWatch agent using the following command:

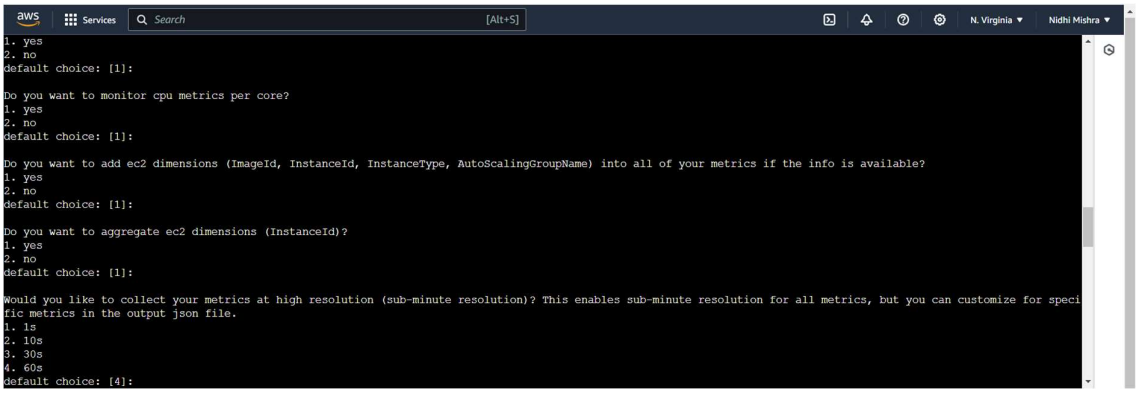
- sudo /opt/aws/amazon-cloudwatch-agent/bin/amazon-cloudwatch-agent-config-wizard



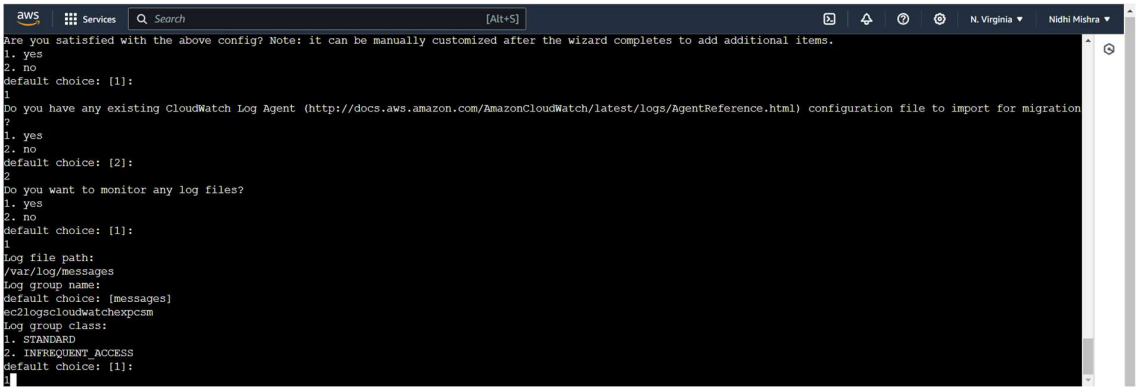
STEP 7: Now Choose the options as per your need or as shown below

