

# Core Assignment 4: Log the State of an Amazon EC2 Instance Using CloudWatch Events

Muhannad Sinan

Log the State of an AWS EC2 Instance Using CloudWatch Events (now EventsBridge)

You can create an AWS Lambda function that logs the changes in state for an Amazon EC2 instance.

You can choose to create a rule that runs the function whenever there is a state transition or a transition to one or more states that are of interest.

**Follow the below steps:**

1. Create a Lambda function to log the state change events. You specify this function when you create your rule.

Example code for logging the state of EC2 instance

```
1 'use strict';
2 exports.handler = (event, context, callback) => {
3   console.log('LogEC2InstanceStateChange');
4   console.log('Received event:', JSON.stringify(event, null, 2));
5   callback(null, 'Finished');
6 };
```

1. Create a rule to run your Lambda function whenever you launch an Amazon EC2 instance.
2. Test the rule -- To test your rule, launch an Amazon EC2 instance. After waiting a few minutes for the instance to launch and initialize, you can verify that your Lambda function was invoked.
3. Take a screen shot of your results.

## Step1:


Creating a Lambda function to log the state change events. You specify this function when you create your rule.


Lambda > Functions > EC2StatetoCloudWatch

## EC2StatetoCloudWatch

Throttle Copy ARN Actions

▼ Function overview Info

 EC2StatetoCloudWatch

 Layers (0)

+ Add trigger

+ Add destination

Description  
-

Last modified  
27 seconds ago

Function ARN  
arn:aws:lambda:me-south-1:568935291733:function:EC2StatetoCloudWatch

Function URL Info  
-

Code Test Monitor Configuration Aliases Versions

Code source Info Upload from

File Edit Find View Go Tools Window Test Deploy Changes not deployed

Go to Anything (Ctrl-P)

Environment

index.js

```
1 'use strict';
2 exports.handler = (event, context, callback) => {
3   console.log('LogEC2InstanceStateChange');
4   console.log('Received event:', JSON.stringify(event, null, 2));
5   callback(null, 'finished');
6 }
```

6:3 JavaScript Spaces: 4

Creating a rule to run your Lambda function whenever you launch an Amazon EC2 instance.

### Event pattern Info

Event pattern form Custom patterns (JSON editor)

**AWS service**  
The name of the AWS service as the event source

EC2

**Event type**  
The type of events as the source of the matching pattern

EC2 Instance State-change Notification

☐ Any state

☒ Specific state(s)

pending

☒ Any instance

☐ Specific instance Id(s)

**Event pattern**  
Event pattern, or filter to match the events

```
1 {
2   "source": ["aws.ec2"],
3   "detail-type": ["EC2 Instance State-change Notification"]
4   "detail": {
5     "state": ["pending"]
6   }
7 }
```

Copy Test pattern Edit pattern

Cancel Previous Next

## Target 1

### Target types

Select an EventBridge event bus, EventBridge API destination (SaaS partner), or another AWS service as a target.

- ☐ EventBridge event bus
- ☐ EventBridge API destination
- ☒ AWS service

### Select a target [Info](#)

Select target(s) to invoke when an event matches your event pattern or when schedule is triggered (limit of 5 targets per rule)

Lambda function ▼

### Function

EC2StatetoCloudWatch ▼



► [Configure version/alias](#)

► [Additional settings](#)

[Add another target](#)

[Cancel](#)

[Previous](#)

[Next](#)

Amazon EventBridge > Rules > Create rule

Step 1

Define rule detail

Step 2

Build event pattern

Step 3

Select target(s)

Step 4 - optional

Configure tags

Step 5

Review and create

## Review and create

Step 1: Define rule detail Edit

Define rule detail

Rule name EC2StatetoCloudWatch	Status ✔ Enabled	Event bus default
Description	Rule type Standard rule	

Step 2: Build event pattern Edit

Event pattern Info

```

1 {
2   "source": ["aws.ec2"],
3   "detail-type": ["EC2 Instance State-change Notification"],
4   "detail": {
5     "state": ["pending"]
6   }
7 }

```

Copy

Step 3: Select target(s) Edit

Targets

Targets

Target Name	Type	Arn	Input	Role
▼ EC2StatetoCloudWatch <span>🔗</span>	Lambda function	arn:aws:lambda:me-south-1:568935291733:function:EC2StatetoCloudWatch	Matched event	-

Input to target:

Matched event

Additional parameters:

-

Dead-letter queue (DLQ):

-

Step 4: Configure tag(s) Edit

Tags

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key	Value

Cancel

Previous

Create rule

Testing the rule -- To test your rule, launch an Amazon EC2 instance. After waiting a few minutes for the instance to launch and initialize, you can verify that your Lambda function was invoked.

- lunching an instance

Instances (1) <span>Info</span>									
<input type="text" value="Search"/>									
<input type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	
<input type="checkbox"/>	Testing	i-0baf05e7efd274c5c	✔ Running <span>🔍</span>	t3.micro	⌚ Initializing	No alarms <span>+</span>	me-south-1a	ec2-15-185-66-68.	

- EventBridge Monitoring




EC2StatetoCloudWatch


Throttle


Copy ARN


Actions


Function overview

 EC2StatetoCloudWatch

 Layers (0)

 EventBridge (CloudWatch Events)





Description

-

Last modified

7 minutes ago

Function ARN

arn:aws:lambda:me-south-1:568935291753:function:EC2StatetoCloudWatch

Function URL

[Info](#)

Code | Test | Monitor | Configuration | Aliases | Versions

Metrics | Traces

View logs in CloudWatch | View X-Ray traces in ServiceLens | View Lambda Insights

CloudWatch metrics

Filter by Function

Lambda sends runtime metrics for your functions to Amazon CloudWatch. The metrics shown are an aggregate view of all function runtime activity. To view metrics for the unqualified or \$LATEST resource, choose **Filter by**. To view metrics for a specific function version or alias, choose **Aliases** or **Versions**, select the alias or version, and then choose **Monitor**.

1h 3h 12h 1d 3d 1w Custom | Add to dashboard

