

Install a CloudWatch Agent on an Existing EC2 Instance

You are working as a SysOps Engineer in the IT Software Industry. The R&D team has asked you to create a CloudWatch Dashboard and configure various business use-cases for an existing EC2 production instance so that the Client Success Team can monitor an EC2 instance for performance purposes.

In this lab, you will create metrics at Amazon CloudWatch and install a CloudWatch Agent on an existing EC2 instance to start collecting Memory and Network utilizations. You will also create an alarm when CPU goes up above 50% and send an automatic notification email to your email address.

1. Log in to the AWS web console with the lab credentials provided. First, ensure that the region drop-down in the top right is set to **US West (Oregon)**, then type **EC2** in the search box and click **Enter**.
2. On the left-hand **Instances** column menu, click on the **Instances** option.
3. Choose an existing EC2 instance named **CloudWatchEC2** and wait until Instance State is **Running**.
4. Click on **Connect** button at the top of the page.
5. Choose the **Session Manager** tab and click on **Connect** button at the bottom right of the page.

6. You'll be redirected to an EC2 Linux console window in a different browser tab. In EC2 console, download the CloudWatch Agent using below command :

```
sudo wget https://s3.amazonaws.com/amazoncloudwatch-
agent/amazon_linux/amd64/latest/amazon-cloudwatch-agent.rpm
```

Note: Above instruction is for Amazon Linux OS. Find the right agent link for your OS by visiting a [link](#).

7. To install the CloudWatch Agent, type below command on EC2 console :

```
sudo rpm -U ./amazon-cloudwatch-agent.rpm
```

8. Run below command to create a CloudWatch Agent configuration file for EC2 instance.

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

Note: Before running the CloudWatch Agent on any server, you must create a CloudWatch Agent configuration JSON file that specifies the metrics and logs which agent needs to collect.

9. In the AWS CloudWatch Agent Configuration Manager, answer the question as following :

- On which OS are you planning to use the agent? – **1** (linux)
- Are you using EC2 or On-Premises hosts? – **1** (EC2)
- Which user are you planning to run the agent? – **2** (cwagent)
- Do you want to turn on StatsD daemon? – **2** (no)
- Do you want to monitor metrics from CollectD? – **2** (no)
- Do you want to monitor any host metrics? E.g CPU, memory, etc – **1** (yes)
- Do you want to monitor cpu metrics per core? – **1** (yes)
- Do you want to add ec2 dimensions (ImageId, InstanceId, InstanceType, AutoScalingGroupName) into all of your metrics if the info is available? – **1** (yes)
- Do you want to aggregate ec2 dimensions (InstanceId)? – **1** (yes)
- Would you like to collect your metrics at high resolution (sub-minute resolution)?
– **2** (10s)
- Which default metrics config do you want? – **2** (Standard)
- Are you satisfied with the above config? – **1** (yes)
- Do you have any existing CloudWatch Log Agent (<http://docs.aws.amazon.com/AmazonCloudWatch/latest/logs/AgentReference.html>) configuration file to import for migration? – **2** (no)
- Do you want to monitor any log files? – **2** (no)
- Do you want to store the config in the SSM parameter store? – **2** (no)

10. Start the CloudWatch Agent by running below command :

```
sudo /opt/aws/amazon-cloudwatch-agent/bin/amazon-cloudwatch-agent-ctl -  
a fetch-config -m ec2 -c file:/opt/aws/amazon-cloudwatch-  
agent/bin/config.json -s
```

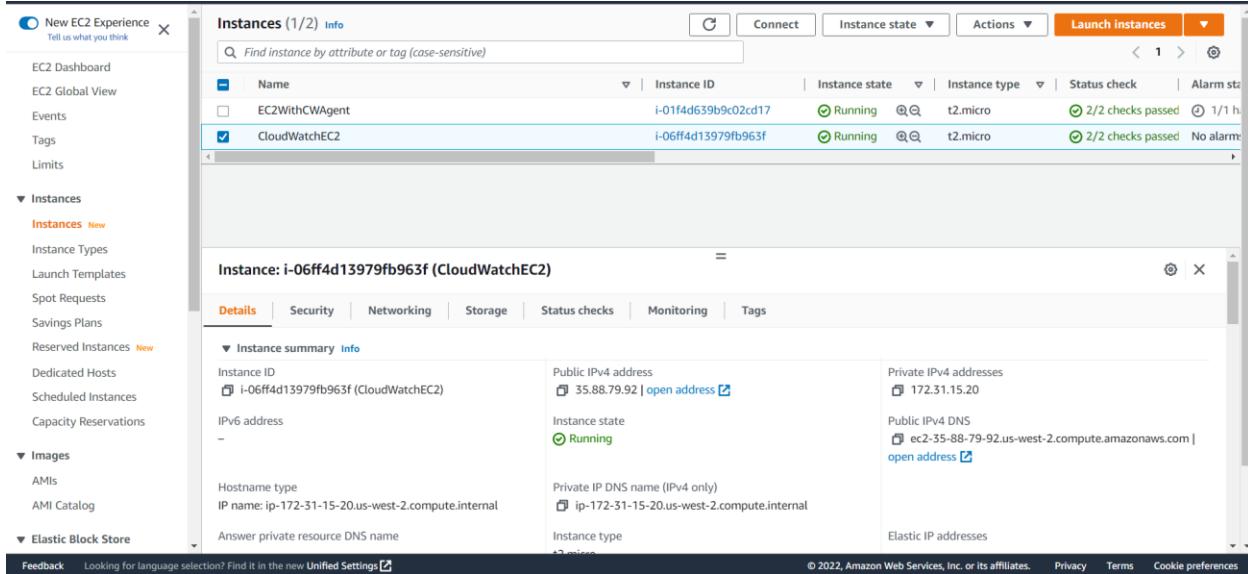
11. Validate CloudWatch Agent service is in running state or not using below command

:

```
sudo systemctl status amazon-cloudwatch-agent
```

```
[ec2-user@ip-172-31-30-199 ~]$ systemctl status amazon-cloudwatch-agent  
● amazon-cloudwatch-agent.service - Amazon CloudWatch Agent  
  Loaded: loaded (/etc/systemd/system/amazon-cloudwatch-agent.service; enabled; vendor preset: disabled)  
  Active: active (running) since Tue 2021-06-15 16:18:45 UTC; 12s ago  
    Main PID: 1457 (amazon-cloudwat)  
      CGroup: /system.slice/amazon-cloudwatch-agent.service  
             └─1457 /opt/aws/amazon-cloudwatch-agent/bin/amazon-cloudwatch-agent -config /opt/aws/amazon-clc  
  
Jun 15 16:18:45 ip-172-31-30-199.us-west-2.compute.internal systemd[1]: Started Amazon CloudWatch Agent.  
Jun 15 16:18:45 ip-172-31-30-199.us-west-2.compute.internal systemd[1]: Starting Amazon CloudWatch Agent..  
Jun 15 16:18:45 ip-172-31-30-199.us-west-2.compute.internal start-amazon-cloudwatch-agent[1457]: /opt/aws/  
Jun 15 16:18:45 ip-172-31-30-199.us-west-2.compute.internal start-amazon-cloudwatch-agent[1457]: Valid Jsc  
Jun 15 16:18:45 ip-172-31-30-199.us-west-2.compute.internal start-amazon-cloudwatch-agent[1457]: I! Detect  
Hint: Some lines were ellipsized, use -l to show in full.  
[ec2-user@ip-172-31-30-199 ~]$
```

Congratulations! You have successfully installed and configured the CloudWatch Agent on your EC2 instance. Now, you should be able to capture Memory and Network utilization of an EC2 instance in the CloudWatch.



The image shows two screenshots illustrating the AWS Session Manager connection process.

The top screenshot is a modal window titled "Connect to instance" with the sub-section "Info". It displays the message: "Connect to your instance i-06ff4d13979fb963f (CloudWatchEC2) using any of these options". Below this are four tabs: "EC2 Instance Connect" (disabled), "Session Manager" (selected, highlighted in orange), "SSH client", and "EC2 serial console". A section titled "Session Manager usage:" lists the following benefits:

- Connect to your instance without SSH keys or a bastion host.
- Sessions are secured using an AWS Key Management Service key.
- You can log session commands and details in an Amazon S3 bucket or CloudWatch Logs log group.
- Configure sessions on the Session Manager [Preferences](#) page.

At the bottom of the modal are "Cancel" and "Connect" buttons, with "Connect" being orange.

