

Implementation Guide:

PagerDuty

November 2020

PagerDuty

Table of Contents

Foreword.....	3
Solution overview and features	4
Architecture diagram	4
Pre-requisites	5
Deployment and configuration steps	5
Getting started	5
Configuration: Set up PagerDuty profile.....	8
Deployment: AWS Control Tower.....	9
Deployment: PagerDuty.....	11
Validate PagerDuty integration.....	15
Best practices	16
Solution estimated pricing	16
FAQs	16
Additional resources	16
Partner contact information	17

Foreword

PagerDuty is an AWS Advanced Technology Partner with a DevOps competency. PagerDuty performs Digital Operations Management in the AWS Cloud. By implementing PagerDuty, you can mobilize the right individuals at the right time during any incident.

The purpose of this AWS Implementation Guide is to enable every AWS Marketplace customer to seamlessly activate, deploy and configure PagerDuty in the AWS Control Tower environment. Additionally, it enables them to take full advantage of the resources pre-configured by AWS Control Tower as part of the initialization.

Solution overview and features

The solution automatically configures new AWS accounts enrolled by AWS Control Tower with PagerDuty via Amazon SNS. This automation simplifies the provisioning process of PagerDuty with AWS Services in your AWS Control Tower multi-account environment. PagerDuty integrated with AWS Control Tower provides customers the ability to manage the alerts that come from many of the AWS Services. A list of AWS specific integrations can be found on the [PagerDuty-AWS Partner page](#).

PagerDuty with AWS Control Tower enables customers to effectively:

- **Provide the right information** – Ingest alerts from Amazon CloudWatch or AWS Security Hub through Amazon SNS and determining based on PagerDuty rules and policies what alerts need to be escalated.
- **For the right person** – Use PagerDuty's Service Directory to determine who needs to be notified for the specific incident
- **At the right time** – PagerDuty On-Call Schedules determines the on-call resource that needs to be notified
- **To Take the right action** – Use Amazon EventBridge or Amazon GuardDuty to initiate automated actions in your AWS Environment

Architecture diagram

The solution is deployed using AWS CloudFormation Stack Set and connects via Amazon SNS. The solution primarily addresses AWS Control Tower Guardrails status, AWS Config rules, and conformance packs compliance status across all account/regions in AWS Control Tower environment.

