



AWS Incident Detection and Response Concepts and Procedures

AWS Incident Detection and Response User Guide



Version May 12, 2026

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

AWS Incident Detection and Response User Guide: AWS Incident Detection and Response Concepts and Procedures

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon's trademarks and trade dress may not be used in connection with any product or service that is not Amazon's, in any manner that is likely to cause confusion among customers, or in any manner that disparages or discredits Amazon. All other trademarks not owned by Amazon are the property of their respective owners, who may or may not be affiliated with, connected to, or sponsored by Amazon.

Table of Contents

What is AWS Incident Detection and Response?	1
Terms of use	1
Architecture	2
Roles and responsibilities	3
Region availability	6
Get started	8
About workloads	8
About alarms	8
Onboard workloads	9
Onboard with the IDR CLI	9
Alarm Ingestion	10
Steps for alarm ingestion	10
Alternative options for ingesting alarms	11
Provision access	11
Alarm definition	12
Alarm optimization	33
Alarm review	33
Alarm testing	34
Alarms go live	35
Onboarding questionnaires (exception path)	36
Workload onboarding questionnaire - General questions	37
Workload onboarding questionnaire - Architecture questions	37
Alarm ingestion questionnaire - Overview	39
Alarm ingestion questionnaire - Runbook questions	40
Alarm matrix	41
Manage workloads	46
Develop runbooks and response plans	46
Test onboarded workloads	53
CloudWatch alarms	34
Third party APM alarms	35
Key outputs	35
Request changes to a workload	55
Suppress alarms	58
Suppress alarms at the alarm source	58

Submit a workload change request to suppress alarms	63
Tutorial: Use a metric math function to suppress an alarm	64
Tutorial: Remove a metric math function to un-suppress an alarm	67
Offboard a workload	68
Monitoring and observability	70
Implementing observability	71
Incident management	72
Provision access for application teams	75
Request an Incident Response	75
Request through the AWS Support Center Console	76
Request through the AWS Support API	77
Request through the AWS Support App in Slack	77
Manage Incident Detection and Response support cases with the AWS Support App in Slack	83
Alarm-initiated incident notifications in Slack	84
Create an Incident Response Request in Slack	87
Reporting	88
Security and resiliency	89
Access to your accounts	90
Your alarm data	90
Document history	91

What is AWS Incident Detection and Response?

AWS Incident Detection and Response offers eligible AWS Enterprise Support customers proactive incident engagement to reduce the potential for failure and accelerate recovery of critical workloads from disruption. Incident Detection and Response facilitates your collaboration with AWS to develop runbooks and response plans customized to each onboarded workload.

Incident Detection and Response offers the following key features:

- **Improved observability:** AWS experts provide guidance to help you define and correlate metrics and alarms between the application and infrastructure layers of your workload to detect disruptions early.
- **5-minute response time:** Incident Management Engineers proactively engage you within 5 minutes of an alarm, from your workloads, or in response to a critical case that you submit.
- **Faster resolution:** IMEs use pre-defined and custom runbooks developed for your workloads, create a Support case on your behalf, and manage incidents on your workload. IMEs provide single-threaded ownership for incidents and keep you engaged with the right AWS experts until the incident is resolved.
- **Reduced potential for failure:** After resolution, the IMEs provide you with a post-incident review (upon request). And, AWS experts work with you to apply lessons learned to improve the incident response plan and runbooks. You can also leverage AWS Resilience Hub for continuous resiliency tracking on your workloads.

Topics

- [Terms of use for Incident Detection and Response](#)
- [Architecture of Incident Detection and Response](#)
- [Roles and responsibilities in Incident Detection and Response](#)
- [Region availability for Incident Detection and Response](#)

Terms of use for Incident Detection and Response

The following list outlines the key requirements and limitations for using AWS Incident Detection and Response. This information is important for you to understand before using the service, as it covers aspects like support plan requirements, onboarding process, and minimum subscription duration.

- AWS Incident Detection and Response is available to direct and Partner-resold Enterprise Support accounts.
- AWS Incident Detection and Response is not available to accounts on Partner Led Support.
- You must maintain AWS Enterprise Support at all times during the term of your Incident Detection and Response service. For information, see [Enterprise Support](#). Termination of Enterprise Support results in concurrent removal from the AWS Incident Detection and Response service.
- All workloads on AWS Incident Detection and Response must go through the workload onboarding process.
- The minimum duration to subscribe an account to AWS Incident Detection and Response is ninety (90) days. All cancellation requests must be submitted thirty (30) days prior to the intended effective date of cancellation.
- AWS handles your information as described in the [AWS Privacy Notice](#).

Note

For Incident Detection and Response billing related questions, see [Getting help with AWS Billing](#).

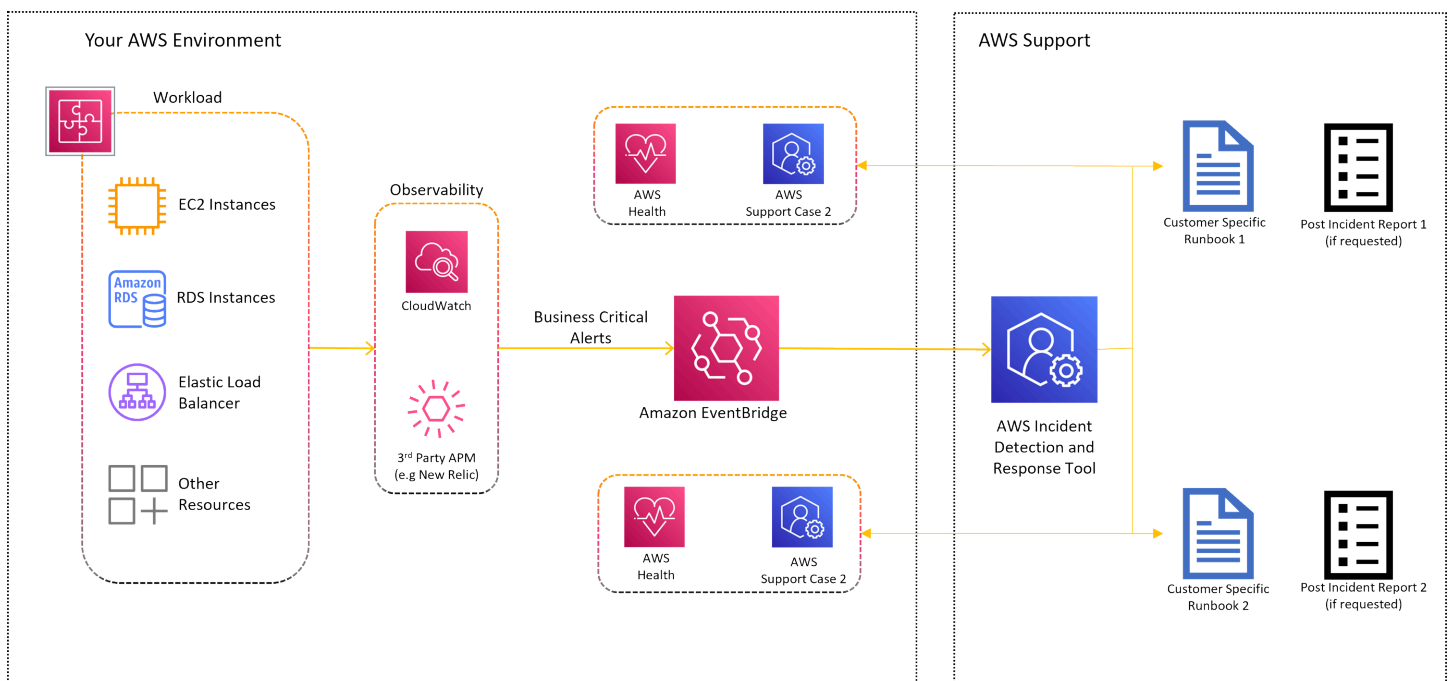
Architecture of Incident Detection and Response