The bottom screenshot shows a browser window for "us-west-2.console.aws.amazon.com/systems-manager/session-manager/i-06ff4d13979fb963f?region=us-west-2#". The address bar includes "Feedback", "Looking for language selection? Find it in the new [Unified Settings](#)", and the URL. The page header includes "© 2022, Amazon Web Services, Inc. or its affiliates.", "Privacy", "Terms", and "Cookie preferences". The main content area is blacked out, and the "Terminate" button is visible in the top right corner of the browser window.

```
Session ID: pluralsight-1408ac03-0a5971921588e5914 Instance ID: i-06ff4d13979fb963f
sh-4.2$ sudo wget https://s3.amazonaws.com/amazon-cloudwatch-agent/amazon_linux/amd64/latest/amazon-cloudwatch-agent.rpm
--2022-11-24 05:46:55-- https://s3.amazonaws.com/amazoncloudwatch-agent/amazon_linux/amd64/latest/amazon-cloudwatch-agent.rpm
Resolving s3.amazonaws.com (s3.amazonaws.com) ... 52.216.49.192, 52.216.76.14, 52.217.33.22, ...
Connecting to s3.amazonaws.com (s3.amazonaws.com)|52.216.49.192|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 49701120 (47M) [application/octet-stream]
Saving to: 'amazon-cloudwatch-agent.rpm'

100%[=====] 49,701,120 8.68MB/s in 6.1s
2022-11-24 05:47:02 (7.83 MB/s) - 'amazon-cloudwatch-agent.rpm' saved [49701120/49701120]

sh-4.2$ sudo rpm -U ./amazon-cloudwatch-agent.rpm
create group cwagent, result: 0
create user cwagent, result: 0
create group aoc, result: 0
create user aoc, result: 0
sh-4.2$ sh-4.2$ sudo /opt/aws/amazon-cloudwatch-agent/bin/amazon-cloudwatch-agent-config-wizard
=====
= Welcome to the Amazon CloudWatch Agent Configuration Manager =
= =
= CloudWatch Agent allows you to collect metrics and logs from =
= your host and send them to CloudWatch. Additional CloudWatch =
= charges may apply. =
=====
On which OS are you planning to use the agent?
1. linux
2. windows
3. darwin
default choice: [1]:
1
Trying to fetch the default region based on ec2 metadata...
Are you using EC2 or On-Premises hosts?
1. EC2
2. On-Premises
default choice: [1]:
1
Session ID: pluralsight-1408ac03-0a5971921588e5914 Instance ID: i-06ff4d13979fb963f
default choice: [1]:
1
Which user are you planning to run the agent?
1. root
2. cwagent
3. others
default choice: [1]:
2
Do you want to turn on StatsD daemon?
1. yes
2. no
default choice: [1]:
2
Do you want to monitor metrics from CollectD? WARNING: CollectD must be installed or the Agent will fail to start
1. yes
2. no
default choice: [1]:
2
Do you want to monitor any host metrics? e.g. CPU, memory, etc.
1. yes
2. no
default choice: [1]:
1
Do you want to monitor cpu metrics per core?
1. yes
2. no
default choice: [1]:
1
Do you want to add ec2 dimensions (ImageId, InstanceId, InstanceType, AutoScalingGroupName) into all of your metrics if the info is available?
1. yes
2. no
default choice: [1]:
1
Do you want to aggregate ec2 dimensions (InstanceId)?
1. yes
2. no
default choice: [1]:
1
```

```
← → C us-west-2.console.aws.amazon.com/systems-manager/session-manager/i-06ff4d13979fb963f?region=us-west-2#
Session ID: pluralsight-1408ac03-0a5971921588e5914 Instance ID: i-06ff4d13979fb963f
Terminate

do you want to aggregate ec2 dimensions (InstanceId)?
1. yes
2. no
default choice: [1]:
1
Would you like to collect your metrics at high resolution (sub-minute resolution)? This enables sub-minute resolution for all metrics, but you can customize for specific metrics in the output json file.
1. 1s
2. 10s
3. 30s
4. 60s
default choice: [4]:
2
Which default metrics config do you want?
1. Basic
2. Standard
3. Advanced
4. None
default choice: [1]:
2
Current config as follows:
{
    "agent": {
        "metrics_collection_interval": 10,
        "run_as_user": "cwagent"
    },
    "metrics": {
        "aggregation_dimensions": [
            [
                "InstanceId"
            ]
        ],
        "append_dimensions": {
            "AutoScalingGroupName": "${aws:AutoScalingGroupName}",
            "ImageId": "${aws:ImageId}",
            "InstanceId": "${aws:InstanceId}",
            "InstanceType": "${aws:InstanceType}"
        }
    },
    "metrics_collected": {
        "cpu": {
            "measurement": [
                "cpu_usage_idle",
                "cpu_usage_iowait",
                "cpu_usage_user",
                "cpu_usage_system"
            ],
            "metrics_collection_interval": 10,
            "resources": [
                "*"
            ],
            "totalcpu": false
        },
        "disk": {
            "measurement": [
                "used_percent",
                "inodes_free"
            ],
            "metrics_collection_interval": 10,
            "resources": [
                "*"
            ]
        },
        "diskio": {
            "measurement": [
                "io_time"
            ],
            "metrics_collection_interval": 10,
            "resources": [
                "*"
            ]
        },
        "mem": {
            "measurement": [
                "mem_used_percent"
            ],
            "metrics_collection_interval": 10
        }
    }
}

← → C us-west-2.console.aws.amazon.com/systems-manager/session-manager/i-06ff4d13979fb963f?region=us-west-2#
Session ID: pluralsight-1408ac03-0a5971921588e5914 Instance ID: i-06ff4d13979fb963f
Terminate
```

Session ID: pluralsight-1408ac03-0a5971921588e3914 Instance ID: i-06ff4d13979fb963f

Terminate

```

"mem": {
    "measurement": [
        "mem_used_percent"
    ],
    "metrics_collection_interval": 10
},
"swap": {
    "measurement": [
        "swap_used_percent"
    ],
    "metrics_collection_interval": 10
}
}
}

Are you satisfied with the above config? Note: it can be manually customized after the wizard completes to add additional items.
1. yes
2. no
default choice: [1]:
1
Do you have any existing CloudWatch Log Agent (http://docs.aws.amazon.com/AmazonCloudWatch/latest/logs/AgentReference.html) configuration file to import for migration?
1. yes
2. no
default choice: [2]:
2
Do you want to monitor any log files?
1. yes
2. no
default choice: [1]:
2
Saved config file to /opt/aws/amazon-cloudwatch-agent/bin/config.json successfully.
Current config as follows:
{
    "agent": {
        "metrics_collection_interval": 10,
        "run_as_user": "cwagent"
    },
    "metrics": [
        {
            "metrics_collection_interval": 10,
            "run_as_user": "cwagent"
        },
        {
            "metrics": {
                "aggregation_dimensions": [
                    [
                        "InstanceId"
                    ]
                ],
                "append_dimensions": {
                    "AutoScalingGroupName": "${aws:AutoScalingGroupName}",
                    "ImageId": "${aws:ImageId}",
                    "InstanceId": "${aws:InstanceId}",
                    "InstanceType": "${aws:InstanceType}"
                },
                "metrics_collected": {
                    "cpu": {
                        "measurement": [
                            "cpu_usage_idle",
                            "cpu_usage_iowait",
                            "cpu_usage_user",
                            "cpu_usage_system"
                        ],
                        "metrics_collection_interval": 10,
                        "resources": [
                            "*"
                        ],
                        "totalcpu": false
                    },
                    "disk": {
                        "measurement": [
                            "used_percent",
                            "inodes_free"
                        ],
                        "metrics_collection_interval": 10,
                        "resources": [
                            "*"
                        ]
                    }
                }
            }
        }
    }
}

```

Session ID: pluralsight-1408ac03-0a5971921588e3914 Instance ID: i-06ff4d13979fb963f

Terminate

```

"mem": {
    "measurement": [
        "mem_used_percent"
    ],
    "metrics_collection_interval": 10
},
"swap": {
    "measurement": [
        "swap_used_percent"
    ],
    "metrics_collection_interval": 10
}
}
}

Are you satisfied with the above config? Note: it can be manually customized after the wizard completes to add additional items.
1. yes
2. no
default choice: [1]:
1
Do you have any existing CloudWatch Log Agent (http://docs.aws.amazon.com/AmazonCloudWatch/latest/logs/AgentReference.html) configuration file to import for migration?
1. yes
2. no
default choice: [2]:
2
Do you want to monitor any log files?
1. yes
2. no
default choice: [1]:
2
Saved config file to /opt/aws/amazon-cloudwatch-agent/bin/config.json successfully.
Current config as follows:
{
    "agent": {
        "metrics_collection_interval": 10,
        "run_as_user": "cwagent"
    },
    "metrics": {
        "aggregation_dimensions": [
            [
                "InstanceId"
            ]
        ],
        "append_dimensions": {
            "AutoScalingGroupName": "${aws:AutoScalingGroupName}",
            "ImageId": "${aws:ImageId}",
            "InstanceId": "${aws:InstanceId}",
            "InstanceType": "${aws:InstanceType}"
        },
        "metrics_collected": {
            "cpu": {
                "measurement": [
                    "cpu_usage_idle",
                    "cpu_usage_iowait",
                    "cpu_usage_user",
                    "cpu_usage_system"
                ],
                "metrics_collection_interval": 10,
                "resources": [
                    "*"
                ],
                "totalcpu": false
            },
            "disk": {
                "measurement": [
                    "used_percent",
                    "inodes_free"
                ],
                "metrics_collection_interval": 10,
                "resources": [
                    "*"
                ]
            }
        }
    }
}