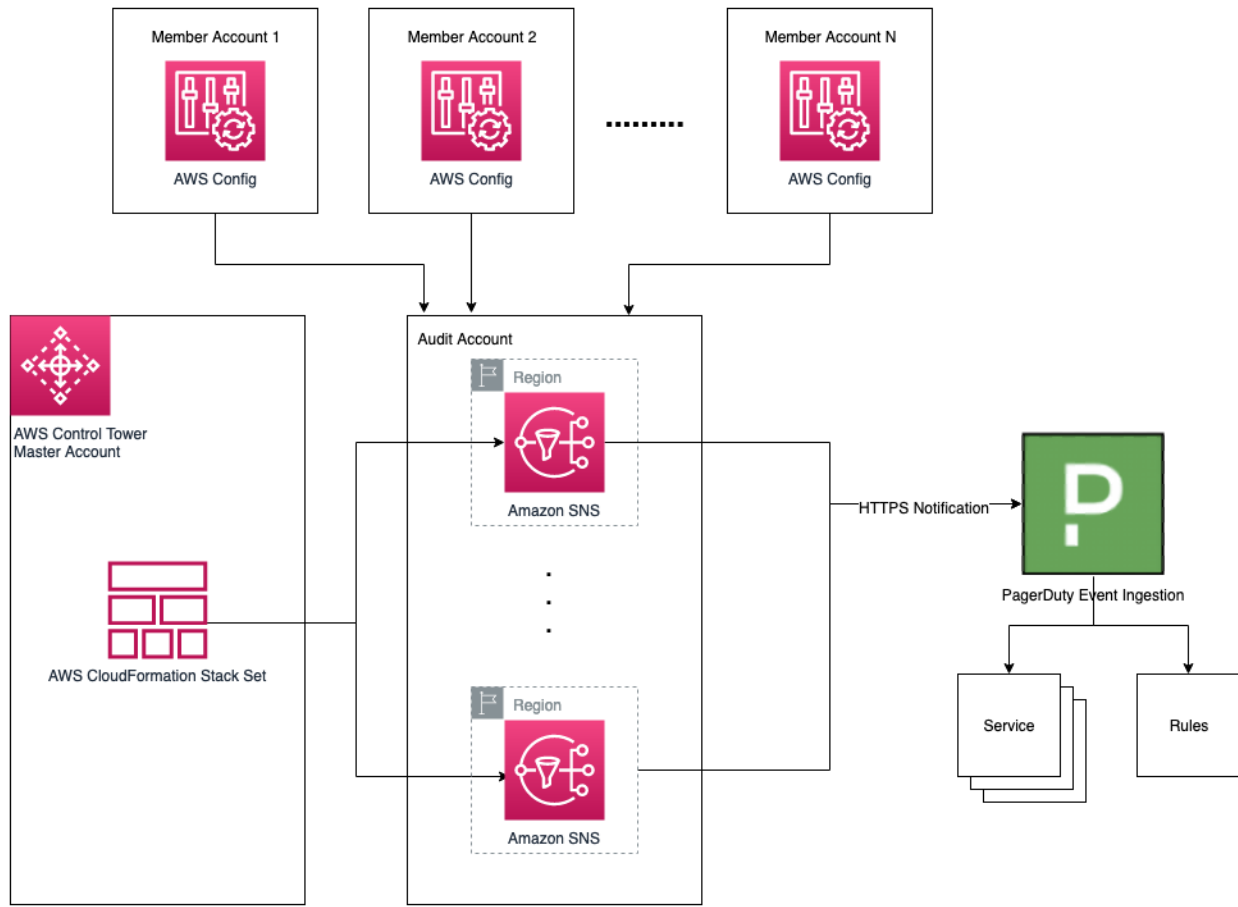


Figure 1. PagerDuty architecture diagram

Pre-requisites

This guide assumes you already have AWS Control Tower deployed. To get started with AWS Control Tower, check out the [Getting Started](#) documentation. If you don't have a PagerDuty subscription, please check the listing on [AWS Marketplace](#).

Deployment and configuration steps

Getting started

Step 1.1: Subscribe to PagerDuty on AWS Marketplace

While in your AWS Control Tower management account, locate the **PagerDuty** in the AWS Marketplace (<https://aws.amazon.com/marketplace/pp/B08HSJSKRG>).

aws marketplace

Categories ▾ Delivery Methods ▾ Solutions ▾ Migration Mapping Assistant Your Saved List Partners Sell in AWS Marketplace

PagerDuty
Sold by: Pagerduty, Inc.

PagerDuty helps teams to proactively mitigate customer-impacting issues by automatically turning any signal into the right insight and action so you can innovate and scale both your

Continue to Subscribe

Save to list

Overview Pricing Usage Support Reviews

Product Overview

PagerDuty's digital operations management platform empowers teams to proactively mitigate customer-impacting issues by automatically turning any signal into the right

Highlights

Click on the **Continue to Subscribe** button.

Step 1.2: Guidance on contract duration and renewal

In the new screen, you can configure your contract. You can select the **contract duration** and set the **renewal settings**.

PagerDuty

Configure your Software Contract

Choose the contract that suits your needs. You're charged for your purchase on your AWS bill. After you purchase a contract, you're directed to the vendor's site to complete setup and begin using this software. For any software use beyond your contract limit, you're charged consumption pricing.

How long do you want your contract to run?

12 months 24 months 36 months

Renewal Settings

Auto Renew when this contract ends on - Thu Oct 28 2021?

Yes No

I understand that when I renew, the seller's pricing terms and end user license agreement (EULA) might have changed. On the renewal date, I will be billed based on the price and EULA applicable on that date, which I can find on the Your Marketplace Software page.

