Link: https://repost.aws/knowledge-center/cloudwatch-receive-sns-for-alarm-trigger

Why didn't I receive an SNS notification for my CloudWatch alarm trigger?

I created an Amazon CloudWatch alarm to send notifications for an Amazon Simple Notification Service (Amazon SNS) topic when the alarm's state changes. However, I don't receive an SNS notification when the CloudWatch alarm changes states.

Resolution

SNS notifications delivery depends on the configuration of the SNS topic and the CloudWatch alarm. To determine why you don't receive SNS notifications, check the CloudWatch alarm's history to see the trigger action's status.

If your trigger action fails because of SNS access policy restrictions, then the CloudWatch alarm history displays a message similar to the following one:

"Failed to execute action arn:aws:sns:us-east-1:ACCOUNT_ID:TOPIC_NAME. Received error: "Resource: arn:aws:cloudwatch:us-east-1:ACCOUNT_ID:alarm:ALARM_NAME is not authorized to perform: SNS:Publish on resource: arn:aws:sns:us-east-1:ACCOUNT_ID:TOPIC_NAME""

SNS uses access policies to restrict the sources that can publish messages to the topic. If a permissions error occurs, then add the following permissions under the **Statement** section of the SNS access policy. The policy update grants permissions to the CloudWatch alarms service to publish messages to the SNS topic:

```
{
  "Sid": "Allow_Publish_Alarms",
  "Effect": "Allow",
  "Principal": {
    "Service": [
      "cloudwatch.amazonaws.com"
    ]
  },
  "Action": "sns:Publish",
  "Resource": "arn:aws:sns:us-east-1:ACCOUNT_ID:TOPIC_NAME"
}
```

Note: Replace **us-east-1** with the AWS Region that the notification is for, **ACCOUNT_ID** with your account ID, and **TOPIC_NAME** with the SNS topic name. To restrict the ability to publish messages to the topic for specific alarms, add global condition keys. The following example policy uses the ArnLike condition operator and the aws:SourceArn global condition key. For more information, see Example cases for Amazon SNS access control.

```
{
  "Sid": "Allow_Publish_Alarms",
  "Effect": "Allow",
  "Principal": {
    "Service": [
      "cloudwatch.amazonaws.com"
    ]
  },
  "Action": "sns:Publish",
  "Resource": "arn:aws:sns:REGION:ACCOUNT_ID:TOPIC_NAME",
  "Condition": {
    "ArnLike": {
      "aws:SourceArn": "arn:aws:cloudwatch:us-east-1:ACCOUNT_ID:alarm:ALARM_NAME"
    }
  }
}
```

Note: Replace **us-east-1** with the Region that the notification is for, **ACCOUNT_ID** with your account ID, **TOPIC_NAME** with the SNS topic name, and **ALARM_NAME** with the alarm name.

Important: Any alarm in the account that's included in the condition can publish to the resource's SNS topic in the policy. For example, the account ID of the alarm resource owner can publish to the topic. Restrict the policy to the same account for both the SNS topic account ID and the account ID that owns the alarm.

If your trigger action fails because of SNS topic encryption, then the CloudWatch alarm history displays a message similar to the following one:

"Failed to execute action arn:aws:sns:us-east-1:ACCOUNT_ID:TOPIC_NAME. Received error: "null (Service: AWSKMS; Status Code: 400; Error Code: AccessDeniedException;)""

SNS allows encryption at rest for its topic. If SNS uses the default AWS Key Management Service (AWS KMS) **alias/aws/sns** key for the encryption, then CloudWatch alarms can't publish to the SNS topic. The default AWS KMS key's policy for SNS doesn't allow CloudWatch alarms to perform **kms:Decrypt** and **kms:GenerateDataKey** API calls. Because AWS manages this key, you can't manually edit the policy.

If you must encrypt the SNS topic at rest, then use a customer managed key. The customer managed key must include the following permissions under the **Statement** section of the key policy. These permissions allow the CloudWatch alarms to publish messages to encrypted SNS topics:

```
{
  "Sid": "Allow_CloudWatch_for_CMK",
  "Effect": "Allow",
  "Principal": {
    "Service": [
      "cloudwatch.amazonaws.com"
    ]
  },
  "Action": [
    "kms:Decrypt",
    "kms:GenerateDataKey*"
  ],
  "Resource": "*"
}
```

If your trigger action succeeds, then the CloudWatch alarm history displays a message similar to the following one:

"Successfully executed action arn:aws:sns:us-east-1:ACCOUNT_ID:TOPIC_NAME"

The preceding message means the CloudWatch alarm successfully published a message to the SNS topic. If the notification isn't delivered by SNS, then check the SNS topic and its metrics for any delivery failures. For more information, see How do I access Amazon SNS topic delivery logs for push notifications?

Note: CloudWatch doesn't test or validate the actions that you specify. CloudWatch also doesn't detect Amazon EC2 Auto Scaling or Amazon SNS errors that result when you try to invoke nonexistent actions. Make sure that your actions exist.

Related information

Using Amazon CloudWatch alarms

Encrypting messages published to Amazon SNS with AWS KMS