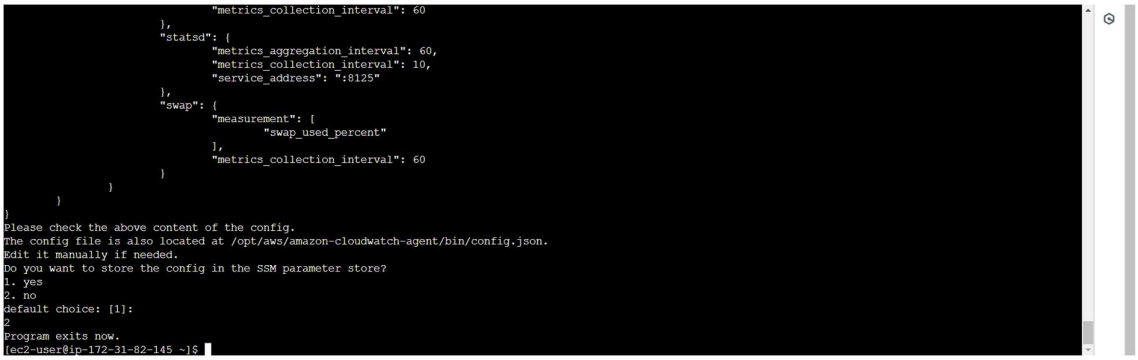






STEP 8: In path write: /var/log/messages

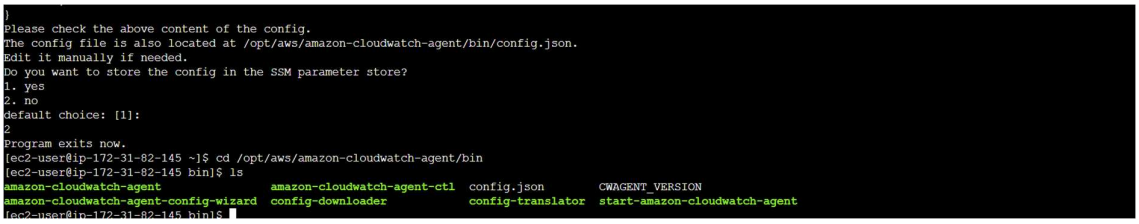




STEP 9: Now go to /opt/aws/amazon-cloudwatch-agent/bin folder using command:

- cd /opt/aws/amazon-cloudwatch-agent/bin

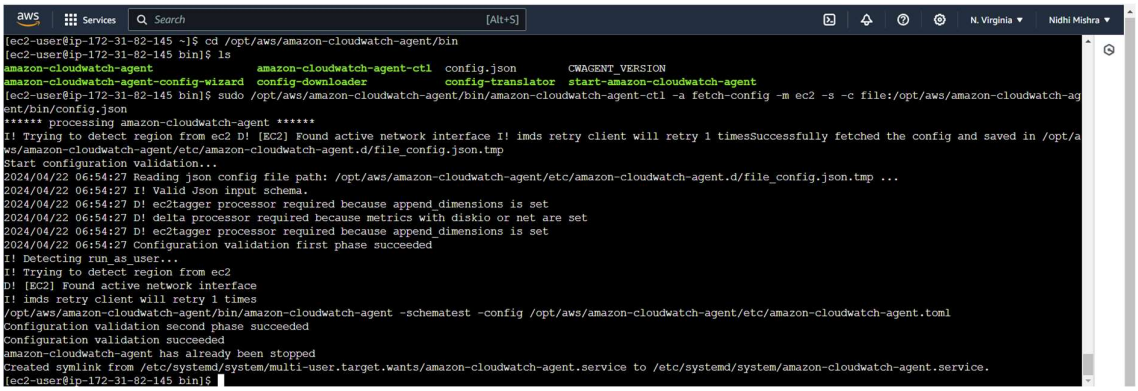
and list using: “ls”



STEP 10: Validate the config.json file using following command:

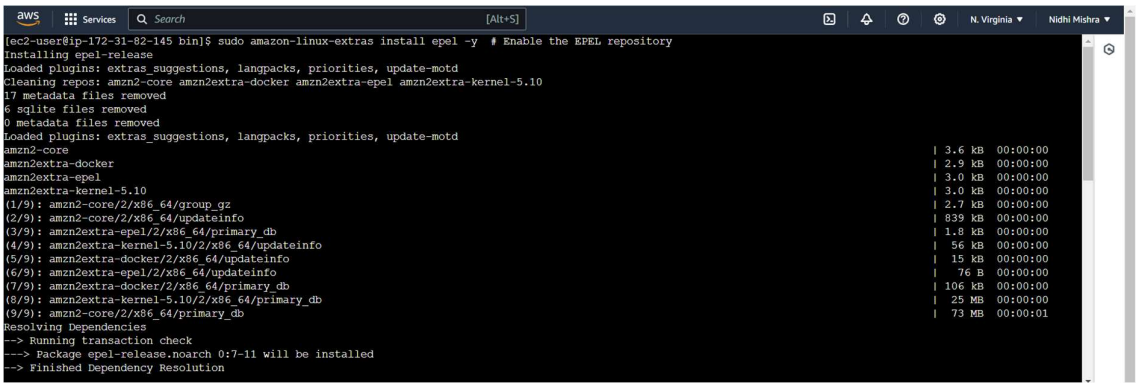
- sudo /opt/aws/amazon-cloudwatch-agent/bin/amazon-cloudwatch-agent-ctl -a fetch-config -m ec2 -s -c file:configuration-file-path