AWS Incident Detection and Response integrates with your existing environment as shown in the following graphic. The architecture includes the following services:

- Amazon EventBridge: Amazon EventBridge serves as the sole integration point between your workloads and AWS Incident Detection and Response. Alarms are ingested from your monitoring tools, such as Amazon CloudWatch, through Amazon EventBridge using predefined rules managed by AWS. To allow Incident Detection and Response to build and manage the EventBridge rule, you install a service-linked role. To learn more about these services, see [What is Amazon EventBridge](#) and [Amazon EventBridge rules](#), [What is Amazon CloudWatch](#), and [Using service-linked roles for AWS Health](#).
- AWS Health: AWS Health provides ongoing visibility into your resource performance and the availability of your AWS services and accounts. Incident Detection and Response uses AWS Health to track events on the AWS services used by your workloads and to notify you when an

alert has been received from your workload. To learn more about AWS Health, see [What is AWS Health](#).

- **AWS Systems Manager:** Systems Manager provides a unified user interface for automation and task management across your AWS resources. AWS Incident Detection and Response hosts information about your workloads including workload architecture details, alarm details and their corresponding incident management runbooks in AWS Systems Manager documents (for details, see [AWS Systems Manager Documents](#)). To learn more about AWS Systems Manager, see [What is AWS Systems Manager](#).
- **Your specific runbooks:** An incident management runbook defines the actions that AWS Incident Detection and Response performs during incident management. Your specific runbooks tell AWS Incident Detection and Response who to contact, how to contact them, and what information to share.



Roles and responsibilities in Incident Detection and Response

The AWS Incident Detection and Response RACI (Responsible, Accountable, Consulted, and Informed) table outlines the roles and responsibilities for various activities related to incident detection and response. This table helps define the involvement of the customer and the AWS Incident Detection and Response team for tasks such as data collection, operations readiness review, account configuration, incident management, and post-incident review.

Activity	Customer	Incident Detection and Response
Data collection		
Customer and workload introduction	Consulted	Responsible
Architecture	Responsible	Accountable
Operations	Responsible	Accountable
Determine CloudWatch alarms to be configured	Responsible	Accountable
Define incident response plan	Responsible	Accountable
Operations readiness review		
Conduct well architected review (WAR) on workload	Consulted	Responsible
Validate incident response	Consulted	Responsible
Validate alarm matrix	Consulted	Responsible
Identify key AWS services being used by the workload	Accountable	Responsible
Account configuration		

Activity	Customer	Incident Detection and Response
Create IAM role in customer account	Responsible	Informed
Install managed EventBridge rule using created role	Informed	Responsible
Test CloudWatch alarms	Responsible	Accountable
Verify that customer alarms engage the incident detection and response	Informed	Responsible
Update alarms	Responsible	Consulted
Update runbooks	Consulted	Responsible
Incident management		
Proactively notify Incidents detected by Incident Detection and Response	Informed	Responsible
Provide incident response	Informed	Responsible
Provide incident resolution / infrastructure restore	Responsible	Consulted
Post-incident review		
Request post-incident review	Responsible	Informed

Activity	Customer	Incident Detection and Response
Provide post-incident review	Informed	Responsible

Region availability for Incident Detection and Response

AWS Incident Detection and Response is available in English, Japanese, Mandarin, and Korean for AWS Enterprise Support accounts hosted in any of the following AWS Regions:

AWS Region	Name
US East (N. Virginia) Region	us-east-1
US East (Ohio) Region	us-east-2
US West (N. California) Region	us-west-1
US West (Oregon) Region	us-west-2
Canada (Central) Region	ca-central-1
Canada West (Calgary) Region	ca-west-1
South America (São Paulo) Region	sa-east-1
Europe (Frankfurt) Region	eu-central-1
Europe (Ireland) Region	eu-west-1
Europe (London) Region	eu-west-2
Europe (Paris) Region	eu-west-3
Europe (Stockholm) Region	eu-north-1

AWS Region	Name
Europe (Zurich) Region	eu-central-2
Europe (Milan) Region	eu-south-1
Europe (Spain) Region	eu-south-2
Asia Pacific (Mumbai)	ap-south-1
Asia Pacific (Tokyo)	ap-northeast-1
Asia Pacific (Seoul)	ap-northeast-2
Asia Pacific (Singapore)	ap-southeast-1
Asia Pacific (Sydney)	ap-southeast-2
Asia Pacific (Hong Kong)	ap-east-1
Asia Pacific (Osaka)	ap-northeast-3
Asia Pacific (Hyderabad)	ap-south-2
Asia Pacific (Jakarta)	ap-southeast-3
Asia Pacific (Melbourne)	ap-southeast-4
Asia Pacific (Malaysia)	ap-southeast-5
Africa (Cape Town)	af-south-1
Israel (Tel Aviv)	il-central-1
Middle East (UAE)	me-central-1
Middle East (Bahrain)	me-south-1
AWS GovCloud (US-East)	us-gov-east-1
AWS GovCloud (US-West)	us-gov-west-1

Get started with Incident Detection and Response

Workloads and alarms are central to AWS Incident Detection and Response. AWS works closely with you to define and monitor specific workloads that are critical to your business. AWS helps you set up alarms that notify your team of significant performance issues or customer impact. Properly configured alarms are essential for proactive monitoring and rapid incident response within Incident Detection and Response.

About workloads in Incident Detection and Response

You can select specific workloads for monitoring and critical incident management using AWS Incident Detection and Response. A workload is a collection of resources and code that work together to deliver business value. A workload might be all the resources and code that make up your banking payment portal or a customer relationship management (CRM) system. You can host a workload in a single AWS account or multiple AWS accounts.

For example, you might have a monolithic application hosted in a single account (for example, Employee Performance App in the following diagram). Or, you might have an application (for example, Storefront Webapp in the diagram) broken into microservices that stretch across different accounts. A workload might share resources, such as a database, with other applications or workloads, as shown in the following diagram.

To get started with workload onboarding, see [Onboard workloads to Incident Detection and Response](#).