Contract Options

Digital Operations	\$1188 / Units	<input type="text"/>
Everything you need for digital ops. \$99/user/month		

Create contract

By subscribing to this software, you agree to the pricing terms and the seller's [End User License Agreement \(EULA\)](#). You also agree and acknowledge that AWS may share information about this transaction (including your payment terms) with the respective seller, reseller or underlying provider, as applicable, in accordance with the [AWS Privacy Notice](#). Your use of AWS services is subject to the [AWS Customer Agreement](#) or other agreement with AWS governing your use of such services

Total Contract Price \$0

Due Today
Auto Renew - No

Select contract Option(s)

Step 1.3: Select contract options

Select the contract options to be activated with your contract.

Configure your Software Contract

Choose the contract that suits your needs. You're charged for your purchase on your AWS bill. After you purchase a contract, you're directed to the vendor's site to complete setup and begin using this software. For any software use beyond your contract limit, you're charged consumption pricing.

How long do you want your contract to run?

12 months 24 months 36 months

Renewal Settings

Auto Renew when this contract ends on - Thu Oct 28 2021?

Yes No

I understand that when I renew, the seller's pricing terms and end user license agreement (EULA) might have changed. On the renewal date, I will be billed based on the price and EULA applicable on that date, which I can find on the Your Marketplace Software page.

Create contract

By subscribing to this software, you agree to the pricing terms and the seller's [End User License Agreement \(EULA\)](#). You also agree and acknowledge that AWS may share information about this transaction (including your payment terms) with the respective seller, reseller or underlying provider, as applicable, in accordance with the [AWS Privacy Notice](#). Your use of AWS services is subject to the [AWS Customer Agreement](#) or other agreement with AWS governing your use of such services

Total Contract Price	\$4680.00
Due Today	
Auto Renew -No	
Business X 10 Units	\$4680.00

Step 1.4: Create the contract and pay

Once you have configured your contract, you can click on the **Create contract** button.

Configure your Software Contract

Choose the contract that suits your needs. You're charged for your purchase on

Create contract

You will be prompted to confirm the contract. If you agree to the pricing, select the **Pay now** button.

PagerDuty

You're purchasing the following contract. When you choose Pay now, your AWS account ID is invoiced. Payment is due when your AWS bill is due. If your usage exceeds your contract, additional usage costs apply.

PagerDuty - 12MONTHS Contract **\$4680.00 - Due Today**

Auto Renew - No

Step 1.5: Set up Account

Customer will be redirected to a PagerDuty Registration page to be completed with contact info, desired subdomain name, and password

Configuration: Set up PagerDuty profile

Step 2.1: Set up your user profile

- Video: [Setting up your user profile](#)
- Walkthrough: [Your user profile](#)

Step 2.2: Make sure your team has been added to PagerDuty

- Go to Configuration > Users and search for your team members
- If you don't see them listed, contact your Account Owner or a Global Admin to have your team members added (here's a video: [Adding Users to your Account](#)).

Step 2.3: Add your Users to an On-Call Schedule

- Video: [Creating a schedule](#)
- Video: [Copying a schedule to make a backup schedule](#)
- Walkthrough: [Schedule concepts](#)
- Walkthrough: [Schedule Examples](#)

Step 2.4: Add your On-Call Schedules to an Escalation Policy

- Video: [Creating an escalation policy](#)
- Walkthrough: [Create an escalation policy](#)

Step 2.5: Set up Services and Integrations

- Walkthrough: [Creating a PagerDuty service](#)
- Walkthrough: [Configuring services and integrations](#)
- We integrate with over 150 tools - view our [integration guides](#)

Step 2.5.1 Set up First Service

- Navigate to **Services** → **Service Directory** and click the **+New Service** button on the right side of the screen.
- Under General Settings, enter a **Name** that represents the application, component or team that you wish to open incidents against (examples: "MobileApp", "Shopping Cart" or "BizOps"). Please note that when an incident is triggered, this is the service name it will be associated with.

- Add a **Description** of what this service represents in your infrastructure.
- Then select which **Integration type** you will be using and enter an integration name in the format monitoring-tool-service-name (e.g. “Datadog-Shopping-Cart”).

Step 2.6: Test your Configuration

- [Trigger a test incident](#)
- Troubleshooting help? Contact support@pagerduty.com

Deployment: AWS Control Tower

Step 3.1: Obtain the ruleset’s integration key

In PagerDuty, choose Services...Event Rules from the menu.