```

```
← → C us-west-2.console.aws.amazon.com/systems-manager/session-manager/i-06ff4d13979fb963f?region=us-west-2#
Session ID: pluralsight-1408ac03-0a5971921588e3914 Instance ID: i-06ff4d13979fb963f
Terminate

        "totalcpu": false
    },
    "disk": {
        "measurement": [
            "used_percent",
            "inodes_free"
        ],
        "metrics_collection_interval": 10,
        "resources": [
            "*"
        ]
    },
    "diskio": {
        "measurement": [
            "io_time"
        ],
        "metrics_collection_interval": 10,
        "resources": [
            "*"
        ]
    },
    "mem": {
        "measurement": [
            "mem_used_percent"
        ],
        "metrics_collection_interval": 10
    },
    "swap": {
        "measurement": [
            "swap_used_percent"
        ],
        "metrics_collection_interval": 10
    }
}
}

Please check the above content of the config.
The config file is also located at /opt/aws/amazon-cloudwatch-agent/bin/config.json.

← → C us-west-2.console.aws.amazon.com/systems-manager/session-manager/i-06ff4d13979fb963f?region=us-west-2#
Session ID: pluralsight-1408ac03-0a5971921588e3914 Instance ID: i-06ff4d13979fb963f
Terminate

"resources": [
    "*"
],
"diskio": {
    "measurement": [
        "io_time"
    ],
    "metrics_collection_interval": 10,
    "resources": [
        "*"
    ]
},
"mem": {
    "measurement": [
        "mem_used_percent"
    ],
    "metrics_collection_interval": 10
},
"swap": {
    "measurement": [
        "swap_used_percent"
    ],
    "metrics_collection_interval": 10
}
}

Please check the above content of the config.
The config file is also located at /opt/aws/amazon-cloudwatch-agent/bin/config.json.
Edit it manually if needed.
Do you want to store the config in the SSM parameter store?
1. yes
2. no
default choice: [1]:
2
Program exits now.
sh-4.2$
```

```

← → C us-west-2.console.aws.amazon.com/systems-manager/session-manager/l-06ff4d13979fb963?region=us-west-2#
Session ID: pluralsight-1408ac03-0a5971921588e5914 Instance ID: i-06ff4d13979fb963f
Terminate

}

Please check the above content of the config.
The config file is also located at /opt/aws/amazon-cloudwatch-agent/bin/config.json.
Edit it manually if needed.
Do you want to store the config in the SSM parameter store?
1. yes
2. no
default choice: [1]:
2
Program exits now.
sh-4.2$ sudo /opt/aws/amazon-cloudwatch-agent/bin/amazon-cloudwatch-agent-ctl -a fetch-config -m ec2 -c file:/opt/aws/amazon-cloudwatch-agent/bin/config.json -s
***** processing amazon-cloudwatch-agent *****
/opt/aws/amazon-cloudwatch-agent/bin/config-downloader --output-dir /opt/aws/amazon-cloudwatch-agent/etc/amazon-cloudwatch-agent.d --download-source file:/opt/aws/amazon-c
loudwatch-agent/bin/config.json --mode ec2 --config /opt/aws/amazon-cloudwatch-agent/etc/common-config.toml --multi-config default
I! Trying to detect region from ec2
D! [EC2] Found active network interface
Successfully fetched the config and saved in /opt/aws/amazon-cloudwatch-agent/etc/amazon-cloudwatch-agent.d/file_config.json.tmp
Start configuration validation...
/opt/aws/amazon-cloudwatch-agent/bin/config-translator --input /opt/aws/amazon-cloudwatch-agent/etc/amazon-cloudwatch-agent.json --input-dir /opt/aws/amazon-cloudwatch-age
nt/etc/amazon-cloudwatch-agent.d --output /opt/aws/amazon-cloudwatch-agent/etc/amazon-cloudwatch-agent.toml --mode ec2 --config /opt/aws/amazon-cloudwatch-agent/etc/common
-config.toml --multi-config default
2022/11/24 05:52:09 Reading json config file path: /opt/aws/amazon-cloudwatch-agent/etc/amazon-cloudwatch-agent.d/file_config.json.tmp ...
2022/11/24 05:52:09 I! Valid Json input schema.
I! Detecting run_as user...
I! Trying to detect region from ec2
D! [EC2] Found active network interface
No csm configuration found.
No log configuration found.
Configuration validation first phase succeeded
/opt/aws/amazon-cloudwatch-agent/bin/amazon-cloudwatch-agent -schematest -config /opt/aws/amazon-cloudwatch-agent/etc/amazon-cloudwatch-agent.toml
Configuration validation second phase succeeded
Configuration validation succeeded
amazon-cloudwatch-agent has already been stopped
Created symlink from /etc/systemd/system/multi-user.target.wants/amazon-cloudwatch-agent.service to /etc/systemd/system/amazon-cloudwatch-agent.service.
Redirecting to /bin/systemctl restart amazon-cloudwatch-agent.service
sh-4.2$ 
sh-4.2$ 
sh-4.2$ sudo systemctl status amazon-cloudwatch-agent
● amazon-cloudwatch-agent.service - Amazon CloudWatch Agent
   Loaded: loaded (/etc/systemd/system/amazon-cloudwatch-agent.service; enabled; vendor preset: disabled)
     Active: active (running) since Thu 2022-11-24 05:52:09 UTC; 27s ago
       Main PID: 3449 (amazon-cloudwatch-agent)
          CGroup: /system.slice/amazon-cloudwatch-agent.service
                  └─3449 /opt/aws/amazon-cloudwatch-agent/bin/amazon-cloudwatch-agent -config /opt/aws/amazon-cloudwatch-agent/etc/amazon-cloudwatch-agent.toml -envconfig /opt...

Nov 24 05:52:09 ip-172-31-15-20.us-west-2.compute.internal systemd[1]: Started Amazon CloudWatch Agent.
Nov 24 05:52:09 ip-172-31-15-20.us-west-2.compute.internal start-amazon-cloudwatch-agent[3449]: /opt/aws/amazon-cloudwatch-agent/etc/amazon-cloudwatch-agent.json d...g it.
Nov 24 05:52:09 ip-172-31-15-20.us-west-2.compute.internal start-amazon-cloudwatch-agent[3449]: I! Detecting run_as user...
Hint: Some lines were ellipsized, use -l to show in full.

```

Configure Multiple Metrics with Alarms

Now that you've installed and configured CloudWatch Agent on your EC2 instance in the previous challenge, you have been asked to create and configure detailed metrics using the Alarms option and associate a new SNS topic with a valid email subscription.

1. In the AWS console, type **CloudWatch** in the search box and click **Enter**.
 2. On the left-hand **CloudWatch** column menu, click on the **Alarms** option.
 3. Click on the **Create alarm** button from the right side of the Alarms page.
 4. Click on the **Select metric** button and under the **Metrics** tab click on the **CWAgent** custom namespace.
 5. Click on **ImageID,InstanceId,InstanceType** and select a **CloudWatchEC2** instance with a Metric Name **mem_used_percent** from the list.
- Note:** It may take a few minutes to display available metrics for **CloudWatchEC2** instance. If metrics are still not displayed for your EC2 instance then try to reload the page and restart the process from step 4 again.
6. Click on the **Select metric** button at the bottom right of the page.

7. Provide field values under the Graph section as below:

- **Metric name** - Keep default value.
- **InstanceId** - Keep default value.
- **ImageId** - Keep default value.
- **InstanceType** - Keep default value.
- **Statistics** - Type and select **p50**.
- **Period** - Choose **10 seconds** from dropdown.

8. Provide field values under the Conditions section as below:

- **Threshold type** - keep **Static** as default selected.
- **Whenever mem_available_percent is...** - **Greater/Equal** (\geq threshold).
- **than....** - Provide the threshold value as **50**.

9. Leave the Additional configuration section untouched and click on the **Next** button on the bottom right of the page.

10. In the Configure actions page, click the **Add notification** button.

11. In the **Notification** section, keep **In alarm** option as selected.

12. Choose the **Create new topic** option in the Select an SNS topic section.

13. Provide the topic name as **EC2_CloudWatch_SNS_Topic** in the **Create a new topic...** field and provide a valid email address that you can use to receive the notification in the **Email endpoints that will receive the notification...** field.

14. Click on the **Create topic** button and wait for the process to finish.

15. Leave all other sections untouched and click on the **Next** button at the right bottom of the page.

16. Provide the **Alarm name** as **EC2-CloudWatch-Alarm**, leave the **Alarm description** field untouched and click on the **Next** button at the bottom of the page.

17. Review the alarm on the preview page and click on the **Create alarm** button at the right bottom of the page.

Congratulations! You have successfully created an Alarm for an EC2 instance in the CloudWatch. You can search for the same alarm on the list of the **Alarms** page by using the name provided in the Alarm Name field. Also, click on the **Name** hyperlink to see more details. You can also refresh the page until Alarm state is changed from **Insufficient data** to **OK**.

The screenshot shows the AWS CloudWatch Metrics Alarms interface. On the left, a sidebar menu includes 'Alarms' (with 1 pending confirmation), 'Logs', 'Metrics', 'X-Ray traces', 'Events', 'Application monitoring', and 'Insights'. Under 'Alarms', there is a 'Create alarm' button. The main area has two sections: 'Alarms by AWS service' (empty) and 'Recent alarms'. The 'Recent alarms' section shows a single entry for 'EC2-Alarm-Default' with a status graph. The graph shows three data points: 87.3 (mem_used_percent >= 50 for 1 datapoint within 10 seconds), 49.3, and 11.2. The status is 'OK' with a green circle icon. Below the graph is a detailed view of the alarm configuration:

Condition	Description
mem_used_percent >= 50 for 1 datapoint within 10 seconds	Actions enabled: Warning

At the bottom of the interface, a message says 'Some subscriptions are pending confirmation'.

CloudWatch > Alarms > Create alarm

Specify metric and conditions

Step 1 Specify metric and conditions

Step 2 Configure actions

Step 3 Add name and description

Step 4 Preview and create

Metric

Graph Preview of the metric or metric expression and the alarm threshold.

Select metric

Next

Feedback Looking for language selection? Find it in the new Unified Settings.

© 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Select metric

Untitled graph

mem_used_percent

1h 3h 12h 1d 3d 1w Custom Line

Browse Query Graphed metrics (1) Options Source Add math Add query Graph with SQL Graph search

Metrics (4)

All > CWAgent > ImageId, InstanceId, InstanceType

Instance name (4)	ImageId	InstanceId	InstanceType	Metric name
<input checked="" type="checkbox"/> CloudWatchEC2	ami-0fcfa5bf60c06651d	i-057a283a54ad40448	t2.micro	mem_used_percent
<input type="checkbox"/> CloudWatchEC2	ami-0fcfa5bf60c06651d	i-057a283a54ad40448	t2.micro	swap_used_percent
<input type="checkbox"/> EC2WithCWAgent	ami-0fcfa5bf60c06651d	i-06b8c06b9bf65da1	t2.micro	mem_used_percent
<input type="checkbox"/> EC2WithCWAgent	ami-0fcfa5bf60c06651d	i-06b8c06b9bf65da1	t2.micro	swap_used_percent

Select a single metric to continue Cancel

Feedback Looking for language selection? Find it in the new Unified Settings.

© 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

CloudWatch > Alarms > Create alarm

Step 1
Specify metric and conditions

Step 2
Configure actions

Step 3
Add name and description

Step 4
Preview and create

Specify metric and conditions

Metric

Graph
This alarm will trigger when the blue line goes above the red line for 1 datapoints within 5 minutes.