For Example, here path is /opt/aws/amazon-cloudwatch-agent/bin/config.json



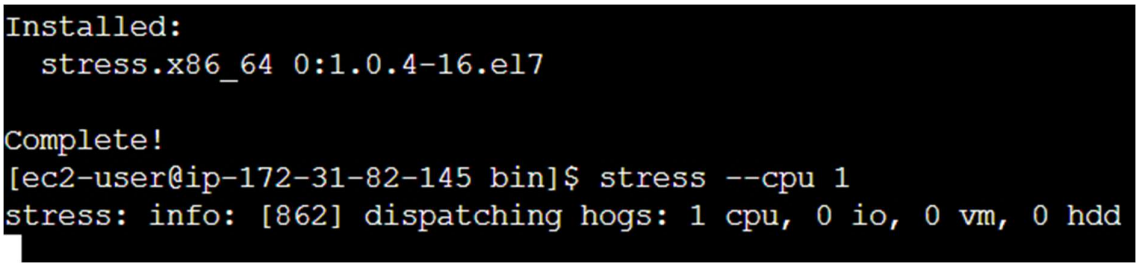
STEP 11: Now install the stress package using following command and stress it to test the cpu utilization:

- sudo amazon-linux-extras install epel -y # Enable the EPEL repository
- sudo yum install stress -y # Install the stress package

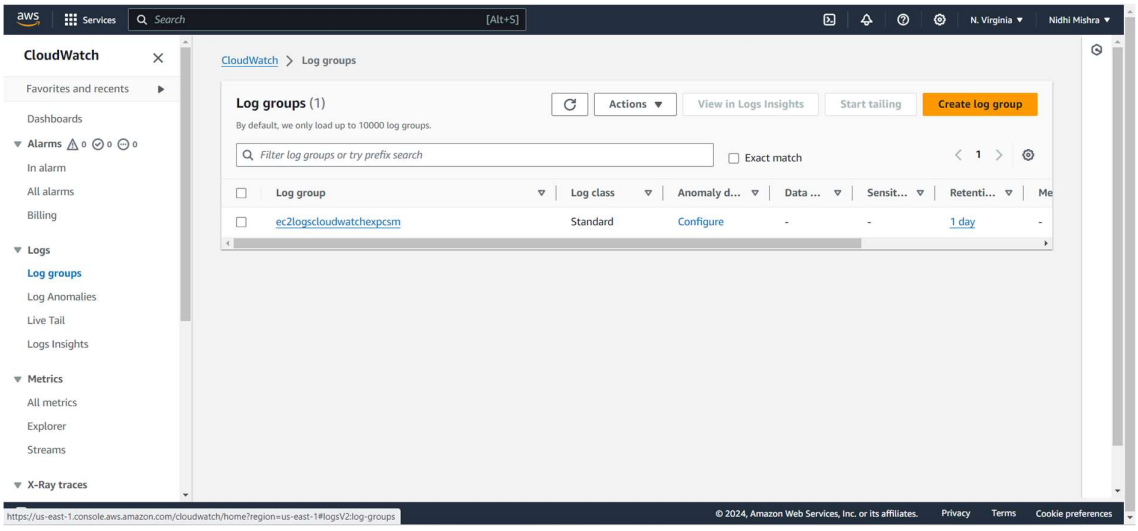


STEP 12: Use stress command:

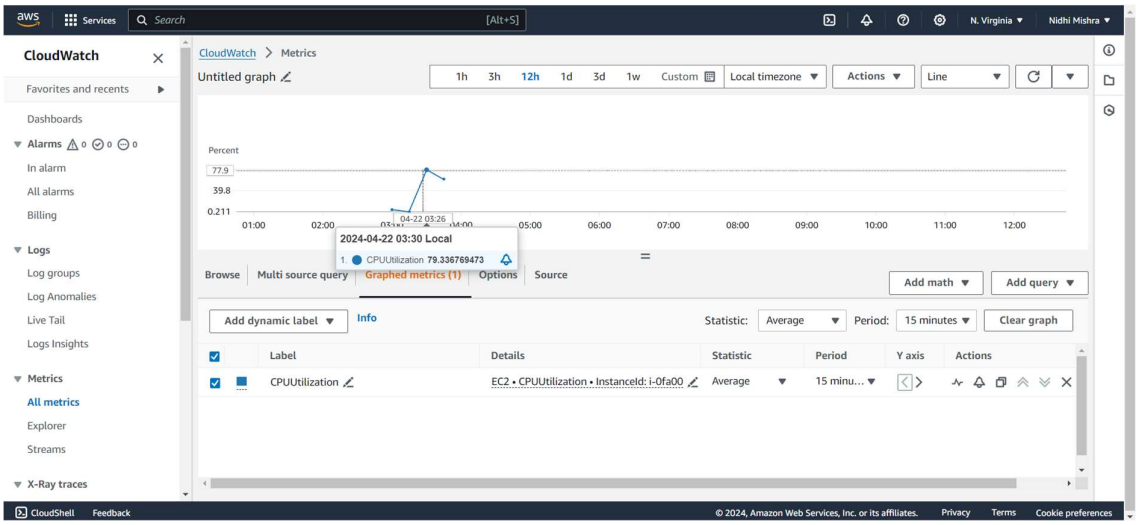
- stress –cpu 1



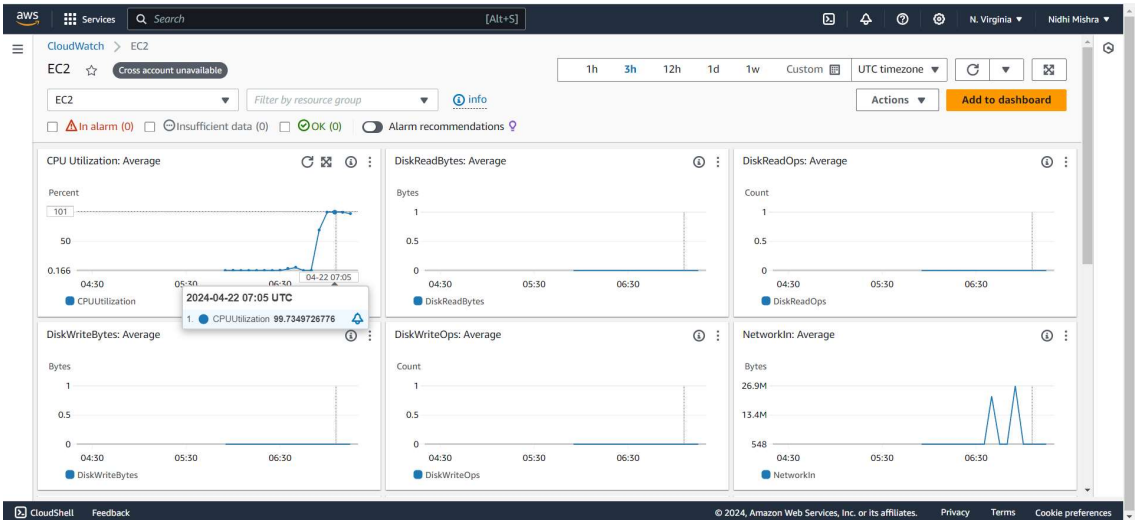
STEP 13: Now go to CloudWatch and go to Log Groups tab and see whether log files are created or not.



STEP 14: Go to CloudWatch, Matrices Tab and in EC2 tab select to get matrices:



STEP 15: You can check the graph of all EC2 resources.



QUESTIONS:

1. **What is CloudWatch and how does it integrate with EC2 instances?**

CloudWatch is a monitoring service provided by AWS that allows you to collect and track metrics, logs, and events from various AWS resources in real-time. It provides insights into the operational health of your AWS infrastructure and applications. CloudWatch integrates seamlessly with EC2 instances by automatically collecting and monitoring key performance metrics such as CPU utilization, disk I/O, and network traffic. This integration enables you to monitor the performance and health of your EC2 instances and take proactive measures to optimize their usage and performance.

2. **Explain the key metrics monitored by CloudWatch for EC2 instances:**

CloudWatch monitors a range of metrics for EC2 instances, including:

- CPU utilization: Percentage of CPU capacity in use.
- Disk metrics: Read/write operations and throughput on instance storage volumes.
- Network metrics: Incoming and outgoing network traffic.
- Status checks: Checks on the instance's system status and instance status. These metrics provide valuable insights into the performance and health of EC2 instances, helping you to optimize resource utilization, identify performance bottlenecks, and troubleshoot issues.