On the next page, click the triangle to the left of “Incoming Event Source” to show the Integration information. On the first line labeled “Integration Key,” click the button “Copy to Clipboard.”

▼ Incoming Event Source

To use event rules, configure your monitoring tools and other integrations to send events using your new account-level integration key. Event rules are compatible with [200+ integrations](#), including:

splunk >

DATADOG

Nagios

amazon

New Relic

More...

Integrations will require either a PagerDuty integration key, an e-mail address, or an HTTP endpoint which are listed below.

Integration Key	W324VN2DHXZ3R8JVSAIBV3TM01IX4OWP	Copy to Clipboard
Email address	W324VN2DHXZ3R8JVSAIBV3TM01IX4OWP@subdomain.pagerduty.com	Copy to Clipboard
HTTP endpoint for API	https://events.pagerduty.com/v2/enqueue	Copy to Clipboard

The integration key must be part of the [API v2 payload](#)

Step 3.2: Deploy Solution in Management Account

Login to the management account in AWS Control Tower as an Administrator. Now [Launch](#) an AWS CloudFormation StackSet in the AWS Control Tower management account, providing a StackSet Name.

Step 3.3: Complete Set Up

- Launch this [CloudFormation Template](#)
- Replace your PagerDuty Integration Key from the one you copied in Step 3.1

SNS Subscription Configuration

AuditTopicName
 SNS Topic in Audit Account that AWS Config delivers notifications to.

Enter your PagerDuty HTTPS Endpoint
 https://events.pagerduty.com/x-ere/[YOUR_PAGERDUTY_INTEGRATION_KEY_HERE]

- Select **Self-Service Permissions**

Permissions

Choose an IAM role to explicitly define how CloudFormation will manage your target accounts. If you don't choose a role, CloudFormation uses permissions based on your user credentials. [Learn more](#)


Service managed permissions
 StackSets automatically configures the permissions required to deploy to target accounts managed by AWS Organizations. With this option, you can enable automatic deployment to accounts in your organization

Self service permissions
 You create the execution roles required to deploy to target accounts

- For **IAM role name**, choose **AWSControlTowerStackSetRole**
- For **IAM execution role name**, copy and paste this text: **AWSControlTowerExecution**. Choose **Next**

IAM admin role ARN - optional
 Choose the IAM role for CloudFormation to use for all operations performed on the stack.


IAM role na...

 StackSets will use this role for administering your individual accounts.

IAM execution role name

IAM execution role name can include letters (A-Z and a-z), numbers (0-9), and select special characters (+,=,@,-) characters. Maximum length is 64 characters.

- For **Account Numbers**, enter the AWS account ID for your AWS Control Tower audit account
- Under **Specify regions**, Select the AWS Control Tower supported regions by referring to this [link](#). Choose **Next**
- **Acknowledge** that AWS CloudFormation might create IAM resources

 **The following resource(s) require capabilities: [AWS::IAM::Role]**

This template contains Identity and Access Management (IAM) resources that might provide entities access to make changes to your AWS account. Check that you want to create each of these resources and that they have the minimum required permissions. [Learn more](#)