Percent
10
9.71
9.43
04:30 05:30 06:30
mem_used_percent

Namespace
CWAgent

Metric name
mem_used_percent

InstanceId
i-037a283a54ad40448

ImageId
ami-0fcaca5bf60c06651d

InstanceType
t2.micro

Instance name
CloudWatchEC2

Statistic
Q_Average

Period
5 minutes

Feedback Looking for language selection? Find it in the new [Unified Settings](#).

CloudWatch > Alarms > Create alarm

Step 1
Specify metric and conditions

Step 2
Configure actions

Step 3
Add name and description

Step 4
Preview and create

Specify metric and conditions

Metric

Graph
This alarm will trigger when the blue line goes above the red line for 1 datapoints within 10 seconds.

Percent
9.86
9.74
9.65
06:55 07:00 07:05
mem_used_percent

Namespace
CWAgent

Metric name
mem_used_percent

InstanceId
i-037a283a54ad40448

ImageId
ami-0fcaca5bf60c06651d

InstanceType
t2.micro

Instance name
CloudWatchEC2

Statistic
Q_p50

Period
10 seconds

Feedback Looking for language selection? Find it in the new [Unified Settings](#).

© 2022, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

Statistic: Q_p50

Period: 10 seconds

Conditions

Threshold type: Static (selected)

Whenever mem_used_percent is...:

than... Define the threshold value: 10000

Required
Must be a number

Next

Feedback Looking for language selection? Find it in the new Unified Settings.

CloudWatchLogs

Statistic: Q_p50

Period: 10 seconds

Conditions

Threshold type: Static (selected)

Whenever mem_used_percent is...:

than... Define the threshold value: 50

Must be a number

Next

CloudWatch > Alarms > Create alarm

Step 1 Specify metric and conditions

Step 2 Configure actions

Step 3 Add name and description

Step 4 Preview and create

Configure actions

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

Select an email list Only email lists for this account are available.

Add notification

Auto Scaling action

Add Auto Scaling action

EC2 action

Feedback Looking for language selection? Find it in the new Unified Settings.

© 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Select an existing SNS topic
 Create new topic
 Use topic ARN to notify other accounts

Send a notification to...

Only email lists for this account are available.

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.
Remove

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

Create a new topic...
The topic name must be unique.

SNS topic names can contain only alphanumeric characters, hyphens (-) and underscores (_).

Email endpoints that will receive the notification...
Add a comma-separated list of email addresses. Each address will be added as a subscription to the topic above.

user1@example.com, user2@example.com

Create topic
Add notification

Feedback Looking for language selection? Find it in the new Unified Settings. © 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Create new topic
 Use topic ARN to notify other accounts

Send a notification to...
 X
Only email lists for this account are available.

Email (endpoints)
mayankchaturvedi9991@gmail.com - View in SNS Console
Add notification

Auto Scaling action
Add Auto Scaling action

EC2 action
Add EC2 action

Systems Manager action [Info](#)
This action will create an Incident or OpItem in Systems Manager when the alarm is In alarm state.
Add Systems Manager action

Cancel Previous Next

Feedback Looking for language selection? Find it in the new Unified Settings. © 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

AWS Notification - Subscription Confirmation ▶ Inbox Updates 🖨️ 🔗



AWS Notifications <no-reply@sns.amazonaws.com>
to me ▾

12:41 PM (0 minutes ago)

☆ ↵ ⋮

You have chosen to subscribe to the topic:
arn:aws:sns:us-west-2:084044351869:EC2_CloudWatch_SNS_Topic

To confirm this subscription, click or visit the link below (If this was in error no action is necessary):
[Confirm subscription](#)

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](#)



The screenshot shows the "Add name and description" step of the CloudWatch alarm creation wizard. The left sidebar shows steps: Step 1 (Specify metric and conditions), Step 2 (Configure actions), Step 3 (Add name and description, which is active), and Step 4 (Preview and create). The main form has fields for "Name and description": "Alarm name" (set to "EC2-CloudWatch-Alarm") and "Alarm description - optional" (empty). A note says "Up to 1024 characters (0/1024)". At the bottom are "Cancel", "Previous", and "Next" buttons, with "Next" being orange.

CloudWatch > Alarms > Create alarm

Preview and create

Step 1: Specify metric and conditions

Metric

Graph
This alarm will trigger when the blue line goes above the red line for 1 datapoints within 10 seconds.

Percent
50

29.8

9.67
07:00 07:05 07:10
mem_used_percent

Namespace
CWAgent

Metric name
mem_used_percent

Instanceid
i-057a285a54ad40448

Imageid
ami-0fcac5bf60c06651d

Instance type
t2.micro

Instance name
CloudWatchEC2

Statistic
p50

Period
10 seconds

Conditions

Threshold type
Static

Whenever mem_used_percent is
Greater/Equal (\geq)
than...
50

► Additional configuration

Step 2: Configure actions

Actions

Notification
When In alarm, send a notification to "EC2_CloudWatch_SNS_Topic"

Notification
When In alarm, send a notification to "EC2_CloudWatch_SNS_Topic"

Step 3: Add name and description

Name and description

Name
EC2-CloudWatch-Alarm

Description
~

Feedback Looking for language selection? Find it in the new Unified Settings.

© 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

CloudWatch

Favorites and recent dashboards

Alarms (0)

All alarms

Logs

Metrics

X-Ray traces

Events

Application monitoring

Insights

Settings

Getting Started

Some subscriptions are pending confirmation
Amazon SNS doesn't send messages to an endpoint until the subscription is confirmed

Successfully created alarm EC2-CloudWatch-Alarm.

View SNS Subscriptions

CloudWatch > Alarms

Alarms (2)

Name	State	Last state update	Conditions	Actions
EC2-CloudWatch-Alarm	Insufficient data	2022-11-24 12:46:18	mem_used_percent >= 50 for 1 datapoints within 10 seconds	Actions enabled Warning
EC2-Alarm-Default	OK	2022-11-24 12:28:45	mem_used_percent >= 50 for 1 datapoints within 10 seconds	Actions enabled Warning

Create alarm

Feedback Looking for language selection? Find it in the new Unified Settings

CloudWatch

Favorites and recent dashboards

Alarms (0)

In alarm

All alarms

Logs

Metrics

X-Ray traces

Events

Application monitoring

Insights

Settings

Getting Started

Some subscriptions are pending confirmation
Amazon SNS doesn't send messages to an endpoint until the subscription is confirmed

Successfully created alarm EC2-CloudWatch-Alarm.

View SNS Subscriptions

CloudWatch > Alarms

Alarms (2)

Name	State	Last state update	Conditions	Actions
EC2-CloudWatch-Alarm	OK	2022-11-24 12:47:09	mem_used_percent >= 50 for 1 datapoints within 10 seconds	Actions enabled Warning
EC2-Alarm-Default	OK	2022-11-24 12:28:45	mem_used_percent >= 50 for 1 datapoints within 10 seconds	Actions enabled Warning

Create alarm

Feedback Looking for language selection? Find it in the new Unified Settings

CloudWatch

Favorites and recent dashboards

Alarms (0)

In alarm

All alarms

Logs

Metrics

X-Ray traces

Events

Application monitoring

Insights

Settings

Getting Started

Some subscriptions are pending confirmation
Amazon SNS doesn't send messages to an endpoint until the subscription is confirmed

Successfully created alarm EC2-CloudWatch-Alarm.

View SNS Subscriptions

CloudWatch > Alarms

Alarms (2)

Name	State	Last state update	Conditions	Actions
EC2-CloudWatch-Alarm	OK	2022-11-24 12:47:09	mem_used_percent >= 50 for 1 datapoints within 10 seconds	Actions enabled Warning
EC2-Alarm-Default	OK	2022-11-24 12:28:45	mem_used_percent >= 50 for 1 datapoints within 10 seconds	Actions enabled Warning

Create alarm

Enable an Email Notification with the SNS

Now that you've created an Alarm and SNS Topic for an email notification in the previous challenge, you should confirm an email subscription in order to receive an alarm on your email address.

1. In the AWS web console type **SNS (Simple Notification Service)** and click **Enter**.

2. On the left-hand **Amazon SNS** column menu, click on the **Subscriptions** option.

Note: The SNS topic **EC2_CloudWatch_SNS_Topic** is listed here with the status as **Pending confirmation**.

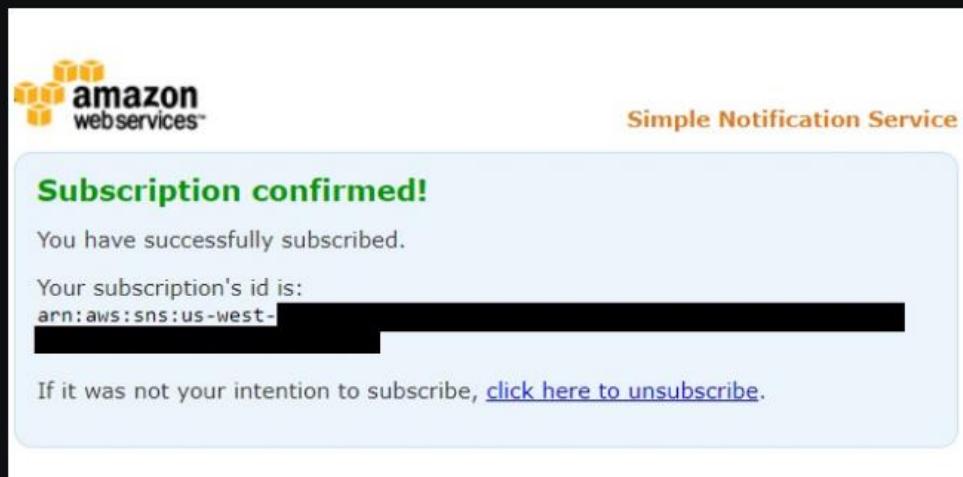
3. Open your email inbox to confirm the subscription:

- Check your email inbox and choose **Confirm subscription** in the email from Amazon SNS.