3. **How do you enable detailed monitoring for EC2 instances in CloudWatch and what benefits does it provide?**

Detailed monitoring for EC2 instances provides metrics at a granularity of one-minute intervals, compared to the default five-minute intervals of basic monitoring. You can enable detailed monitoring during instance launch or modify it later through the AWS Management Console, CLI, or API. Detailed monitoring offers more granular visibility into the performance of your EC2 instances, especially for workloads with rapid scaling or short-lived bursts of activity. This enables you to capture and analyze performance trends more accurately, making it easier to optimize resource allocation and respond to dynamic workload demands.

4. **Explain the process of setting up CloudWatch alarms for EC2 instances, and what actions can be triggered by these alarms:**

To set up CloudWatch alarms for EC2 instances, you first define the metric you want to monitor (e.g., CPU utilization), set the threshold value, and configure the alarm action. Alarm actions can include sending notifications via Amazon SNS (Simple Notification Service), triggering Auto Scaling actions to automatically scale your EC2 fleet based on demand, executing an AWS Lambda function, or stopping/terminating instances. CloudWatch alarms help you proactively monitor the health and performance of your EC2 instances and take automated actions based on predefined thresholds.

5. **Outline the steps to configure CloudWatch logs for EC2 instances:**

Configuring CloudWatch logs for EC2 instances involves several steps:

- Install the CloudWatch Logs agent on the EC2 instance.
- Configure the agent to monitor log files and send log data to CloudWatch Logs.
- Create log groups and log streams in CloudWatch Logs to organize and store the log data.
- Optionally, configure log retention settings and access controls for the log data. By configuring CloudWatch logs, you can centralize and analyze log data from multiple EC2 instances in a single location, enabling you to monitor application and system-level logs, troubleshoot issues, and comply with regulatory requirements.

6. **Difference between basic monitoring and detailed monitoring in CloudWatch:**

Basic monitoring provides metrics at five-minute intervals and is enabled by default for all AWS resources, including EC2 instances. It provides a cost-effective way to monitor resource utilization and performance. Detailed monitoring, on the other hand, provides metrics at one-minute intervals and offers more granular visibility into resource utilization, especially for workloads with rapid scaling or short-lived bursts of activity. Detailed monitoring incurs additional charges compared to basic monitoring but provides more accurate insights for real-time monitoring and analysis.

7. **What is the CloudWatch agent, and how does it enhance monitoring capabilities for EC2 instances?**

The CloudWatch agent is a lightweight software component that runs on EC2 instances and enables you to collect system-level metrics, logs, and custom metrics. It enhances monitoring capabilities by providing additional insights into the performance and health of EC2 instances beyond what's available by default in CloudWatch. The agent can collect metrics such as memory usage, disk space, and custom application metrics, as well as monitor log files and send log data to CloudWatch Logs. This enables you to gain deeper visibility into your EC2 instances and applications, troubleshoot issues more effectively, and optimize resource utilization.

8. **How do you use CloudWatch dashboards to visualize metrics?**

CloudWatch dashboards allow you to create custom dashboards to visualize metrics from various AWS resources, including EC2 instances. You can add widgets to the dashboard, such as line charts, bar charts, or numerical displays, to visualize specific metrics and performance trends. You can customize the layout and design of the dashboard to suit your monitoring needs, such as grouping related metrics, configuring refresh intervals, and defining dashboard permissions. CloudWatch dashboards provide a centralized and customizable way to monitor the health and performance of your AWS infrastructure and applications in real-time.

9. **What is AWS Lambda function logs from CloudWatch to S3?**

AWS Lambda function logs from CloudWatch to S3 involves configuring CloudWatch Logs to export log data from Lambda functions to an Amazon S3 bucket. This allows you to archive and retain Lambda function logs for long-term storage, analysis, and compliance purposes. You can configure log exports using the AWS Management Console, CLI, or SDKs, specifying the log group, S3 bucket, and optional prefix for storing the log data. Once configured, CloudWatch Logs automatically exports log data to the specified S3 bucket, making it available for further analysis and archival outside of CloudWatch.

10. **What is the role of IAM in configuring log exports?**

IAM (Identity and Access Management) plays a crucial role in configuring log exports by defining the permissions required for CloudWatch to access and export log data to S3 buckets. You need to create IAM policies that grant permissions for CloudWatch to read log data from CloudWatch Logs and write log data to S3 buckets. These policies specify the actions allowed (e.g., **logs:CreateExportTask**, **s3:PutObject**) and the resources (e.g., log groups, S3 buckets) that CloudWatch can access. You then attach these policies to IAM roles used by CloudWatch, ensuring that the service has the necessary permissions to export log data securely and reliably.