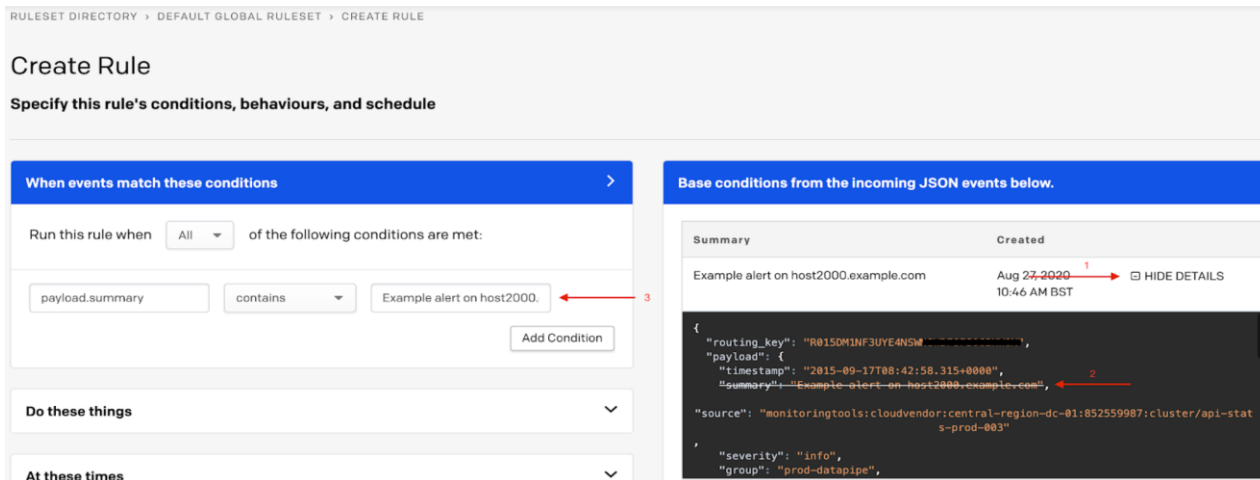
I acknowledge that AWS CloudFormation might create IAM resources.

- Choose **Submit** after validating your selections

Deployment: PagerDuty

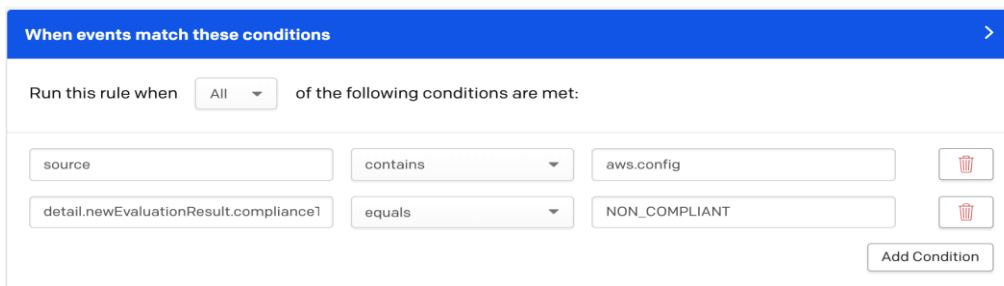
Step 4.1: Create a rule to initiate incidents

Step 4.1.1: Click the “New Event Rule” button.



Step 4.1.2: Set the following conditions:

- Create a condition where source contains aws.config
- Create a condition where detail.newEvaluationResult.complianceType equals NON_COMPLIANT



Step 4.1.3 Customize field events:

Step 4.1.3.1 Add a variable where:

- Name is resourceId
- Value is (.*)
- Regex is detail.newEvaluationResult.evaluationResultIdentifier.evaluationResultQualifier.resourceId

Step 4.1.3.2 Add a variable where:

- Name is ruleName
- Value is (.*)
- Regex is detail.newEvaluationResult.evaluationResultIdentifier.evaluationResultQualifier.configRuleName

Step 4.1.3.3 Add an event field where:

- Name is dedup_key

- Value is `{{resourceId}}-{{ruleName}}`
- Set the drop-down as Template

Step 4.1.3.4 Add an event field where:

- Name is summary
- Value is Rule `{{ruleName}}` on `{{resourceId}}` is now NON_COMPLIANT
- Set the drop-down as Template

Step 4.1.4: Set up:

Step 4.1.4.1: Click the radio button “Create an incident on a service”

Step 4.1.4.2: Choose a Service from the drop down under “Route to a Service” on the right-hand side.

Step 4.1.4.3: Click the Advanced tab on the right-hand side.

Step 4.1.4.4: Click the checkbox “Set a custom trigger / resolve action”

Step 4.1.4.5 Choose “Always trigger an alert”

Step 4.1.4.6: Click the button “Save Rule” at the bottom of the page.

Step 4.2: Create a rule to resolve incidents

Step 4.2.1: Click the “New Event Rule” button.

RULESET DIRECTORY > DEFAULT GLOBAL RULESET > CREATE RULE

Create Rule

Specify this rule's conditions, behaviours, and schedule

When events match these conditions

Run this rule when **All** of the following conditions are met:

payload.summary contains Example alert on host2000. **3**

Do these things

At these times

Base conditions from the incoming JSON events below.