- Amazon SNS opens another tab in the web browser and displays a subscription confirmation with your subscription ID.

- Amazon SNS opens another tab in the web browser and displays a subscription confirmation with your subscription ID.



Note: Please **do not** click the *Unsubscribe* button on this email subscription confirmation page; doing so is unnecessary and can interfere with the lab. Once you're done with the lab you will be unsubscribed automatically and will receive no further email alerts.

- Now, navigate back to the AWS web console tab and refresh the Subscriptions page.
- In the Subscriptions list page, you will see the **Status** for **EC2_CloudWatch_SNS_Topic** is now **Confirmed**.

Congratulations! You have successfully subscribed to an email notification whenever an alarm is triggered. You can search for the same subscription on the list of the Subscriptions page by using the name provided in the **SNS Topic Name** field. Also, click on the **ID** hyperlink to see more details.

Amazon SNS

Dashboard

Topics Subscriptions

▼ Mobile Push notifications Text messaging (SMS) Origination numbers

Amazon SNS > Dashboard

Dashboard

Resources for us-west-2

Topics	Platform applications	Subscriptions
2	0	2

▼ Overview of Amazon SNS

Application-to-application (A2A)
Amazon SNS is a managed messaging service that lets you decouple publishers from subscribers. This is useful for application-to-application messaging for microservices, distributed systems, and serverless applications. [Learn more](#)

Application-to-person (A2P)
Amazon SNS lets you send push notifications to mobile apps, text messages to mobile phone numbers, and plain-text emails to email addresses. You can fan out messages with a topic, or publish to mobile devices.

Feedback Looking for language selection? Find it in the new [Unified Settings](#)

© 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Amazon SNS

Dashboard Topics **Subscriptions**

▼ Mobile Push notifications Text messaging (SMS) Origination numbers

Amazon SNS > Subscriptions

Subscriptions

Subscriptions (2)

ID	Endpoint	Status	Protocol	Topic
a4d07d6c-59f7-41f2-a1...	mayankchaturvedi9991...	Confirmed	EMAIL	EC2_CloudWatch_SNS_T...
Pending confirmation	admin@pluralsight.com	Pending confirmation	EMAIL	ec2-email-sns-topic-Email...

Feedback Looking for language selection? Find it in the new [Unified Settings](#)

© 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Amazon SNS X ⚠ **Important changes for sending text messages (SMS) to US destinations**
US mobile carriers have recently changed their regulations, and will require that all toll-free numbers (TFNs) complete a registration process with a regulatory body before September 30, 2022. If you currently have a toll-free number you must register your toll-free number by September 30, 2022 or you will no longer be able to use the toll-free number. [Learn more](#)

[View origination numbers](#) ⓘ

Dashboard
Topics
Subscriptions
▼ Mobile
Push notifications
Text messaging (SMS)
Origination numbers

Amazon SNS > Topics > [EC2_CloudWatch_SNS_Topic](#) > Subscription: a4d07d6c-59f7-41f2-a1de-649221e17c1d

Subscription: a4d07d6c-59f7-41f2-a1de-649221e17c1d [Edit](#) [Delete](#)

Details	
ARN	Status
arn:aws:sns:us-west-2:084044351869:EC2_CloudWatch_SNS_Topic:a4d07d6c-59f7-41f2-a1de-649221e17c1d	Confirmed
Endpoint	Protocol
mayankchaturvedi9991@gmail.com	EMAIL
Topic	Subscription Principal
EC2_CloudWatch_SNS_Topic	arn:aws:iam::084044351869:user/pluralsight-d45faaac

[Subscription filter policy](#) | [Redrive policy \(dead-letter queue\)](#)

[Feedback](#) Looking for language selection? Find it in the new [Unified Settings](#) ⓘ © 2022, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

Display Metrics in a Dashboard

Now that you have set up and configured CloudWatch Agent on your EC2 instance, an Alarm with a threshold above 50% CPU usage, you need to create a CloudWatch Dashboard and validate whether the metrics graphs are functioning properly or not.

1. In the AWS web console type **CloudWatch** in the search box and click **Enter**.
2. On the left-hand **CloudWatch** column menu, click on the **Dashboards** option.
Note: If you have not done so already, make sure you click “**Try out the new interface**” to see the updated CloudWatch interface.
3. Click on the **Create dashboard** button.
4. Provide a Dashboard name as **EC2-Cloudwatch-Dashboard** and click on the **Create dashboard** button.
5. In the Add widget section and choose the **Line** widget and then choose **Metrics** from the **Add to this dashboard** popup window.
6. In the **Add metric graph** popup window, click on the **CWAgent** custom namespace under the Metrics section.
7. Click on the **ImageId**, **Instanceld**, **InstanceType** and select a **CloudWatchEC2** instance with a **Metric Name as mem_used_percent** from the list.
8. Click on the **View graphed metrics** button and choose the **30 Seconds** value for **Period** field dropdown.
9. Click on the **Create widget** button on the right bottom of the page.
10. Click on the **Add widget** button again on the right top of the page.

11. Choose **Line** widget and then choose **Metrics** from **Add to this dashboard** popup window.
12. In the **Add metric graph** popup window, click on the **CWAgent** custom namespace under the Metrics section.
13. Click on the **ImageId**, **InstanceId**, **InstanceType**, **cpu** and select a **CloudWatchEC2** instance with a **Metric Name** as **cpu_usage_user** from the list.
14. Click on the **View graphed metrics** button and choose the **30 Seconds** value for **Period** field dropdown.
15. Click on the **Create widget** button on the right bottom of the page.
16. Click on the **Save dashboard** button on the right top of the page.
17. Navigate back to an EC2 Linux console tab opened previously. In EC2 console, type below command to create a CPU spike:

```
sudo stress --cpu 8 --vm-bytes $(awk '/MemAvailable/[printf "%d\n", $2 * 0.9;}' < /proc/meminfo)k --vm-keep -m 1 --timeout 60
```

Note: If **CloudWatchEC2** instance's Linux console tab has timed out or the session has expired then try to reload the UI page or reconnect using **Session Manager** option from Connect to instance page or you can repeat steps **1-5** provided in the first challenge.

18. Now, navigate to the AWS Console opened in another tab and click on the **reload icon** to refresh the dashboard.

Congratulations! You have successfully created the dashboard. It shows the line chart for CPU and Memory usage collected by a Cloudwatch Agent that is installed on the EC2 instance.

By refreshing the AWS Console you can see the alarm state moved to **In Alarm** state as CPU goes above 50%. Open your email inbox and you can see email by **AWS Notifications** whenever an alarm triggers. It may take a few minutes to receive an email. You can add an alarm into the dashboard and also drag widgets to beautify it.

CloudWatch

Favorites and recents

Dashboards

Alarms ▲ 0 ○ 2 ⊖ 0

In alarm

All alarms

Logs

Metrics

X-Ray traces

Events

Application monitoring

Insights

Settings

Getting Started

We've redesigned the custom dashboards to make them easier to use. Let us know what you think. Or you can use the old console.

CloudWatch > Dashboards

Custom dashboards | Automatic dashboards

Custom Dashboards (1) Info

Share dashboard Delete Create dashboard

Filter dashboards

Name	Sharing	Favorite	Last update (UTC)
EC2-Dashboard-Default		☆	2022-11-24 06:53

Feedback Looking for language selection? Find it in the new [Unified Settings](#).

© 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Create new dashboard

Dashboard name

EC2-Cloudwatch-Dashboard

Valid characters in dashboard names include "0-9A-Za-z-_".

Last update (U)

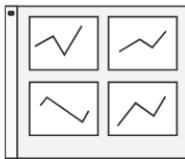
2022-11-24 06:

Cancel Create dashboard

Add widget

Select a widget type to add to the dashboard.

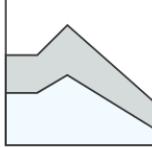
Explorer
A single widget with multiple tag-based graphs



Line
Compare metrics over time



Stacked area
Compare the total over time



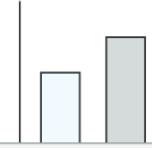
Number
Instantly see the latest value and trend for a metric



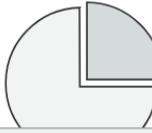
Gauge
See the latest value of a metric within a lower and upper range



Bar
Compare categories of data



Pie
Show percentage or proportional data



Custom widget - New
Code widgets using Lambda and more



Add to this dashboard

From which data source would you like to create the widget?

Metrics
Create widget based on Metrics and configure your widget on the next step.

Logs
Create widget based on query results from CloudWatch Logs Insights.

