

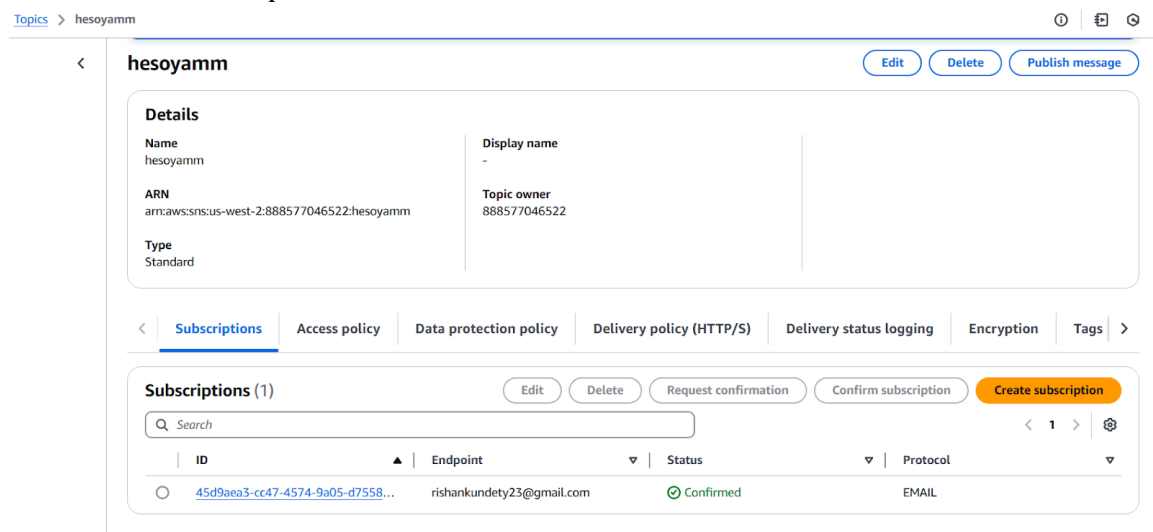
EC2 Health Check Automation Project

Project Objective

Automating the health check of a specific EC2 instance. The check must occur every day and a specific time of day. If the instance is found to be other than running, the user must get the notification. Use AWS Lambda, Amazon CloudWatch, Amazon SNS.

Step-by-Step Approach

1. Created an SNS Topic named and subscribed an email address to receive notifications.



2. Created an IAM Role with the following permissions:

- AmazonEC2ReadOnlyAccess
- AmazonSNSFullAccess
- AWSLambdaBasicExecutionRole

Additionally, a custom trust policy was used to allow Lambda and EventBridge Scheduler to assume the role:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": [
          "scheduler.amazonaws.com",
          "lambda.amazonaws.com"
        ]
      }
    }
  ]
}
```

```

    ],
    },
    "Action": "sts:AssumeRole"
  }
]
}

```

healthbarcheck

Trust policy updated.

Permissions policies (3) Info

You can attach up to 10 managed policies.

Filter by Type

Search

All types

Policy name	Type	Attached entities
AmazonEC2ReadOnlyAccess	AWS managed	1
AmazonSNSFullAccess	AWS managed	2
AWSLambdaBasicExecutionRole	AWS managed	2

Permissions boundary (not set)

Permissions Trust relationships Tags Last Accessed Revoke sessions

Trusted entities

Entities that can assume this role under specified conditions.

```

1 {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Effect": "Allow",
6       "Principal": {
7         "Service": [
8           "scheduler.amazonaws.com",
9           "lambda.amazonaws.com"
10        ]
11      },
12      "Action": "sts:AssumeRole"
13    }
14  ]
15 }

```

3. Created a Lambda function named with the IAM role above. The function checks the status of a specific EC2 instance and sends an SNS alert if it is not in the 'running' state.

Lambda > Functions > youalivehuhh

youalivehuhh

Throttle Copy ARN Actions

Function overview Info

Export to Infrastructure Composer Download

Diagram Template

youalivehuhh

Layers (0)

+ Add trigger

SNS

+ Add destination

Description

Last modified 41 minutes ago

Function ARN arn:aws:lambda:us-west-2:888577046522:function:youalivehuhh

Function URL Info

4. Configured Amazon EventBridge (CloudWatch) to trigger the Lambda function daily at

the specified time using a scheduled rule.

Schedules > checkmamanshealth

Schedule checkmamanshealth has been saved successfully. [X]

checkmamanshealth [Disable] [Edit] [Delete]

Schedule detail			
Schedule name checkmamanshealth	Status Enabled	Schedule start time -	Flexible time window -
Description -	Schedule ARN arn:aws:scheduler:us-west-2:888577046522:schedule/default/checkmamanshealth	Schedule end time -	Created date Apr 15, 2025, 16:40:00 (UTC+05:30)
Schedule group name default	Action after completion NONE	Execution time zone Asia/Calcutta	Last modified date Apr 15, 2025, 17:13:37 (UTC+05:30)

Schedule | Target | Retry policy | Dead-letter queue | Encryption

Schedule
One time schedule [Info](#)
Your schedule is set for Tue, Apr 15, 2025, 17:14:00 (UTC+05:30).

Python Code Used in Lambda Function