About alarms in Incident Detection and Response

Alarms are a key part of Incident Detection and Response. Alarms provide visibility into the performance of your applications and underlying AWS infrastructure. AWS works with you to define appropriate metrics and alarm thresholds that only trigger when there is critical impact to your monitored workloads. The goal is for alarms to engage your specified resolvers, who then collaborate with the incident management team to quickly mitigate issues. Configure your alarms to only enter the **Alarm** state when there is a significant degradation in performance or customer experience that requires immediate attention. Some key types of alarms include those that indicate business impact, Amazon CloudWatch canaries, and aggregate alarms that monitor dependencies.

To get started with alarm ingestion, see [Alarm Ingestion](#).

Onboard workloads to Incident Detection and Response

AWS Incident Detection and Response enables monitoring and critical incident management for your selected workloads. A workload is a collection of resources working together to deliver business value, such as a payment portal or a customer relationship management (CRM) system. You can host these workloads in either a single AWS account or distributed across multiple accounts, depending on your architecture.

Contents

- [Onboard to Incident Detection and Response with the IDR CLI](#)
 - [Language support for the IDR CLI](#)
 - [Alternative options for onboarding workloads](#)

Onboard to Incident Detection and Response with the IDR CLI

The AWS Incident Detection and Response Customer Command Line Interface (IDR CLI) is a command line interface tool that streamlines onboarding to AWS Incident Detection and Response.

The IDR CLI runs in AWS CloudShell to perform the following functions:

- Collect onboarding information
- Gather AWS resource data through the Resource Groups Tagging API
- Manage AWS Support cases
- Create new Amazon CloudWatch alarms or ingest your existing ones
- Deploy and test infrastructure through AWS CloudFormation to allow third-party tools to send alerts to Incident Detection and Response.

The IDR CLI can run in an **interactive mode** to guide you through the onboarding steps, or in **offline mode** for bulk or DevOps use cases.

For more information on how to use the IDR CLI, including installation, prerequisites, and end-to-end examples, see [CLI for AWS Incident Detection and Response](#).

Language support for the IDR CLI

AWS Incident Detection and Response is available in English, Japanese, Mandarin, and Korean. If you need support in Japanese, Mandarin, or Korean, contact AWS through the AWS Support case created by the IDR CLI, or contact your Technical Account Manager (TAM).

Alternative options for onboarding workloads

If you can't use the IDR CLI for onboarding, consult your Technical Account Manager (TAM) for alternative options. For more information, see [Workload onboarding and alarm ingestion questionnaires in Incident Detection and Response \(exception path\)](#)

Alarm Ingestion

The AWS Incident Detection and Response Customer Command Line Interface (IDR CLI) can create new Amazon CloudWatch alarms or ingest your existing ones and can deploy and test infrastructure through AWS CloudFormation to allow third-party tools to send alerts to AWS Incident Detection and Response.

AWS Incident Detection and Response can ingest alarms from Amazon CloudWatch and third party Application Performance Monitoring (APM) tools via Amazon EventBridge:

- [Ingesting CloudWatch alarms](#)
- [Ingesting Third Party Application Performance Monitoring Alarms](#)

Steps for alarm ingestion

The following steps need to be completed for alarm ingestion:

- [Alarm definition](#)
- [Alarm ingestion using the IDR CLI](#)
- [Alarm review and feedback](#)
- [Provision access for alarm ingestion to Incident Detection and Response](#)
- [Alarm testing \(Gameday\)](#)
- Alarms are enabled for active monitoring by AWS Incident Detection and Response after the preceding steps are complete.

Alternative options for ingesting alarms

If you can't use the IDR CLI for alarm ingestion, consult your Technical Account Manager (TAM) for alternative options. For more information, see [Workload onboarding and alarm ingestion questionnaires in Incident Detection and Response \(exception path\)](#)

Provision access for alarm ingestion to Incident Detection and Response

Note

If you didn't create the service-linked role (SLR) during the IDR CLI onboarding, follow the steps below to manually provision access.

To allow AWS Incident Detection and Response to ingest alarms from your account, create the `AWSServiceRoleForHealth_EventProcessor` SLR. AWS assumes the SLR to create a Managed EventBridge rule in your account. The managed EventBridge rule sends notifications from your account to AWS Incident Detection and Response. For information about this SLR, including the associated AWS managed policy, see [Using service-linked roles](#) in the *User Guide*.

You can create this service-linked role in your account by following the instructions in [Create service-linked role](#) in the *AWS Identity and Access Management User Guide*. Or, you can use the following AWS Command Line Interface (AWS CLI) command:

```
aws iam create-service-linked-role --aws-service-name event-processor.health.amazonaws.com
```

Key outputs

- Successful creation of the service-linked role in your account.

Note

The service-linked role - `AWSServiceRoleForHealth_EventProcessor` needs to be created in each account you will use to send alarms to AWS Incident Detection and Response.

Related information

For more information, see the following topics:

- [Using service-linked roles for](#)
- [Creating a service-linked role](#)
- [AWS managed policy: AWSHealth_EventProcessorServiceRolePolicy](#)

Alarm definition

When onboarding your alarms to AWS Incident Detection and Response, you're responsible for defining the metrics and alarm configurations that provide visibility into the performance of your applications. As part of this process, you must also identify the teams within your organization who is responsible for responding to these alarms.

When preparing alarms, we recommend the following best practices:

- Alarms only enter the "Alarm" state when there is ongoing critical impact to your monitored workload that requires immediate attention from your team and AWS. Alarms that trigger and don't automatically recover require your teams to join an incident bridge with AWS Incident Detection and Response.
- Ensure the contact information you provide allows AWS Incident Detection and Response to reliably engage the appropriate teams within your organization to an incident bridge 24/7.

Key outputs

- A list of alarms and contact details, which you provide to AWS Incident Detection and Response using the [IDR CLI](#).

For more information about defining and ingesting Amazon CloudWatch alarms see [Ingesting CloudWatch alarms](#).

For more information about ingesting third party Application Performance Monitoring alarms see [Ingesting Third Party Application Performance Monitoring Alarms](#).

Ingesting CloudWatch alarms

AWS Incident Detection and Response can ingest Amazon CloudWatch alarms to provide proactive monitoring for your critical workloads. By ingesting your Amazon CloudWatch alarms for monitoring, AWS Incident Detection and Response can:

- Automatically detect when your alarms enter the "Alarm" state.
- Engage your teams to collaboratively respond and resolve incidents.

To ensure the alarms you onboard are effective, AWS Incident Detection and Response recommends the following best practices:

- Configure alarms with [metric math expressions](#) to suppress them during periods of regular maintenance or batch job executions to avoid false positive alarm engagements.
- Set the Missing Data Treatment on alarms based on the expected datapoint delivery frequency. For example, alarms monitoring metrics that generate a continuous stream of datapoints should treat missing data as "Breaching" (bad) as missing datapoints could indicate an issue with the underlying resource monitored. Inversely, alarms monitoring metrics that infrequently report datapoints, for example alarms monitoring metrics that only record datapoints when a failure or error occurs, should treat missing data as NotBreaching (good).
- Define alarms that enter the "Alarm" state when there is critical, ongoing impact to your workload. For example, configure alarms to trigger after the expected time required to automatically replace unhealthy resources, rather than on the initial detection of unhealthy resources.
- Identify and create alarms for [custom metrics](#) that directly represent the customer experience for your workload.

