AWS Certification : How to setup Cloudwatch Logs for AWS services?

AWS Cloudwatch is a monitoring mechanism provided by Amazon. It has the ability to monitor services as mentioned below

Amazon EC2 instances

Amazon RDS

Amazon Dynamo DB

And a host of other custom metrics

Since [Cloudwatch](https://aws.amazon.com/cloudwatch/) fulfills a key security compliance requirement from a monitoring perspective, this is an important security aspect provided by AWS.

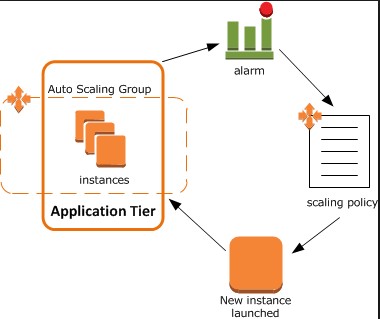
**What is AWS CloudWatch?**

So as mentioned in the Exam objective, **AWS Cloudwatch is a monitoring mechanism provided by Amazon** . It helps in a lot of aspects, with one of the most common one shown below. The below diagram from AWS documentation shows how Auto scaling is combined with Cloudwatch.

So let’s look at each part in a little bit more detail

The Auto Scaling Group consists of EC2 instances which can increase or decrease in number depending on the load which is required.

Now one way to trigger the auto scaling group is via a Cloud watch alarm. So let’s say an alarm is created which says that if the EC2 instances in the group reach an upper threshold of 80% in CPU time, then we need to scale or increase the capacity of the EC2 instances to take the additional load. So the 80% CPU usage is a Cloudwatch metric. When this metric is reached an alarm is triggered by Cloudwatch to the Auto scaling group to increase the number of the EC2 instances in the Auto scaling group.

[](https://www.whizlabs.com/wp-content/uploads/2016/12/AWS-Article3-1.jpg)

**What are AWS CloudWatch logs?**

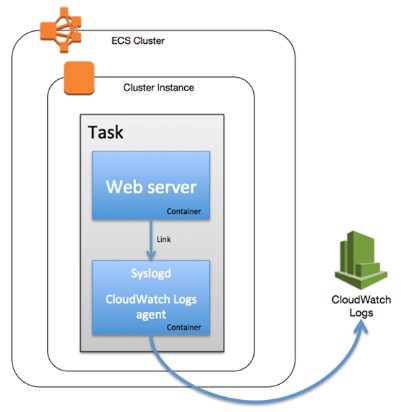
AWS Cloudwatch logs can be used to take logs from EC2 instances and process them accordingly. You can use Cloudwatch logs to achieve the following

Monitor and store the logs for your operating system to understand better how your application is performing

It can be used to track the number of errors in your application by processing the logs for error messages.

It can also send notification if the number of error messages goes beyond a threshold value.

The below diagram from AWS documentation shows how an example of Cloudwatch Logs. So let’s look at each part in a little bit more detail

[](https://www.whizlabs.com/wp-content/uploads/2016/12/AWS-Article3-2.jpg)

We have a web server inside a cluster. This web server is hosted inside a container.

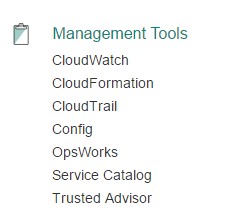
We then use custom scripts which are present for Linux which creates custom logs. These custom logs are sent via an agent to the Cloudwatch logs service.

The cloud watch logs can process these sent logs from the web server and see any patterns for any errors or any other sort of analysis which is required.

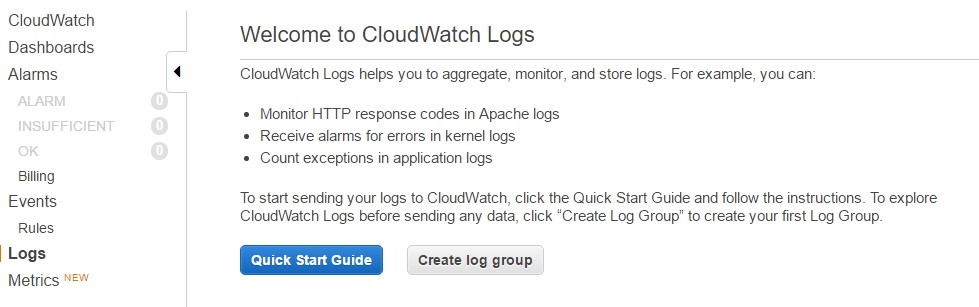
**A closer look at AWS CloudWatch**

Now let’s have a closer look at AWS Cloudwatch.