Summary	Created
Example alert on host2000.example.com	Aug 27, 2020 10:46 AM BST 1

```
{  "routing_key": "R015DM1NF3UYE4NSW",  "payload": {    "timestamp": "2015-09-17T08:42:58.315+0000",    "summary": "Example alert on host2000.example.com", 2    "source": "monitoringtools:cloudvendor:central-region-dc-01:852559987:cluster/api-status-prod-003"  },  "severity": "info",  "group": "prod-datapipe"}
```

Step 4.2.2 Set the conditions:

Step 4.2.2.1 Create a condition where source contains aws.config

Step 4.2.2.2 Create a condition where detail.newEvaluationResult.complianceType equals COMPLIANT

When events match these conditions

Run this rule when **All** of the following conditions are met:

source contains aws.config

detail.newEvaluationResult.complianceType equals COMPLIANT

Add Condition

Step 4.2.3 Customize field events:

Step 4.2.3.1 Add a variable where:

- Name is resourceId
- Value is (.*)
- Regex is detail.newEvaluationResult.evaluationResultIdentifier.evaluationResultQualifier.resourceId

Step 4.2.3.2 Add a variable where:

- Name is ruleName
- Value is (.*)
- Regex is detail.newEvaluationResult.evaluationResultIdentifier.evaluationResultQualifier.configRuleName

Step 4.2.3.3 Add an event field where:

- Name is dedup_key

- Value is `{{resourceId}}-{{ruleName}}`
- Set the drop-down as Template

Step 4.2.3.4 Add an event field where:

- Name is summary
- Value is Rule `{{ruleName}}` on `{{resourceId}}` is now NON_COMPLIANT
- Set the drop-down as Template

Step 4.2.4: Set up:

Step 4.2.4.1: Click the radio button “Suppress Alert”

Step 4.2.4.2: Choose a Service from the drop down under “Route to a Service” on the right-hand side.

Step 4.2.4.3: Click the Advanced tab on the right-hand side.

Step 4.2.4.4: Click the checkbox “Set a custom trigger / resolve action”

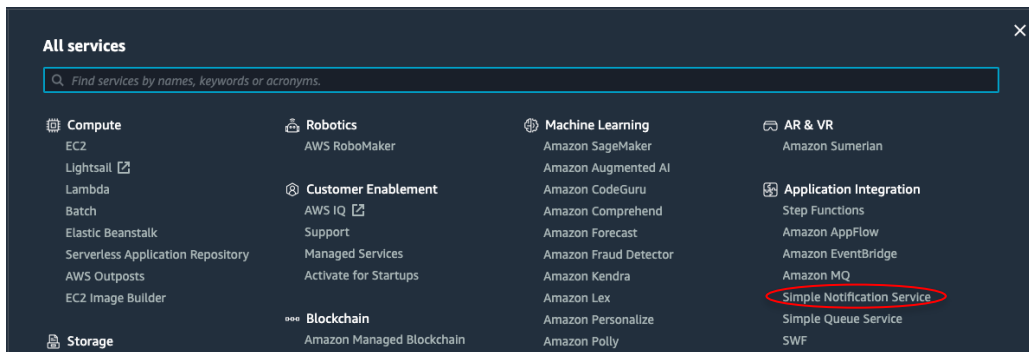
Step 4.2.4.5 Choose “Always resolve an alert”

Step 4.2.4.6: Click the button “Save Rule” at the bottom of the page.

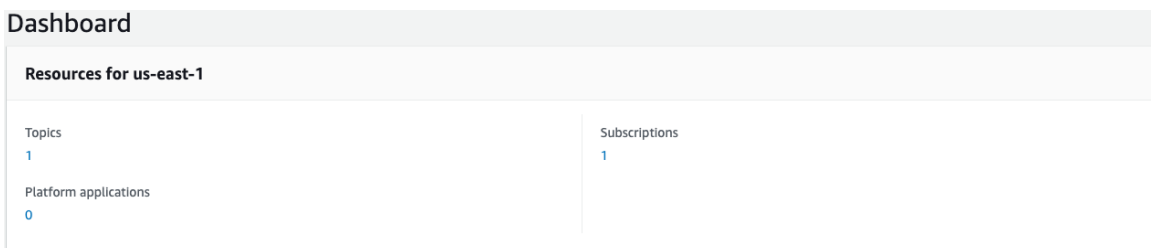
You should now have two rules: one to trigger when a rule indicates that a resource is not in compliance and one to resolve that same incident when it comes back into compliance.