For a list of recommended Amazon CloudWatch alarms for common AWS services, see the [Incident Detection and Response Alarm Best Practices on AWS re:Post](#).

Ingesting Third Party Application Performance Monitoring Alarms

AWS Incident Detection and Response supports alarm ingestion from third-party Application Performance Monitoring (APM) tools through Amazon EventBridge. This integration provides flexibility by ingesting APM alerts, allowing routing of APM events via various AWS services to an Amazon EventBridge event bus in your account.

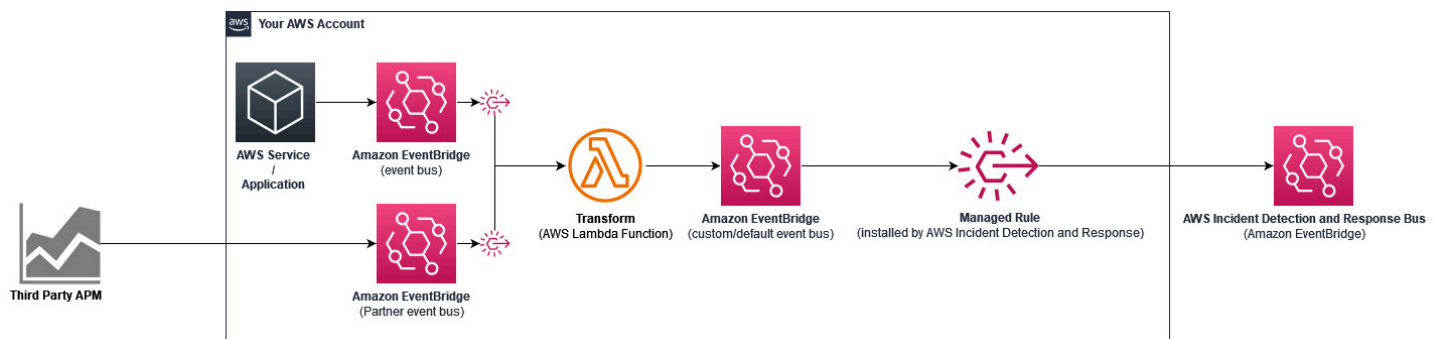
Integration path examples:

- Source (APM) → AWS Service (Example: Amazon API Gateway or Amazon SNS) → Transform Lambda Function → Custom Amazon EventBridge Event Bus → AWS Incident Detection and Response
- Source (APM) → Partner Amazon EventBridge Event Bus → Transform Lambda Function → Custom Amazon EventBridge Event Bus → AWS Incident Detection and Response

AWS Incident Detection and Response installs a managed rule on the custom event bus to ingest alerts sent to it by Transform Lambda Functions. It's important to note that for SaaS Amazon EventBridge Integrations, the partner event bus isn't the event bus that has a managed rule installed. For a complete list of APMs with partner integrations to Amazon EventBridge, see [Amazon EventBridge integrations](#).

Example integration using a partner event bus or other AWS event bus sources

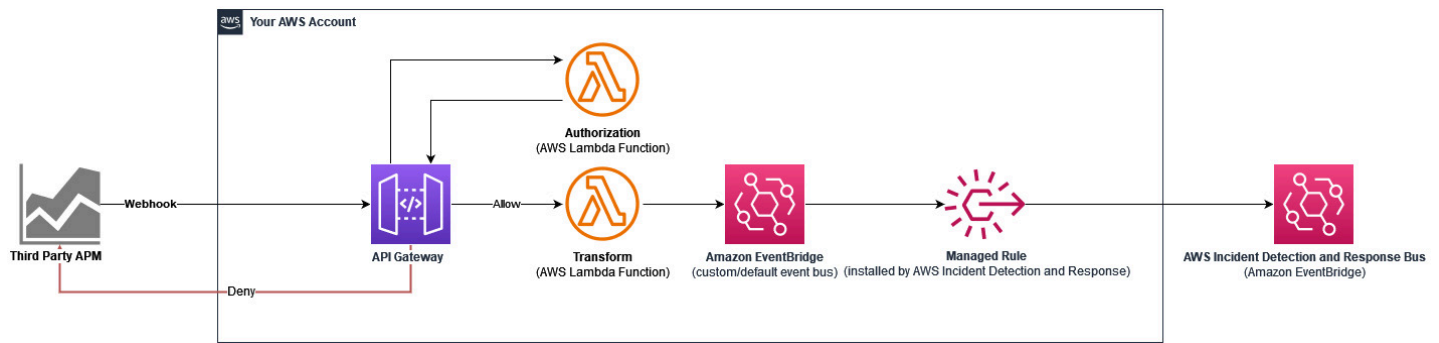
The following diagram shows an example integration using a partner event bus or other AWS event bus sources.



For a complete list of APMs with partner integrations to Amazon EventBridge, see [Amazon EventBridge integrations](#).

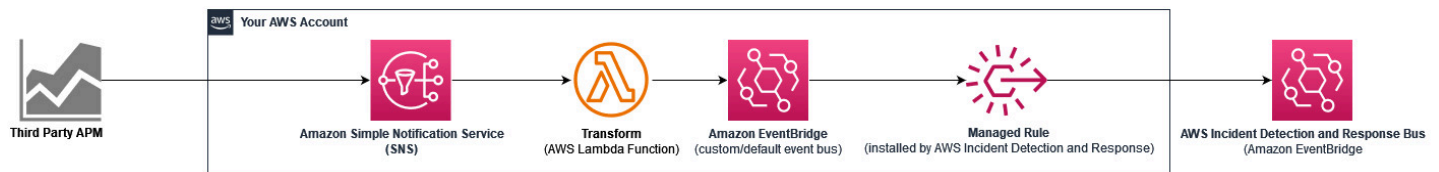
Example integration using Amazon API Gateway

The following diagram shows an example of integration using a API Gateway.



Example integration using Amazon Simple Notification Service

The following diagram shows an example of integration using a Amazon SNS.



To simplify the integration process, AWS Incident Detection and Response provides CloudFormation templates for the most commonly used integration types. These templates automate the setup of AWS resources, and necessary IAM roles.

CloudFormation Templates and instructions to manually create various integration types can be found in the corresponding integration documentation below:

- [Ingest Alarms from APMs with direct EventBridge integration](#)
- [Ingest alarms from APMs without direct integration with EventBridge](#)
- [Ingest Alarms from APMs with direct Amazon SNS integration](#)

Note

The CloudFormation templates require modifications. These modifications are explained in the preceding topics. For more information on the required payload format for sending APM alerts to AWS Incident Detection and Response see [Payload Requirements For Ingesting APM Alerts with EventBridge](#).

Payload Requirements For Ingesting APM Alerts with EventBridge

Where does Incident Detection and Response ingest APM alerts from?

AWS Incident Detection and Response installs a managed rule on the event bus that you send your final transformed payload to. It's a best practice to create a custom event bus for this purpose.

What format must payloads be in?

