

Project Overview

The goal of this project is to ensure that any newly created EBS volume of type GP2 is automatically converted to GP3 using AWS CloudWatch Events and AWS Lambda.

Understanding the Problem: Amazon EBS provides different volume types, each designed to cater to specific workload requirements. GP2 volumes offer a balance between performance and cost, while GP3 volumes provide higher performance at a lower price per gigabyte. Converting GP2 volumes to GP3 when appropriate can lead to significant cost savings without sacrificing performance.

Solution Overview: To automate the conversion process, we will utilize the power of AWS CloudWatch Events and Lambda. CloudWatch Events enables us to monitor events within our AWS environment, while Lambda allows us to execute custom code in response to those events. By combining these services, we can seamlessly convert GP2 volumes to GP3 type whenever specific conditions are met.

Prerequisites

1. **AWS Account:** Ensure you have access to an AWS account with appropriate permissions.
2. **IAM Roles and Policies:** Create IAM roles with the necessary permissions for Lambda functions and CloudWatch Events.

Steps

Step 1: Set Up IAM Roles and Policies

1. **Create IAM Role for Lambda Function:**
 - Navigate to the IAM console.
 - Create a new role and select Lambda as the trusted entity.
 - Attach the following policies:
 - AmazonEC2FullAccess
 - CloudWatchEventsFullAccess

- `AWSLambdaBasicExecutionRole`

2. Create IAM Policy for Specific Permissions (Optional):

- If you prefer more restrictive permissions, create a custom policy that allows actions such as describing, modifying, and attaching EBS volumes.

▼ **EC2** Allow 2 Actions

Specify what actions can be performed on specific resources in `EC2`.

▼ **Actions allowed**

Specify actions from the service to be allowed.

Q volume X

Effect
☒ Allow ☐ Deny

List

<input type="checkbox"/> DescribeReplaceRootVolumeTasks Info	<input type="checkbox"/> DescribeVolumeAttribute Info	<input checked="" type="checkbox"/> DescribeVolumes Info
<input type="checkbox"/> DescribeVolumesModifications Info	<input type="checkbox"/> DescribeVolumeStatus Info	

Write

<input type="checkbox"/> AttachVolume Info	<input type="checkbox"/> CreateReplaceRootVolumeTask Info	<input type="checkbox"/> CreateVolume Info
<input type="checkbox"/> DeleteVolume Info	<input type="checkbox"/> DetachVolume Info	<input type="checkbox"/> EnableVolumeIO Info
<input type="checkbox"/> ImportVolume Info	<input checked="" type="checkbox"/> ModifyVolume Info	<input type="checkbox"/> ModifyVolumeAttribute Info
<input type="checkbox"/> PauseVolumeIO Info		

Step 2: Create a CloudWatch Event Rule

1. **Navigate to CloudWatch:**
 - Go to the CloudWatch console.
 - Select "Rules" under the "Events" section.
2. **Create a New Rule:**
 - Click "Create rule".
 - Configure the rule with the following settings:
 - **Event Source:** Select "AWS events".
 - **Service Name:** Choose "EC2".
 - **Event Type:** Choose "EBS Volume Notification".
 - **Specific Event:** Select "CreateVolume".

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](#)

Build event pattern Info

Event source

Select the event source from which events are sent.

- ☒ **AWS events or EventBridge partner events**
Events sent from AWS services or EventBridge partners.
- ☐ **Other**
Custom events or events sent from more than one source, e.g. events from AWS services and partners.
- ☐ **All events**
All events sent to your account.

Sample event - optional

You don't have to select or enter a sample event, but it's recommended so you can reference it when writing and testing the event pattern, or filter criteria.

You can reference the sample event when you write the event pattern, or use the sample event to test if it matches the event pattern. Find a sample event, enter your own, or edit a sample event below. [Learn more about the required fields in a sample event.](#)

Sample event type

- ☒ **AWS events**
- ☐ EventBridge partner events
- ☐ Enter my own

Sample events

Filter by event source and type or by keyword.

Select

Creation method

Method

- ☐ Use schema
Use an Amazon EventBridge schema to generate the event pattern.
- ☒ **Use pattern form**
Use a template provided by EventBridge to create an event pattern.
- ☐ Custom pattern (JSON editor)
Write an event pattern in JSON.

Event pattern Info

Event source

AWS service or EventBridge partner as source

AWS services

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

EBS Volume Notification

Event Type Specification 1

- ☐ Any event
- ☒ **Specific event(s)**

Specific event(s)

createVolume

Event Type Specification 2

- ☒ **Any volume ARN**
- ☐ Specific volume ARN(s)

Event pattern

Event pattern, or filter to match the events

```
1 {
2   "source": ["aws.ec2"],
3   "detail-type": ["EBS Volume Notification"],
4   "detail": {
5     "event": ["createVolume"]
6   }
7 }
```

Copy Test pattern Edit pattern

Cancel Previous **Next**

3. Add Target:

- Add a target to trigger the Lambda function.
- Select "Lambda function" and choose the function you will create in the next step.

Note: When using the EventBridge console, EventBridge will automatically configure the proper permissions for the selected targets. If you're using the AWS CLI, SDK, or CloudFormation, you'll need to configure the proper permissions.

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
ebs_volume ▼ ↻

► Configure version/alias

► Additional settings

[Add another target](#)
[Cancel](#)
[Skip to Review and create](#)
[Previous](#)
[Next](#)

Step 3: Create the Lambda Function

1. **Navigate to Lambda:**
 - Go to the Lambda console.
 - Click "Create function".
2. **Configure the Function:**
 - Name: Convert GP2 To GP3.
 - Runtime: Choose a suitable runtime, e.g., Python 3.8.
 - Role: Choose the IAM role created in Step 1.
3. **Add Code:**
 - Use the following Python code as an example to describe the volume and modify its type:

```
import json
```

```
import boto3
```

```
def convert_volume_to_gp3(volume_arn):
```

```
    ec2_client = boto3.client('ec2')
```

```
    arn_parts = volume_arn.split(':')
```

```
    volume_id = arn_parts[-1].split('/')[-1]
```

```
    return volume_id
```

```
def lambda_handler(event, context):

    volume_arn = event['resources'][0]

    volume_id = convert_volume_to_gp3(volume_arn)

    ec2_client = boto3.client('ec2')

    response = ec2_client.modify_volume(

        VolumeId=volume_id,

        VolumeType = 'gp3',

    )
```

4. **Deploy the Function:**
 - Save and deploy the Lambda function.

Step 4: Test the Setup

1. **Create a New GP2 Volume:**
 - Go to the EC2 console and create a new EBS volume of type GP2.
 - Ensure the volume creation triggers the CloudWatch Event and Lambda function.
2. **Verify Conversion:**
 - Check the Lambda function logs in CloudWatch Logs to confirm the volume type was changed from GP2 to GP3.

Conclusion

This setup ensures that any newly created EBS volume of type GP2 is automatically converted to GP3, maintaining compliance with organizational policies. Adjust the Lambda function and CloudWatch Event rules as needed to fit your specific requirements.