```
import boto3
```

```
sns = boto3.client('sns')
```

```
ec2 = boto3.client('ec2')
```

```
SNS_TOPIC_ARN = 'arn:aws:sns:us-west-2:888577046522:hesoyamm'
```

```
INSTANCE_ID = 'i-0decbfa3d65889407'
```

```
def lambda_handler(event, context):
```

```
    try:
```

```
        response = ec2.describe_instance_status(
            InstanceIds=[INSTANCE_ID],
            IncludeAllInstances=True
        )
```

```
    if not response['InstanceStatuses']:
```

```
        msg = f"Warning: No status available for EC2 instance {INSTANCE_ID}."
```

```
        sns.publish(TopicArn=SNS_TOPIC_ARN, Message=msg, Subject='EC2 Health Check Warning')
```

```
        return {
```

```
            'statusCode': 404,
```

```
            'body': msg
```

```
        }
```

```

state = response['InstanceStatuses'][0]['InstanceState']['Name']

if state != 'running':
    msg = f"Alert: EC2 instance {INSTANCE_ID} is in '{state}' state!"
    sns.publish(TopicArn=SNS_TOPIC_ARN, Message=msg, Subject='EC2 Health Alert')

return {
    'statusCode': 200,
    'body': f"Instance state: {state}"
}

except Exception as e:
    err_msg = f"Error checking instance status: {str(e)}"
    sns.publish(TopicArn=SNS_TOPIC_ARN, Message=err_msg, Subject='EC2 Health Check
Error')
    return {
        'statusCode': 500,
        'body': err_msg
    }

```

Challenges Solved

- Encountered permission error (UnauthorizedOperation) due to missing `ec2:DescribeInstanceStatus` permission in the IAM role.
- Resolved it by adding a custom inline policy to the Lambda role.
- Ensured the trust relationship allowed Lambda and scheduler services.
- Verified CloudWatch EventBridge successfully triggered the function daily.

Result

Successfully implemented a daily health check system for a specific EC2 instance. Alerts are now sent via email through SNS when the instance is not running, using Lambda and CloudWatch automation.

Instances > i-05c8316f9f95d5e19

View

Instance summary for i-05c8316f9f95d5e19 (health) info

Updated less than a minute ago

Instance ID

i-05c8316f9f95d5e19

IPv6 address

-

Hostname type

-

Answer private resource DNS name

-

Auto-assigned IP address

-

IAM Role

-

IMDSv2

Required

Operator

-

Public IPv4 address

-

Instance state

Terminated

Instance type

t2.micro

VPC ID

-

Subnet ID

-

Instance ARN

arn:aws:ec2:us-west-2:888577046522:instance/i-05c8316f9f95d5e19

Private IPv4 addresses

-

Public IPv4 DNS

-

Elastic IP addresses

-

AWS Compute Optimizer finding

Opt-in to AWS Compute Optimizer for recommendation 5. | Learn more

Auto Scaling Group name

-

Managed

false

Feedback

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

EC2 Health Alert

Inbox x

17:11 (23 minutes ago)

Alert: EC2 instance i-05c8316f9f95d5e19 is in 'terminated' state!

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/subscribe.html?SubscriptionArn=arn:aws:sns:us-west-2:888577046522:heshoyamm45d9aea3-cc47-4574-9a05-d75581fd7cdf&Endpoint=rishankundety23@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>

Reply

Forward

/aws/lambda/youalievuh > 2025/04/15/[SLATEST]9f5c442c3b594788ab1a1368a2188325

Log events