The following minimum JSON key:value pairs are required in event bus events ingested by AWS Incident Detection and Response:

```
{
  "detail-type": "ams.monitoring/generic-apm",
  "source": "GenericAPMEvent"
  "detail": {
    "incident-detection-response-identifier": "Your alarm name from your APM",
  }
}
```

The following examples show an event from a partner event bus before and after it is transformed.

Before transformation:

```
{
  "version": "0",
  "id": "a6150a80-601d-be41-1a1f-2c5527a99199",
  "detail-type": "Datadog Alert Notification",
  "source": "aws.partner/datadog.com/Datadog-aaa111bbbc",
  "account": "123456789012",
  "time": "2023-10-25T14:42:25Z",
  "region": "us-east-1",
  "resources": [],
  "detail": {
    "alert_type": "error",
    "event_type": "query_alert_monitor",
    "meta": {
      "monitor": {
        "id": 222222,
        "org_id": 3333333333,
        "type": "query alert",
        "name": "UnHealthyHostCount",
        "message": "@awseventbridge-Datadog-aaa111bbbc",
        "query":
"max(last_5m):avg:aws.applicationelb.un_healthy_host_count{aws_account:123456789012}
<= 1",
      }
    }
  }
}
```

```
    "created_at": 1686884769000,
    "modified": 1698244915000,
    "options": {
      "thresholds": {
        "critical": 1.0
      }
    },
  },
  "result": {
    "result_id": 7281010972796602670,
    "result_ts": 1698244878,
    "evaluation_ts": 1698244868,
    "scheduled_ts": 1698244938,
    "metadata": {
      "monitor_id": 222222,
      "metric": "aws.applicationelb.un_healthy_host_count"
    }
  },
  "transition": {
    "trans_name": "Triggered",
    "trans_type": "alert"
  },
  "states": {
    "source_state": "OK",
    "dest_state": "Alert"
  },
  "duration": 0
},
"priority": "normal",
"source_type_name": "Monitor Alert",
"tags": [
  "aws_account:123456789012",
  "monitor"
]
}
```

Note that before the event is transformed, `detail-type` and `source` indicates the APM details where the alert originated. These must be modified before ingestion. The `incident-detection-response-identifier` key is not yet present and must also be added before ingestion.

An Lambda Function transforms the above event and puts it in to the target custom or default event bus. The transformed payload must include the required key:value pairs.

After transformation:

```
{
  "version": "0",
  "id": "7f5e0fc1-e917-2b5d-a299-50f4735f1283",
  "detail-type": "ams.monitoring/generic-apm",
  "source": "GenericAPMEvent",
  "account": "123456789012",
  "time": "2023-10-25T14:42:25Z",
  "region": "us-east-1",
  "resources": [],
  "detail": {
    "incident-detection-response-identifier": "UnHealthyHostCount",
    "alert_type": "error",
    "event_type": "query_alert_monitor",
    "meta": {
      "monitor": {
        "id": 222222,
        "org_id": 3333333333,
        "type": "query alert",
        "name": "UnHealthyHostCount",
        "message": "@awseventbridge-Datadog-aaa111bbbc",
        "query":
"max(last_5m):avg:aws.applicationelb.un_healthy_host_count{aws_account:123456789012}
<= 1",
        "created_at": 1686884769000,
        "modified": 1698244915000,
        "options": {
          "thresholds": {
            "critical": 1.0
          }
        },
      },
    },
    "result": {
      "result_id": 7281010972796602670,
      "result_ts": 1698244878,
      "evaluation_ts": 1698244868,
      "scheduled_ts": 1698244938,
      "metadata": {
        "monitor_id": 222222,
        "metric": "aws.applicationelb.un_healthy_host_count"
      }
    },
  },
  "transition": {
```

```
        "trans_name": "Triggered",
        "trans_type": "alert"
    },
    "states": {
        "source_state": "OK",
        "dest_state": "Alert"
    },
    "duration": 0
},
"priority": "normal",
"source_type_name": "Monitor Alert",
"tags": [
    "aws_account:123456789012",
    "monitor"
]
}
```

Note that `detail-type` is now `ams.monitoring/generic-apm`, `source` is now `GenericAPMEvent`, and under `detail` there is new key:value pair: `incident-detection-response-identifier`.

The `incident-detection-response-identifier` value is taken from the alert name based on whatever payload your APM sends. APM alert name paths are different from one APM to another. An Lambda function must be set up to take the alarm name from the correct path in the APM JSON payload received by Lambda and use it for the `incident-detection-response-identifier` value.

`incident-detection-response-identifier` values must be unique per alarm type sent to AWS Incident Detection and Response. Each unique name that is set on the `incident-detection-response-identifier` must be provided to the AWS Incident Detection and Response team during on-boarding. Events that have an unknown or missing value for the `incident-detection-response-identifier` key are not processed.

Ingest Alarms from APMs with direct EventBridge integration

The following topic shows the process for sending alarms to AWS Incident Detection and Response from Application Performance Monitoring (APM) tools that have direct integration with Amazon EventBridge. For a complete list of APMs that have direct integration with Amazon EventBridge, see [Amazon EventBridge integrations](#).

You can deploy the provided [CloudFormation template](#) or manually setup this integration. Before setting up the integration, verify that the AWS service-linked role (SLR) `AWSServiceRoleForHealth_EventProcessor`, is [created](#) in your accounts.

Option 1: Using CloudFormation

A CloudFormation template is available to simplify the process of creating the integration infrastructure required to ingest alarms to AWS Incident Detection and Response from your APM with Amazon EventBridge integration.

Note

- Additional costs are incurred for resources deployed through this CloudFormation template (eg: Lambda and EventBridge). For more information about the pricing of these services, see [AWS Pricing](#).
- Deploy this CloudFormation template in every AWS account and Region where AWS Incident Detection and Response needs to ingest alarms. Incidents and Support Cases are opened on the AWS Account where the APM alert was received from.
- This document uses New Relic as an example, however the CloudFormation template can be used for any APM that has [SaaS integration with Amazon EventBridge](#).
- After testing the integration, remove `logger.info()` statements from the `TransformLambdaFunction` to prevent the payload from appearing in Amazon CloudWatch Logs.

Prerequisites for deploying this CloudFormation template:

- A Partner Event source must be setup in Amazon EventBridge. For instructions on setting up your APM as an event source, see [Receiving events from a SaaS partner with Amazon EventBridge](#) in the *Amazon EventBridge User Guide*.
- The `TransformLambdaFunction` (Lambda function) in the template must be modified to set `["detail"]["incident-detection-response-identifier"]` to the desired value based on the JSON path of the alert name in the APM payload.