Add metric graph

Untitled graph 

1h 3h 12h 1d 3d 1w Custom 
   

Percent



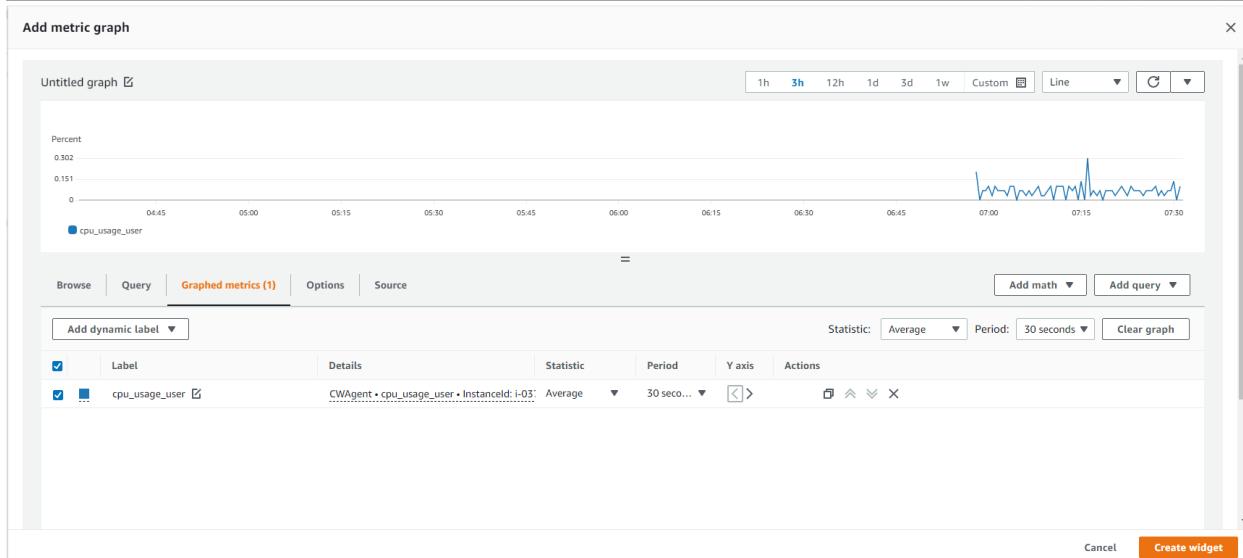
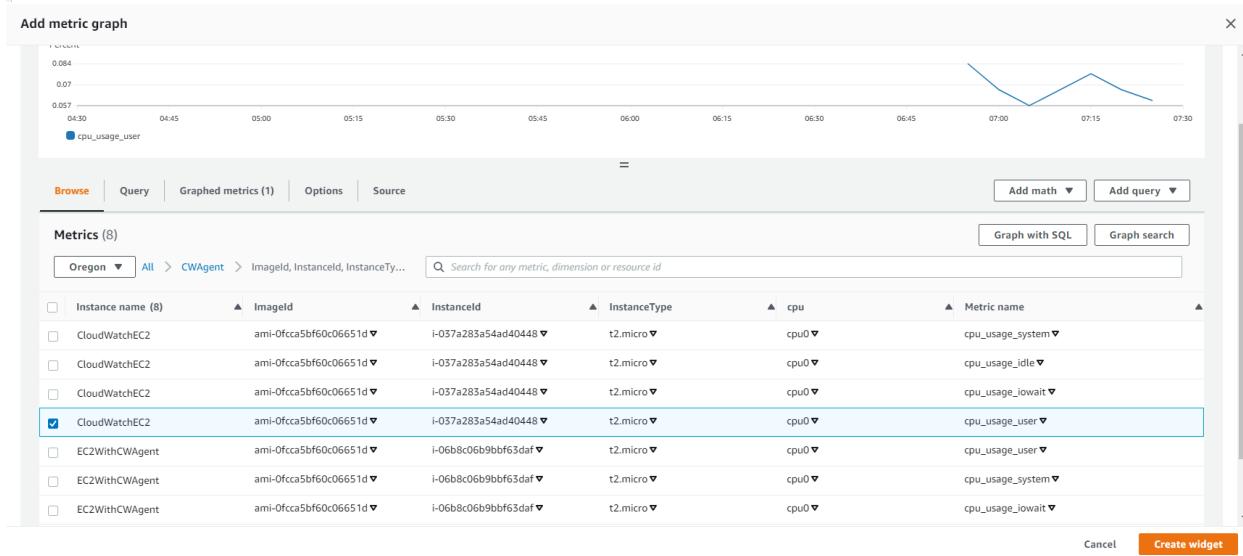
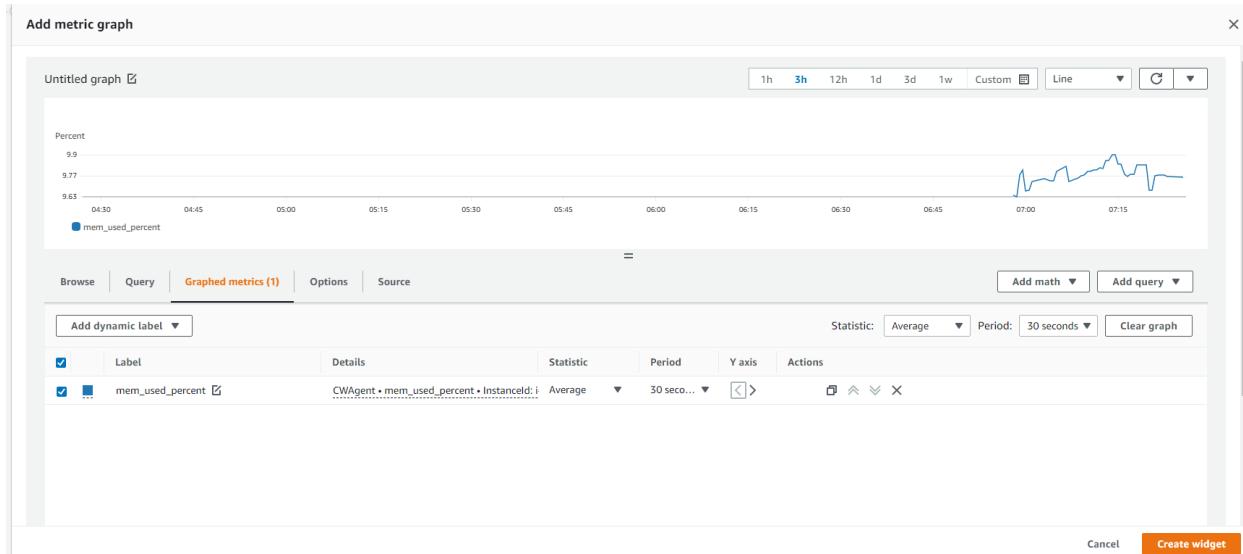
mem_used_percent

Browse Query Graphed metrics (1) Options Source
Add math Add query

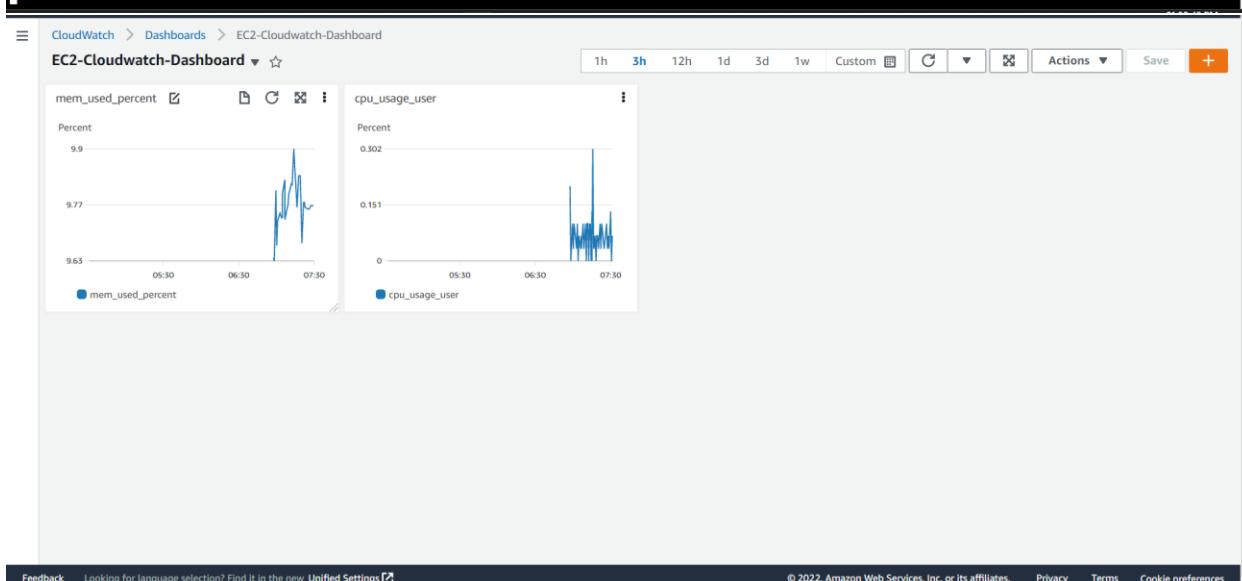
Metrics (4)

Oregon	All	CWAgent	ImageId, InstanceId, InstanceType	Search for any metric, dimension or resource id
<input type="checkbox"/> Instance name (4)	<input type="checkbox"/> ImageId	<input type="checkbox"/> InstanceId	<input type="checkbox"/> InstanceType	<input type="checkbox"/> Metric name
<input checked="" type="checkbox"/> CloudWatchEC2	ami-0fcfa5bf60c06651d	i-037a283a54ad40448	t2.micro	mem_used_percent
<input type="checkbox"/> CloudWatchEC2	ami-0fcfa5bf60c06651d	i-037a283a54ad40448	t2.micro	swap_used_percent
<input type="checkbox"/> EC2WithCWAgent	ami-0fcfa5bf60c06651d	i-06b8c06b9bbf63daf	t2.micro	mem_used_percent
<input type="checkbox"/> EC2WithCWAgent	ami-0fcfa5bf60c06651d	i-06b8c06b9bbf63daf	t2.micro	swap_used_percent

Cancel Create widget



```
sh-4.2$ sudo stress --cpu 8 --vm-bytes $(awk '/MemAvailable/{printf "%d\n", $2 * 0.9;} < /proc/meminfo)k --vm-keep -m 1 --timeout 60
stress: info: [3595] dispatching hogs: 8 cpu, 0 io, 1 vm, 0 hdd
```



Feedback Looking for language selection? Find it in the new [Unified Settings](#)

© 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

ALARM: "EC2-CloudWatch-Alarm" in US West (Oregon)

AWS Notifications <no-reply@sns.amazonaws.com> to me 1:04 PM (0 minutes ago)

You are receiving this email because your Amazon CloudWatch Alarm "EC2-CloudWatch-Alarm" in the US West (Oregon) region has entered the ALARM state, because "Threshold Crossed: 1 out of the last 1 datapoints [77.4420482066659 (24/11/22 07:34:00)] was greater than or equal to the threshold (50.0) (minimum 1 datapoint for OK -> ALARM transition)." at "Thursday 24 November, 2022 07:34:59 UTC".