**Step 1:** Log into your AWS Console. You will see Cloud Watch under Management Tools.

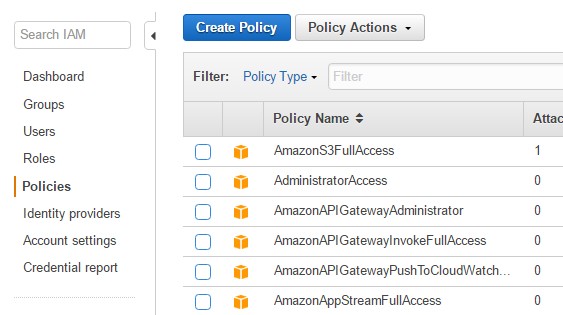
[](https://www.whizlabs.com/wp-content/uploads/2016/12/AWS-Article3-3.jpg)

**Step 2:** You can then click on Logs to get started with Cloudwatch logs.

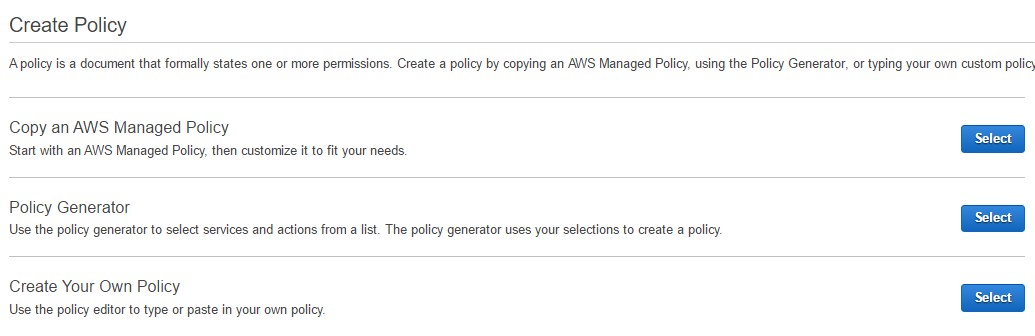
[](https://www.whizlabs.com/wp-content/uploads/2016/12/AWS-Article3-4.jpg)

**Step 3:** So in order to work with CloudWatch logs, we will need to create an Ec2 instance which will send logs to Cloudwatch. The first step is to create a policy which will be used to allow the EC2 instance to work with Cloudwatch.

Go to Identity and Access Management, click on Policies and click on Create Policy

[](https://www.whizlabs.com/wp-content/uploads/2016/12/AWS-Article3-5.jpg)

**Step 4:** In the screen, choose the option of “Create Your Own Policy”

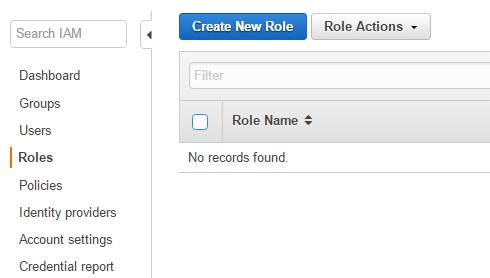
[](https://www.whizlabs.com/wp-content/uploads/2016/12/AWS-Article3-6.jpg)

**Step 5:** Now give the Policy a name and add the following JSON to the policy document

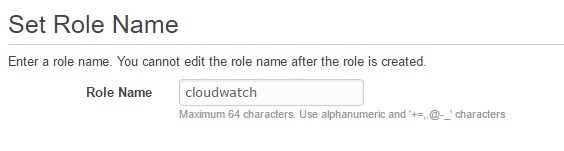
{ "Version":"2012-10-17", "Statement":[ { "Effect":"Allow", "Action":[ "logs:CreateLogGroup", "logs:CreateLogStream", "logs:PutLogEvents", "logs:DescribeLogStreams" ], "Resource":[ "arn:aws:logs:\*:\*:\*" ] } ] }

[](https://www.whizlabs.com/wp-content/uploads/2016/12/AWS-Article3-7.jpg)

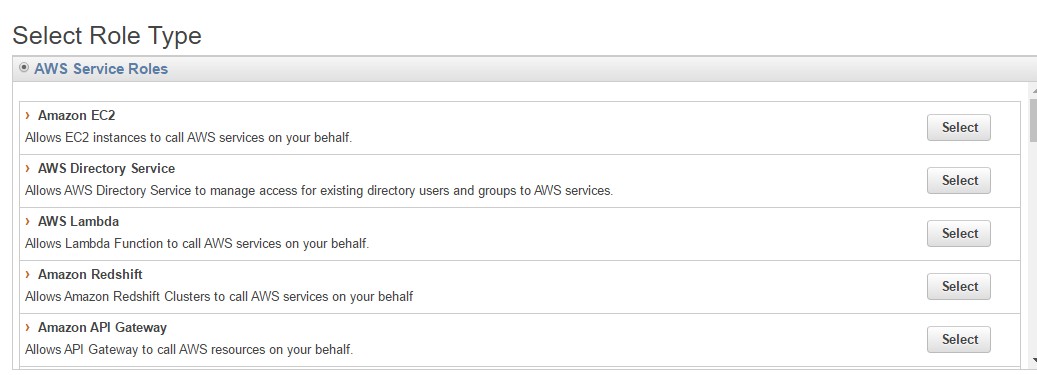
**Step 6:** Once the Policy has been created, it’s now time to create the Role. Go to the Roles section and select the create new role

[](https://www.whizlabs.com/wp-content/uploads/2016/12/AWS-Article3-8.jpg)