Prerequisite Steps:

1. Open the EventBridge Console. Under the **Integration** menu, select **Partner event sources**.

- Search for your APM in the Amazon EventBridge partners box.
 - Choose **Setup**, then follow the instructions provided.
 - **Note:** the last step is to choose **Associate with Event Bus** in the console for the Partner event source. Selecting this option automatically create a Partner Event Bus with the same name as the Partner event source (the names must match).
 - Copy the name of the Partner Event Bus or source. The Event Bus or source is used as a parameter, named `PartnerEventBusNameParameter`, when deploying the CloudFormation template.
 - **Example** for New Relic: `aws.partner/newrelic.com/1234567/source_name`
 - Copy the first part of the Partner Event Bus or source to input into the `PartnerEventBusPrefixParameter` when deploying the CloudFormation template.
 - Example for New Relic is `aws.partner/newrelic.com`
2. Download and edit the [CloudFormation template](#).
- Locate the `TransformLambdaFunction` in the template
 - Under `def lambda_handler(event, context)` set `event["detail"]["incident-detection-response-identifier"]` to the json path where alarm name appears in the JSON payload of the APM alarm. Every APM will have a different path. Some examples can be seen below, however your specific payloads may differ.
 - **New Relic Example:** `event["detail"]["incident-detection-response-identifier"] = event["detail"]["workflowName"]`.
 - **Datadog Example:** `event["detail"]["incident-detection-response-identifier"] = event["detail"]["meta"]["monitor"]["name"]`
 - **Splunk Example:** `event["detail"]["incident-detection-response-identifier"] = event["detail"]["ruleName"]`
 - Save the CloudFormation template.

Deploying the CloudFormation Template:

1. Open the CloudFormation console in your target account and Region.
2. Choose Create stack, With new resources (standard)
 - Select **Choose an existing template, Upload a template file, Choose file**, then upload the CloudFormation template you saved locally.

3. Specify stack details:

- Enter a stack name (*Example:* NewRelicIntegrationForIDR).
 - Specify the **Parameter values** obtained during **Prerequisite completion**.
 - APMNameParameter (*Example:* NewRelic)
 - PartnerEventBusNameParameter (*Example:* aws.partner/newrelic.com/1234567/source_name)
 - PartnerEventBusPrefixParameter (*Example:* aws.partner/newrelic.com)
 - Choose **Next**.
4. Configure stack options:
- Scroll to the bottom of the page and check the box to allow CloudFormation to create IAM resources with custom names.
5. Review and create:
- Validate the parameter values are configured correctly and choose **Submit**.
6. The CloudFormation stack deploys the resources necessary to integrate your APM events to AWS Incident Detection and Response. Wait for the stack status to show CREATE_COMPLETE.
7. The CloudFormation stack creates the following resources, assuming the example values were input into the parameters for New Relic and was run in the US-EAST-1 Region.
- CustomEventBus: NewRelic-AWSIncidentDetectionResponse-EventBus
 - EventBridgeRule: aws.partner/newrelic.com/1234567/source_name|NewRelic-AWSIncidentDetectionResponse-EventBridgeRule
 - TransformLambdaExecutionRole: IDR-TransformLambdaExecutionRole-us-east-1
 - TransformLambdaFunction: NewRelic-AWSIncidentDetectionResponse-Lambda-Transform
 - TransformLambdaPermission: NewRelicIntegrationForIDR-TransformLambdaPermission-[random_string]

Integration testing

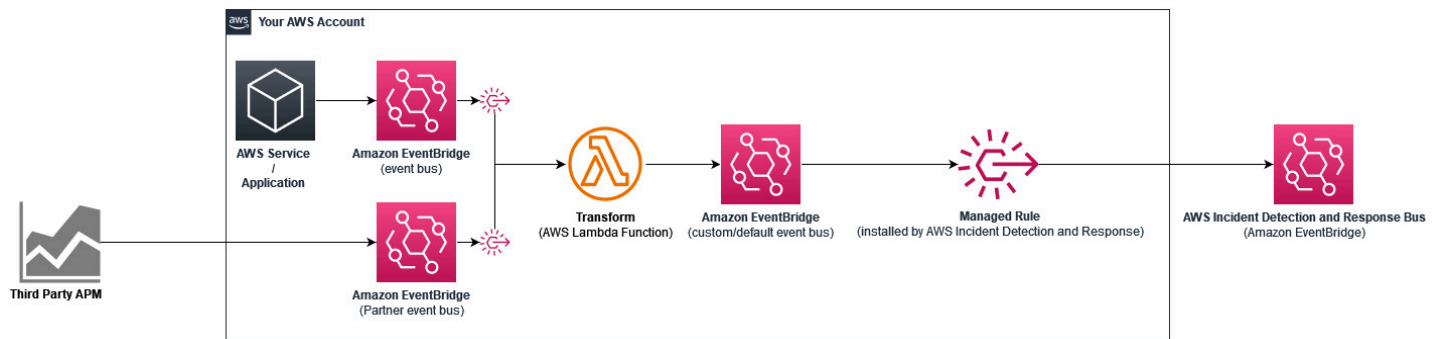
After deploying the stack, test the integration by sending a test payload from your APM:

1. Navigate to the Lambda Console and select the APMNameParameter-AWSIncidentDetectionResponse-Lambda-Transform function. Choose the **Monitor** tab.
2. Look for a successful invocation in the metric graphs.
3. Choose **View Amazon CloudWatch Logs** to check Log streams for your test payload or any errors.

Sharing Your Event Bus ARN to AWS Incident Detection and Response

1. Open the Amazon EventBridge Console. Select **Event buses**.
2. Copy the ARN of the **Custom event bus** created as part of the CloudFormation stack, (example: `arn:aws:events:us-east-1:123456789123:event-bus/NewRelic-AWSIncidentDetectionResponse-EventBus`)
 - Add this ARN to the "EventBridge Event Bus ARN" field in the "Third-Party APM Alarms" section of your [Alarm ingestion questionnaire - Overview](#).
3. During the onboarding process, AWS Incident Detection and Response creates a managed EventBridge rule on this custom event bus to ingest your APM alarms.

Option 2: Manual integration



Complete the following steps for each AWS account and AWS Region where AWS Incident Detection and Response needs to ingest alarms from. AWS Incident Detection and Response recommends to set up alarms in the same AWS account and Region as your application resources to make it quicker to identify and investigate impacted resources. Incidents and Support Cases are opened on the AWS Account where the APM alert was received from.

1. Create an EventBridge partner event bus by setting up your APM as an Amazon EventBridge partner event source (for example, `aws.partner/apm_name/integrationName`). For guidelines on setting up your APM as an event source, see [Receiving events from a SaaS partner with Amazon EventBridge](#).
2. Perform one of the following:
 - (Recommended) Create an EventBridge custom event bus named `$YourApmName-AWSIncidentDetectionResponse-EventBus`.
 - (Alternative) Use the default EventBridge event bus instead of a custom event bus.

AWS Incident Detection and Response will install a managed rule

(`AWSHealthEventProcessorEventSource-DO-NOT-DELETE`) on the custom or default event bus through the `AWSServiceRoleForHealth_EventProcessor` SLR. The rule source will be the custom or default event bus, the rule destination will be AWS Incident Detection and Response, and the rule will match the pattern for ingesting 3rd party APM events.