You can use the filter bar below to search for and match terms, phrases, or values in your log events. [Learn more about filter patterns](#)

Filter events - press enter to search

Clear 1m 30m 1h 12h Custom UTC timezone Display

Timestamp	Message
No older events at this moment. Retry	
2025-04-15T11:17:51.376Z	INIT_START Runtime Version: nodejs12.x V3 Runtime Version ARN: arn:aws:lambda:us-west-2::runtime:sc0e80a3cbacc3e60742d757c16557fd39490b17ec433514cf66810b0ed70
2025-04-15T11:17:51.522Z	2025-04-15T11:17:51.522Z undefined ERROR Uncaught Exception ("errorType":"Runtime.UserCodeSyntaxError","errorMessage":"SyntaxError: Unexpected token 'import'", "stack":["Runtime.UserCodeSyntaxError: SyntaxError...
2025-04-15T11:17:51.541Z	INIT_REPORT Init Duration: 164.99 ms Phase: Init Status: error Error Type: Runtime.UserCodeSyntaxError
2025-04-15T11:17:52.643Z	2025-04-15T11:17:52.643Z undefined ERROR Uncaught Exception ("errorType":"Runtime.UserCodeSyntaxError","errorMessage":"SyntaxError: Unexpected token 'import'", "stack":["Runtime.UserCodeSyntaxError: SyntaxError...
2025-04-15T11:17:52.683Z	INIT_REPORT Init Duration: 1326.00 ms Phase: Invoke Status: error Error Type: Runtime.UserCodeSyntaxError
2025-04-15T11:17:52.683Z	START RequestId: 718090eb-d7ee-45df-a808-5df313efc408 Version: SLATEST
2025-04-15T11:17:52.931Z	END RequestId: 718090eb-d7ee-45df-a808-5df313efc408
2025-04-15T11:17:52.931Z	REPORT RequestId: 718090eb-d7ee-45df-a808-5df313efc408 Duration: 1353.29 ms Billed Duration: 1354 ms Memory Size: 128 MB Max Memory Used: 72 MB Status: error Error Type: Runtime.UserCodeSyntaxError
2025-04-15T11:19:56.964Z	2025-04-15T11:19:56.964Z undefined ERROR Uncaught Exception ("errorType":"Runtime.UserCodeSyntaxError","errorMessage":"SyntaxError: Unexpected token 'import'", "stack":["Runtime.UserCodeSyntaxError: SyntaxError...
2025-04-15T11:19:57.165Z	INIT_REPORT Init Duration: 1267.74 ms Phase: Invoke Status: error Error Type: Runtime.UserCodeSyntaxError
2025-04-15T11:19:57.165Z	START RequestId: 0654f868-0140-4caa-b898-af970d62338 Version: SLATEST
2025-04-15T11:19:57.176Z	END RequestId: 0654f868-0140-4caa-b898-af970d62338
2025-04-15T11:19:57.176Z	REPORT RequestId: 0654f868-0140-4caa-b898-af970d62338 Duration: 1267.83 ms Billed Duration: 1268 ms Memory Size: 128 MB Max Memory Used: 72 MB Status: error Error Type: Runtime.UserCodeSyntaxError
2025-04-15T11:20:35.823Z	2025-04-15T11:20:35.823Z undefined ERROR Uncaught Exception ("errorType":"Runtime.UserCodeSyntaxError","errorMessage":"SyntaxError: Unexpected token 'import'", "stack":["Runtime.UserCodeSyntaxError: SyntaxError...
2025-04-15T11:20:36.043Z	INIT_REPORT Init Duration: 1258.95 ms Phase: Invoke Status: error Error Type: Runtime.UserCodeSyntaxError
2025-04-15T11:20:36.043Z	START RequestId: 6c517044-9438-49dc-a18f-733b7fc7db04 Version: SLATEST
2025-04-15T11:20:36.052Z	END RequestId: 6c517044-9438-49dc-a18f-733b7fc7db04
2025-04-15T11:20:36.052Z	REPORT RequestId: 6c517044-9438-49dc-a18f-733b7fc7db04 Duration: 1259.00 ms Billed Duration: 1260 ms Memory Size: 128 MB Max Memory Used: 72 MB Status: error Error Type: Runtime.UserCodeSyntaxError
No newer events at this moment. Auto retry paused. Resume	