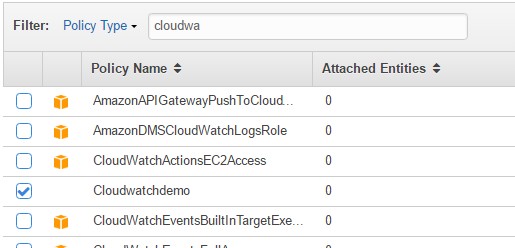
**Step 7:** Now give the role a name

[](https://www.whizlabs.com/wp-content/uploads/2016/12/AWS-Article3-9.jpg)

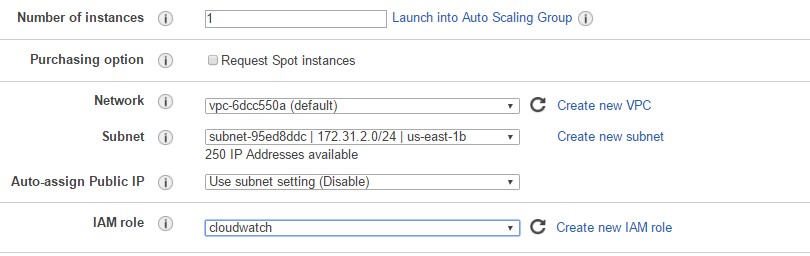
**Step 8:** Next select the role type as Amazon EC2

[](https://www.whizlabs.com/wp-content/uploads/2016/12/AWS-Article3-10.jpg)

**Step 9:** Search for the Cloudwatch policy which was created earlier

[](https://www.whizlabs.com/wp-content/uploads/2016/12/AWS-Article3-11.jpg)

**Step 10:** Now that the role has been created. Create your EC2 instance, when the instance is created, make sure that the IAM role chosen is the one which is created above

[](https://www.whizlabs.com/wp-content/uploads/2016/12/AWS-Article3-12.jpg)

**Step 11:** Now follow the steps in the [this link](http://docs.aws.amazon.com/AmazonCloudWatch/latest/logs/QuickStartEC2Instance.html) for the type of EC2 instance you spin up.

An example set of steps is given below for an Ubuntu instance

Log into the Ec2 instance

Do an apt-get update to download the latest packages on the system

Do an apt-get install –y python to install the python software

Run the command curl https://s3.amazonaws.com/aws-cloudwatch/downloads/latest/awslogs-agent-setup.py -O

Run the command to setup the awslogs – sudo python ./awslogs-agent-setup.py –region us-east-1

You will be asked for the following details during the configuration

The below is only required if the role is not configured for the EC2 instance

The AWS access key ID.

The AWS secret access key

Default region name – The default is us-east-1. You can set this to us-east-1, us-west-1, us-west-2, ap-south-1, ap-northeast-2, ap-southeast-1, ap-southeast-2, ap-northeast-1, eu-central-1, eu-west-1, or sa-east-1.

The default output format

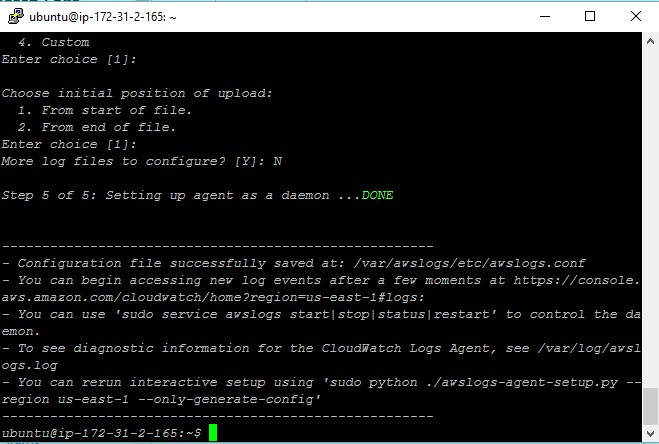
Path of log file to upload – The location of the file that contains the log data you want to send.

Destination Log Group name – The name for your log group.

Destination Log Stream name – By default, this is the name of the host.

Timestamp format – Specify the format of the timestamp within the specified log file.

Initial position – How data is uploaded. Set this to start\_of\_file to upload everything in the data file. Set to end\_of\_file to upload only newly-appended data.

[](https://www.whizlabs.com/wp-content/uploads/2016/12/AWS-Article3-13.jpg)

Once configured all log from the server will then be configured into Cloudwatch logs

**AWS Certification Points about CloudWatch Logs**

The above explanation is good enough to understand the concepts about cloudwatch and how to configure a cloudwatch for EC2 instance. You have to remember the following important points to answer the questions in AWS certification exam.

You can have an indefinite retention period for your logs. This enables one to go through the logs at a future date.

The Cloudwatch agent on the EC2 instance can used for log rotation so that logs can be moved off the host and on to a logging service. The raw data in the log files can then be accessed accordingly.

**Currently the Cloudwatch log agent is supported on Amazon Linux, Ubuntu, CentOS, Red Hat Enterprise Linux, and Windows.**

Summary

We have learnt about the cloudwatch in AWS and how to configure for EC2 instances with logs. There are many other features that are useful for managing your business. If you have any questions about the cloudwatch logs, please write it in the comments section.