3. Create an [Lambda](#) function named `$YourApmName-AWSIncidentDetectionResponse-LambdaFunction` to transform your partner event bus events. The transformed events will match the managed rule `AWSHealthEventProcessorEventSource-DO-NOT-DELETE`.
 - Transformed events include a unique AWS Incident Detection and Response identifier, and sets the source and detail type of the event to the required values. This allows the transformed JSON payload structure to match the managed rule pattern.
 - Set the target of the Lambda function to either the custom event bus (Recommended) created in Step 2 or to your default event bus.
4. Create an EventBridge rule and define the event patterns that match the list of events that you want to push to AWS Incident Detection and Response. The source of the rule is the partner event bus you created in Step 1 (`aws.partner/apm_name/integrationName`). The target of the rule is the Lambda function you created in Step 3 (`[apm_name]-AWSIncidentDetectionResponse-LambdaFunction`). For guidelines on defining your EventBridge rule, see [Amazon EventBridge rules](#).

For a step by step example on how to set up partner event bus integrations manually with AWS Incident Detection and Response, see [Integrating notifications from Datadog and Splunk](#).

Ingest alarms from APMs without direct integration with EventBridge

AWS Incident Detection and Response supports using webhooks for alarm ingestion from third party APMs that don't have direct integration with Amazon EventBridge.

You can deploy a CloudFormation template or manually set up the integration. Before setting up the integration, verify that the AWS service-linked role (SLR) `AWSServiceRoleForHealth_EventProcessor`, is [created](#) in your accounts.

Option 1: Using CloudFormation Template

A CloudFormation template is available to simplify the process of creating the integration infrastructure required to ingest alarms to AWS Incident Detection and Response from your APM that does not have direct Amazon EventBridge integration.

Considerations before deploying this CloudFormation Template

- This solution uses an API Gateway Lambda Authorizer to compare a secret token passed in the payload from your APM against a token in AWS Secrets Manager. If the token does not match, a policy with an explicit deny will be returned. For more information, see [Lambda Authorizers](#).
- Under the AWS Shared Responsibility model, it is your responsibility to ensure you use an authentication approach that meets your organization's security requirements. We recommend using AWS Secrets Manager or a similar service, instead of storing sensitive information like API keys or authorization tokens as hard-coded variables. For more information, see [Create and manage secrets with AWS Secrets Manager](#).
- For an additional example of implementing Hash-Based Message Authentication Code (HMAC), see [receive-webhooks on the aws-samples Github page](#). For more information on implementing token authorization, see [example TOKEN authorizer Lambda function](#) from the API Gateway documentation.
- The solution uses **RateLimit**, **BurstLimit**, and **Quota** in API Gateway to control request volumes. These tools limit how many requests can be processed in a set time. This helps prevent system overload and keeps the service stable. For more information on throttling, refer to the [API Gateway Developer Guide](#).
- Consider using AWS Web Application Firewall (WAF) to protect the API Gateway from known bad IP addresses. This reduces the risk of attackers flooding the API with fake requests that could block real log events.
- AWS Secrets Manager token values should be stored in your Application Performance Monitoring (APM) tool as an HTTP header. Ensure to rotate the token on a regular basis as a security best practice.
- Additional costs will be incurred for resources deployed through this CloudFormation template (eg: Lambda and EventBridge). For more information about the pricing of these services, see [AWS Pricing](#).
- After testing the integration, remove `logger.info()` statements from the `TransformLambdaFunction` (Lambda function) to prevent payloads from appearing in Amazon CloudWatch Logs.
- Deploy this CloudFormation template in every AWS account and Region where AWS Incident Detection and Response needs to ingest alarms from.

Preparing the CloudFormation Template:

Note: The integration steps use Dynatrace as an example, however this template can be used for any APM that can send payloads to an API Gateway.

1. Download and open the [CloudFormation template](#).
2. Locate `APIGWUsagePlan` in the template. Review the values configured for `RateLimit`, `BurstLimit`, and `Quota Limit` which are set to 20, 50 & 2000 by default. Adjust the values to meet your requirements.
3. Locate `AuthorizerLambdaFunction` in the template. This Lambda function serves as an example of an authentication mechanism. It extracts a token value from a header called `authorizationToken`, which is passed from your APM. You can modify this code to align with your organization's security policies and APM requirements.
4. Locate the `TransformLambdaFunction` in the template. Replace the dictionary path, `raw_json["detail"]["ProblemTitle"]`, with the path to your alarm's name that is sent in the JSON payload from your APM. Leave this as is for Dynatrace.

Deploying the CloudFormation template:

1. Open the CloudFormation console in your target account and AWS Region.
2. Choose **Create stack, With new resources (standard)**.
 - Select **Choose an existing template, Upload a template file, Choose file**, then upload the CloudFormation template you saved locally.
3. Specify stack details:
 - Enter a stack name (*example, DynatraceIntegrationForIDR.*)
 - `APMNameParameter` (*example, Dynatrace.*)
 - Choose **Next**.
4. Configure stack options:
 - Scroll to the bottom of the page and check the box to allow CloudFormation to create IAM resources with custom names.
5. Review and create:
 - Validate the parameter values are configured correctly and choose **Submit**.
6. The CloudFormation stack deploys the resources necessary to integrate your APM events to AWS Incident Detection and Response. Wait until the CloudFormation Stack Status is **CREATE_COMPLETE**.

7. The CloudFormation stack creates the below resources assuming the example value `Dynatrace` was input into the parameters and was executed in the US-EAST-1 Region.
 - Secret name: `DynatraceMySecretTokenName` (a random Secret value will be created against Secret key `APMSecureToken`)
 - API Gateway resources:
 - API Name: `Dynatrace-AWSIncidentDetectionResponse-APIGW`
 - Stage Name: `Dynatrace-Stage-Prod`
 - Authorizers: `Dynatrace-APIGW-Authorizer`
 - Usage plan: `APIGW_Throttling_Plan`
 - Lambda functions:
 - Function for authorization: `Dynatrace-AWSIncidentDetectionResponse-Lambda-Authorizer`
 - Function for transformation: `Dynatrace-AWSIncidentDetectionResponse-Lambda-Transform`
 - Custom EventBus Name: `Dynatrace-AWSIncidentDetectionResponse-EventBus`
 - IAM Role:
 - `TransformLambdaExecutionRole`: `IDR-TransformLambdaExecutionRole-us-east-1`
 - `AuthorizerLambdaExecutionRole`: `IDR-AuthorizerLambdaExecutionRole-us-east-1`
8. Record the Webhook URL and Token value:
 - Open the API Gateway console and choose your API Name created as part of the CloudFormation stack.
 - Choose Stages from the left-hand navigation, expand the stage name using the + sign, then choose POST. Record the **Invoke URL**. Configure this URL in your APM as the destination to send webhooks for alarm events.
 - Open the AWS Secrets Manager console and choose the Secret name created as part of the CloudFormation stack. (*Example: `DynatraceMySecretTokenName`.*)
 - In the Secret value tab, choose **Retrieve secret value**. You will see the Secret key as `APMSecureToken`. Record the Secret value. Do not share this secret value with anyone.