Validate PagerDuty Integration

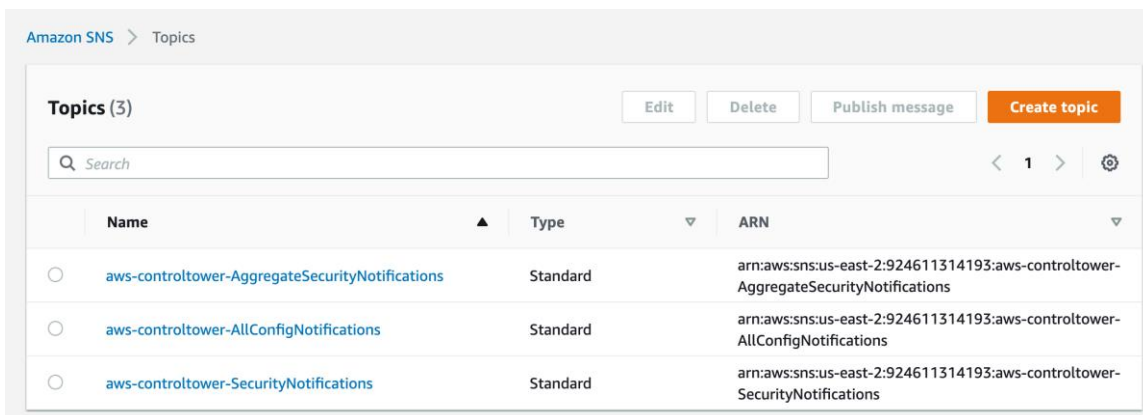
Once you have completed the steps, please validate that the integration is working by logging into the AWS management account and validate that a topic has been created in Amazon SNS. Start by choosing the Simple Notification Service option from the service menu:



Click on the number below Topics in the Dashboard:



Validate that the topics have been created:



Click on the Topics and ensure status is confirmed :

Amazon SNS > Topics > aws-controltower-AggregateSecurityNotifications

aws-controltower-AggregateSecurityNotifications

Edit Delete Publish message

Details

Name aws-controltower-AggregateSecurityNotifications	Display name aws-controltower-AggregateSecurityNotifications
ARN arn:aws:sns:us-east-2:924611314193:aws-controltower-AggregateSecurityNotifications	Topic owner 924611314193
Type Standard	

Subscriptions Access policy Delivery retry policy (HTTP/S) Delivery status logging Encryption Tags

Subscriptions (5) Edit Delete Request confirmation Confirm subscription Create subscription

Q pagerduty X < 1 > ⚙

ID	Endpoint	Status	Protocol
55692d09-4462-440f-9332-e07b9b2e7695	https://events.pagerduty.com/x-ere/R017240BGF7HG80IFRJT06WM510D22BL	Confirmed	HTTPS

Best Practices

[PagerDuty User Roles & Permissions](#)

[PagerDuty Account Subdomains](#)

[Retrieving Incident Details in PagerDuty](#)

Solution estimated pricing

Pricing detail is available on [AWS Marketplace](#)

FAQs

FAQs on [PagerDuty.com](#)

Additional resources

[Getting Started with PagerDuty on AWS Marketplace](#)

[Amazon CloudWatch Integration Guide | PagerDuty](#)

[Amazon EventBridge Integration Guide | PagerDuty](#)

[Amazon GuardDuty Integration Guide | PagerDuty](#)

[AWS CloudTrail Integration Guide | PagerDuty](#)

[AWS Personal Health Dashboard Integration Guide | PagerDuty](#)

[AWS Security Hub Integration Guide | PagerDuty](#)

Partner contact information

support@pagerduty.com

<https://support.pagerduty.com>