View this alarm in the AWS Management Console:
<https://us-west-2.console.aws.amazon.com/cloudwatch/deeplink.js?region=us-west-2#alarmsV2.alarm:EC2-CloudWatch-Alarm>

Alarm Details:

- Name: EC2-CloudWatch-Alarm
- Description:
- State Change: INSUFFICIENT_DATA -> ALARM
- Reason for State Change: Threshold Crossed: 1 out of the last 1 datapoints [77.4420482066659 (24/11/22 07:34:00)] was greater than or equal to the threshold (50.0) (minimum 1 datapoint for OK -> ALARM transition).
- Timestamp: Thursday 24 November, 2022 07:34:59 UTC
- AWS Account: 084044351869
- Alarm Arn: arn:aws:cloudwatch:us-west-2:084044351869:alarm:EC2-CloudWatch-Alarm

Threshold:

- The alarm is in the ALARM state when the metric is GreaterThanOrEqualToThreshold 50.0 for at least 1 of the last 1 period(s) of 10 seconds.

Monitored Metric:

- MetricNamespace: CWAgent
- MetricName: mem_used_percent

- Alarm Arn: arn:aws:cloudwatch:us-west-2:084044351869:alarm:EC2-CloudWatch-Alarm

Threshold:

- The alarm is in the ALARM state when the metric is GreaterThanOrEqualToThreshold 50.0 for at least 1 of the last 1 period(s) of 10 seconds.

Monitored Metric:

- MetricNamespace: CWAgent
- MetricName: mem_used_percent
- Dimensions: [InstanceId = i-037a283a54ad40448] [ImageId = ami-0fcca5bf60c06651d] [InstanceType = t2.micro]
- Period: 10 seconds
- ExtendedStatistic: p50
- Unit: not specified
- TreatMissingData: missing

State Change Actions:

- OK:
- ALARM: [arn:aws:sns:us-west-2:084044351869:ec2-email-sns-topic>EmailSNSTopic-u3CzncbtCv3F] [arn:aws:sns:us-west-2:084044351869:EC2_CloudWatch_SNS_Topic]
- INSUFFICIENT_DATA:

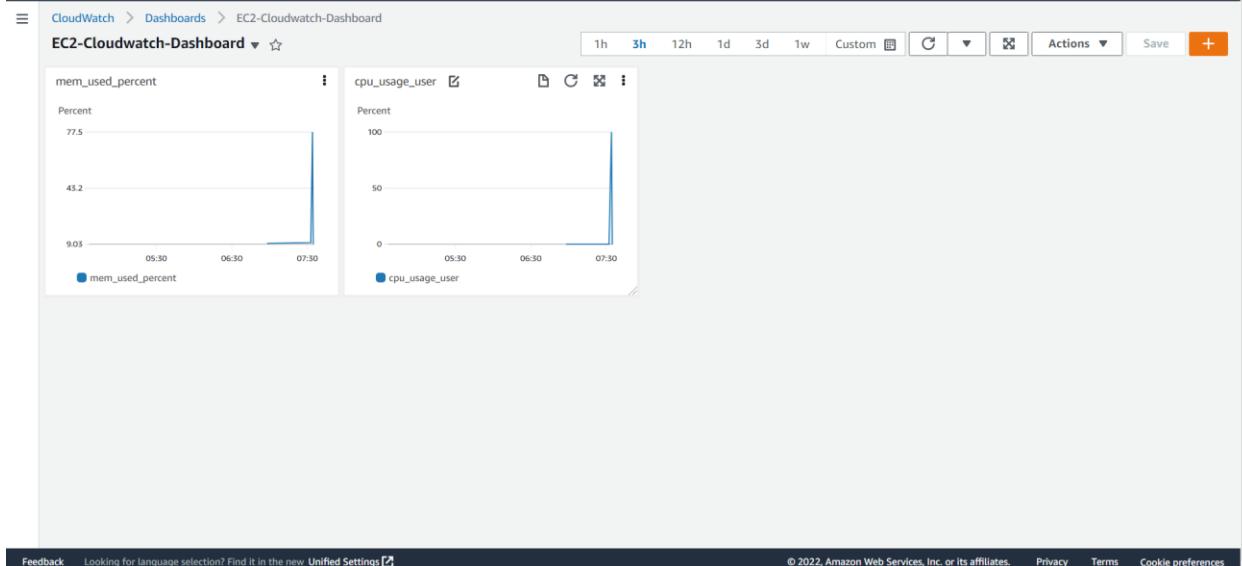
--

If you wish to stop receiving notifications from this topic, please click or visit the link below to unsubscribe:

https://sns.us-west-2.amazonaws.com/unsubscribe.html?SubscriptionArn=arn:aws:sns:us-west-2:084044351869:EC2_CloudWatch_SNS_Topic:a4d07d6c-59f7-41f2-a1de-649221e17c1d&Endpoint=mayankchaturvedi9991@gmail.com

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

```
sh-4.2$ sudo stress --cpu 8 --vm-bytes $(awk '/MemAvailable/{printf "%d\n", $2 * 0.9;} < /proc/meminfo') --vm-keep -m 1 --timeout 60
stress: info: [3595] dispatching hogs: 8 cpu, 0 io, 1 vm, 0 hdd
stress: info: [3595] successful run completed in 60s
sh-4.2$
```



Written Steps :

Install a CloudWatch Agent on an Existing EC2 Instance

You are working as a SysOps Engineer in the IT Software Industry. The R&D team has asked you to create a CloudWatch Dashboard and configure various business use-cases for an existing EC2 production instance so that the Client Success Team can monitor an EC2 instance for performance purposes.

In this lab, you will create metrics at Amazon CloudWatch and install a CloudWatch Agent on an existing EC2 instance to start collecting Memory and Network utilizations. You will also create an alarm when CPU goes up above 50% and send an automatic notification email to your email address.

1. Log in to the AWS web console with the lab credentials provided. First, ensure that the region drop-down in the top right is set to **US West (Oregon)**, then type EC2 in the search box and click **Enter**.
2. On the left-hand **Instances** column menu, click on the **Instances** option.
3. Choose an existing EC2 instance named **CloudWatchEC2** and wait until Instance State is **Running**.
4. Click on **Connect** button at the top of the page.
5. Choose the **Session Manager** tab and click on **Connect** button at the bottom right of the page.
6. You'll be redirected to an EC2 Linux console window in a different browser tab. In EC2 console, download the CloudWatch Agent using below command :

```
sudo wget https://s3.amazonaws.com/amazoncloudwatch-agent/amazon_linux/amd64/latest/amazon-cloudwatch-agent.rpm
```

Note: Above instruction is for Amazon Linux OS. Find the right agent link for your OS by visiting a [link](#).

7. To install the CloudWatch Agent, type below command on EC2 console :

```
sudo rpm -U ./amazon-cloudwatch-agent.rpm
```

8. Run below command to create a CloudWatch Agent configuration file for EC2 instance.

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

Note: Before running the CloudWatch Agent on any server, you must create a CloudWatch Agent configuration JSON file that specifies the metrics and logs which agent needs to collect.

9. In the AWS CloudWatch Agent Configuration Manager, answer the question as following :

- On which OS are you planning to use the agent? – 1 (linux)
- Are you using EC2 or On-Premises hosts? – 1 (EC2)
- Which user are you planning to run the agent? – 2 (cwagent)
- Do you want to turn on StatsD daemon? – 2 (no)

- Do you want to monitor metrics from CollectD? – 2 (no)
- Do you want to monitor any host metrics? E.g CPU, memory, etc – 1 (yes)
- Do you want to monitor cpu metrics per core? – 1 (yes)
- Do you want to add ec2 dimensions (ImageId, InstanceId, InstanceType, AutoScalingGroupName) into all of your metrics if the info is available? – 1 (yes)
- Do you want to aggregate ec2 dimensions (InstanceId)? - 1 (yes)
- Would you like to collect your metrics at high resolution (sub-minute resolution)? – 2 (10s)
- Which default metrics config do you want? – 2 (Standard)
- Are you satisfied with the above config? – 1 (yes)
- Do you have any existing CloudWatch Log Agent (<http://docs.aws.amazon.com/AmazonCloudWatch/latest/logs/AgentReference.html>) configuration file to import for migration? – 2 (no)
- Do you want to monitor any log files? – 2 (no)
- Do you want to store the config in the SSM parameter store? – 2 (no)

10. Start the CloudWatch Agent by running below command :

```
sudo /opt/aws/amazon-cloudwatch-agent/bin/amazon-cloudwatch-agent-ctl -a fetch-config -m ec2
-c file:/opt/aws/amazon-cloudwatch-agent/bin/config.json -s
```

11. Validate CloudWatch Agent service is in running state or not using below command :

```
sudo systemctl status amazon-cloudwatch-agent
[ec2-user@ip-172-31-30-199 ~]$ systemctl status amazon-cloudwatch-agent
● amazon-cloudwatch-agent.service - Amazon CloudWatch Agent
  Loaded: loaded (/etc/systemd/system/amazon-cloudwatch-agent.service; enabled; vendor preset: disabled)
  Active: active (running) since Tue 2021-06-15 16:18:45 UTC; 12s ago
    Main PID: 1457 (amazon-cloudwat)
       CGroup: /system.slice/amazon-cloudwatch-agent.service
               └─1457 /opt/aws/amazon-cloudwatch-agent/bin/amazon-cloudwatch-agent -config /opt/aws/amazon-cl

Jun 15 16:18:45 ip-172-31-30-199.us-west-2.compute.internal systemd[1]: Started Amazon CloudWatch Agent.
Jun 15 16:18:45 ip-172-31-30-199.us-west-2.compute.internal systemd[1]: Starting Amazon CloudWatch Agent..
Jun 15 16:18:45 ip-172-31-30-199.us-west-2.compute.internal start-amazon-cloudwatch-agent[1457]: /opt/aws/
Jun 15 16:18:45 ip-172-31-30-199.us-west-2.compute.internal start-amazon-cloudwatch-agent[1457]: Valid Jso
Jun 15 16:18:45 ip-172-31-30-199.us-west-2.compute.internal start-amazon-cloudwatch-agent[1457]: I! Detect
Hint: Some lines were ellipsized, use -l to show in full.
[ec2-user@ip-172-31-30-199 ~]$
```

Congratulations! You have successfully installed and configured the CloudWatch Agent on your EC2 instance. Now, you should be able to capture Memory and Network utilization of an EC2 instance in the CloudWatch.

Configure Multiple Metrics with Alarms

Now that you've installed and configured CloudWatch Agent on your EC2 instance in the previous challenge, you have been asked to create and configure detailed metrics using the Alarms option and associate a new SNS topic with a valid email subscription.

1. In the AWS console, type CloudWatch in the search box and click **Enter**.
2. On the left-hand **CloudWatch** column menu, click on the **Alarms** option.
3. Click on the **Create alarm** button from the right side of the Alarms page.
4. Click on the **Select metric** button and under the **Metrics** tab click on the **CWAgent** custom namespace.
5. Click on **ImageID,InstanceId,InstanceType** and select a **CloudWatchEC2** instance with a Metric Name **mem_used_percent** from the list.

Note: It may take a few minutes to display available metrics for **CloudWatchEC2** instance. If metrics are still not displayed for your EC2 instance then try to reload the page and restart the process from step 4 again.

6. Click on the **Select metric** button at the bottom right of the page.
7. Provide field values under the Graph section as below:
 - o **Metric name** - Keep default value.
 - o **InstanceId** - Keep default value.
 - o **ImageId** - Keep default value.
 - o **InstanceType** - Keep default value.
 - o **Statistics** - Type and select **p50**.
 - o **Period** - Choose **10 seconds** from dropdown.
8. Provide field values under the Conditions section as below:
 - o **Threshold type** - keep **Static** as default selected.
 - o **Whenever mem_available_percent is... - Greater/Equal** (\geq threshold).
 - o **than....** - Provide the threshold value as 50.
9. Leave the Additional configuration section untouched and click on the **Next** button on the bottom right of the page.
10. In the Configure actions page, click the **Add notification** button.
11. In the **Notification** section, keep **In alarm** option as selected.
12. Choose the **Create new topic** option in the Select an SNS topic section.
13. Provide the topic name as **EC2_CloudWatch_SNS_Topic** in the **Create a new topic...** field and provide a valid email address that you can use to receive the notification in the **Email endpoints that will receive the notification...** field.
14. Click on the **Create topic** button and wait for the process to finish.
15. Leave all other sections untouched and click on the **Next** button at the right bottom of the page.
16. Provide the **Alarm name** as **EC2-CloudWatch-Alarm**, leave the **Alarm description** field untouched and click on the **Next** button at the bottom of the page.
17. Review the alarm on the preview page and click on the **Create alarm** button at the right bottom of the page.

Congratulations! You have successfully created an Alarm for an EC2 instance in the CloudWatch. You can search for the same alarm on the list of the **Alarms** page by

using the name provided in the Alarm Name field. Also, click on the **Name** hyperlink to see more details. You can also refresh the page until Alarm state is changed from **Insufficient data** to **OK**.

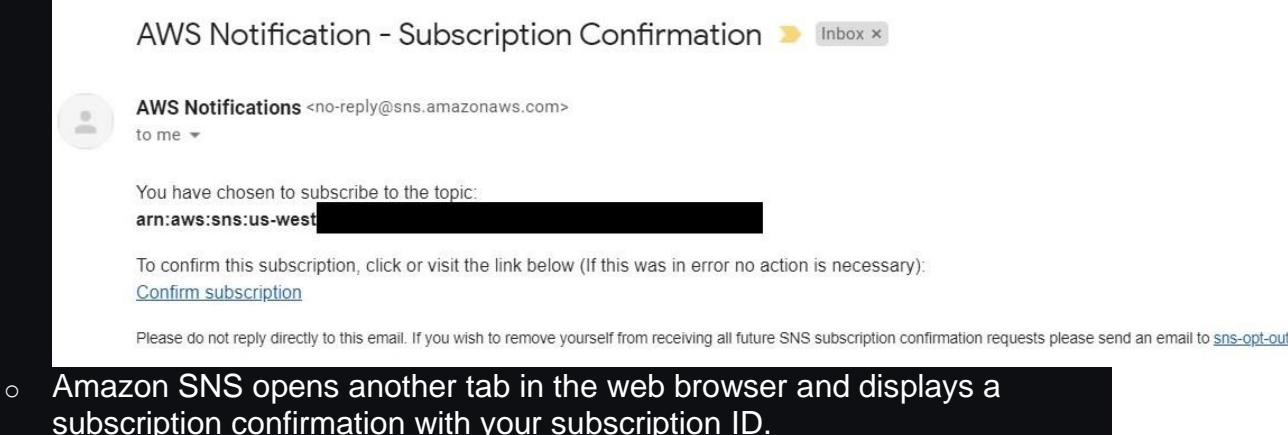
Enable an Email Notification with the SNS

Now that you've created an Alarm and SNS Topic for an email notification in the previous challenge, you should confirm an email subscription in order to receive an alarm on your email address.

1. In the AWS web console type SNS (Simple Notification Service) and click **Enter**.
2. On the left-hand **Amazon SNS** column menu, click on the **Subscriptions** option.

Note: The SNS topic **EC2_CloudWatch_SNS_Topic** is listed here with the status as **Pending confirmation**.

3. Open your email inbox to confirm the subscription:
 - o Check your email inbox and choose **Confirm subscription** in the email from Amazon SNS.



- o Amazon SNS opens another tab in the web browser and displays a subscription confirmation with your subscription ID.



Simple Notification Service

Subscription confirmed!

You have successfully subscribed.

Your subscription's id is:

arn:aws:sns:us-west-

If it was not your intention to subscribe, [click here to unsubscribe](#).

Note: Please **do not** click the *Unsubscribe* button on this email subscription confirmation page; doing so is unnecessary and can interfere with the lab. Once you're done with the lab you will be unsubscribed automatically and will receive no further email alerts.

4. Now, navigate back to the AWS web console tab and refresh the Subscriptions page.
5. In the Subscriptions list page, you will see the **Status** for **EC2_CloudWatch_SNS_Topic** is now **Confirmed**.

Congratulations! You have successfully subscribed to an email notification whenever an alarm is triggered. You can search for the same subscription on the list of the Subscriptions page by using the name provided in the **SNS Topic Name** field. Also, click on the **ID** hyperlink to see more details.

Display Metrics in a Dashboard

Now that you have set up and configured CloudWatch Agent on your EC2 instance, an Alarm with a threshold above 50% CPU usage, you need to create a CloudWatch Dashboard and validate whether the metrics graphs are functioning properly or not.

1. In the AWS web console type CloudWatch in the search box and click **Enter**.
2. On the left-hand **CloudWatch** column menu, click on the **Dashboards** option.

Note: If you have not done so already, make sure you click “**Try out the new interface**” to see the updated CloudWatch interface.

3. Click on the **Create dashboard** button.
4. Provide a Dashboard name as EC2-Cloudwatch-Dashboard and click on the **Create dashboard** button.
5. In the Add widget section and choose the **Line** widget and then choose **Metrics** from the **Add to this dashboard** popup window.
6. In the **Add metric graph** popup window, click on the **CWAgent** custom namespace under the Metrics section.
7. Click on the **ImageId**, **InstanceId**, **InstanceType** and select a **CloudWatchEC2** instance with a **Metric Name** as **mem_used_percent** from the list.
8. Click on the **View graphed metrics** button and choose the **30 Seconds** value for **Period** field dropdown.
9. Click on the **Create widget** button on the right bottom of the page.
10. Click on the **Add widget** button again on the right top of the page.
11. Choose **Line** widget and then choose **Metrics** from **Add to this dashboard** popup window.
12. In the **Add metric graph** popup window, click on the **CWAgent** custom namespace under the Metrics section.
13. Click on the **ImageId**, **InstanceId**, **InstanceType**, **cpu** and select a **CloudWatchEC2** instance with a **Metric Name** as **cpu_usage_user** from the list.
14. Click on the **View graphed metrics** button and choose the **30 Seconds** value for **Period** field dropdown.
15. Click on the **Create widget** button on the right bottom of the page.
16. Click on the **Save dashboard** button on the right top of the page.
17. Navigate back to an EC2 Linux console tab opened previously. In EC2 console, type below command to create a CPU spike:

```
sudo stress --cpu 8 --vm-bytes $(awk '/MemAvailable/{printf "%d\n", $2 * 0.9;}' < /proc/meminfo)k  
--vm-keep -m 1 --timeout 60
```

Note: If **CloudWatchEC2** instance's Linux console tab has timed out or the session has expired then try to reload the UI page or reconnect using **Session Manager** option from Connect to instance page or you can repeat steps **1-5** provided in the first challenge.

18. Now, navigate to the AWS Console opened in another tab and click on the **reload icon** to refresh the dashboard.

Congratulations! You have successfully created the dashboard. It shows the line chart for CPU and Memory usage collected by a Cloudwatch Agent that is installed on the EC2 instance.

By refreshing the AWS Console you can see the alarm state moved to **In Alarm** state as CPU goes above 50%. Open your email inbox and you can see email by **AWS Notifications** whenever an alarm triggers. It may take a few minutes to receive an email. You can add an alarm into the dashboard and also drag widgets to beautify it.