Integration testing

After deploying the stack, test the integration by sending a test payload from your APM:

1. Navigate to the Lambda Console and select `APMNameParameter-`

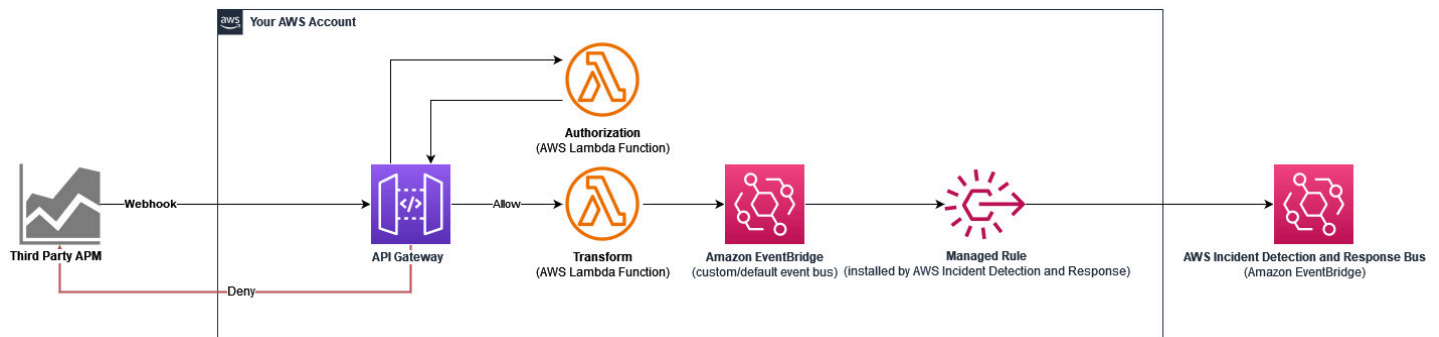
`AWSIncidentDetectionResponse-Lambda-Transform` function. Choose the **Monitor** tab.

2. Look for a successful invocation in the metric graphs.
3. Choose **View Amazon CloudWatch Logs** to check Log streams for your test payload or any errors.

Sharing Your Event Bus ARN to AWS Incident Detection and Response

1. Open the Amazon EventBridge Console. Select Event buses.
2. Copy the ARN of the **Custom event bus** created as part of the CloudFormation stack, *example: `arn:aws:events:us-east-1:123456789123:event-bus/Dynatrace-AWSIncidentDetectionResponse-EventBus`*.
 - Add this ARN to the "EventBridge Event Bus ARN" field in the "Third-Party APM Alarms" section of your [Alarm ingestion questionnaire - Overview](#).
3. During the onboarding process, AWS Incident Detection and Response will create a Managed EventBridge rule on this custom event bus to ingest your APM alarms.

Option 2: Manual integration



Use the following steps to set up integration with AWS Incident Detection and Response.

1. Create an Amazon API Gateway to accept the payload from your APM.
2. Define an Lambda function for authorization using an authentication token.
3. Perform one of the following:
 - (Recommended) Create an EventBridge custom event bus named `$YourApmName-AWSIncidentDetectionResponse-EventBus`.
 - (Alternative) Use the default EventBridge event bus instead of a custom event bus.
4. Define a Transform Lambda function to append the AWS Incident Detection and Response identifier to your payload. You can also use this function to filter for the events that you want to send to AWS Incident Detection and Response.

- The API Gateway must invoke the Transform Lambda function which will transform the payload passed by the API Gateway.
 - The Transform Lambda Function must write transformed events in the event bus defined in point 3 above.
5. Set up your APM to send notifications to the URL generated from the API Gateway.

Ingest Alarms from APMs with direct Amazon SNS integration

If your APM supports sending alarms to Amazon SNS topics you can follow this guide to ingest your APM alarms to AWS Incident Detection and Response.

You can deploy the provided [CloudFormation template](#) or manually set up this integration. Before setting up the integration, verify that the AWS service-linked role (SLR) `AWSServiceRoleForHealth_EventProcessor`, is [created](#) in your accounts.

Option 1: Using CloudFormation

A CloudFormation template is available to simplify the process of creating the integration infrastructure required to ingest alarms to AWS Incident Detection and Response from your APM with Amazon SNS integration.

Note

- Additional costs will be incurred for resources deployed through this CloudFormation template (eg: Lambda and EventBridge). For more information about the pricing of these services, see [AWS Pricing](#).
- This CloudFormation template must be deployed in every AWS account and Region that alarms need to be ingested by AWS Incident Detection and Response from.
- The examples provided in this document are for Grafana, however this template can be used for any APM that has direct integration with Amazon Simple Notification Service.
- For security reasons, AWS recommends removing `logger.info()` statements from the `TransformLambdaFunction` to prevent the payload from being logged in Amazon CloudWatch Logs.

Prerequisites for deploying this CloudFormation template:

- An Amazon Simple Notification Service topic must be created to receive alarm events from your APM. [Create an SNS topic in the Amazon Simple Notification Service console](#).
- The `TransformLambdaFunction` in the template must be modified to set `["detail"]` `["incident-detection-response-identifier"]` to the desired value based on the APM being used.

Prerequisite completion:

1. Open the Amazon SNS Console, then select Topics. Copy the ARN of the SNS Topic created to receive alarm events from your APM.
 - Example: `arn:aws:sns:eu-west-1:012345678912:<your-apm-name>-sns`
2. Download and open the [CloudFormation template](#)
 - Locate the `TransformLambdaFunction` in the template
 - Under `def lambda_handler(event, context)` set `event["detail"]["incident-detection-response-identifier"]` to the json path where the alarm name appears in the JSON payload of the SNS record.
 - Any event sent to the `TransformLambdaFunction` via SNS has a parent payload structure as `event["Records"][n]["Sns"]["Message"]`. The actual payload origin from the source (APM) is wrapped inside the parent structure.
 - **Example for Grafana:** `event["Records"][n]["Sns"]["Message"]["alerts"][n]["labels"]["alertname"]`

Deploying the CloudFormation Template:

1. Navigate to the CloudFormation console in the account and Region you need to set up the integration in.
2. Navigate to CloudFormation.
 - Choose Create stack, With new resources (standard)
 - Select Choose an existing template, Upload a template file, Choose file, then upload the CloudFormation template you saved locally.
3. Specify stack details:
 - Enter a stack name *Example:* `<your-apm-name>IntegrationForIDR`
 - Specify the Parameter values obtained during **Prerequisite completion**
 - `APMNameParameter` *Example:* `Grafana`

- TriggerSNSParameter *Example*: `arn:aws:sns:eu-west-1:012345678912:<your-apm-name>-sns`
 - Choose Next.
4. Configure stack options:
 - Scroll to the bottom of the page and acknowledge the checkbox to allow CloudFormation to create IAM resources with custom names.
 5. Review and create:
 - Validate the parameter values are configured correctly, then choose Submit.
 6. The CloudFormation stack will deploy the resources necessary to integrate your APM events to AWS Incident Detection and Response. Wait until the CloudFormation Stack Status is **CREATE_COMPLETE**.
 7. The CloudFormation stack creates the below resources assuming the example values were input into the parameters for Grafana and was executed in the EU-WEST-1 Region.
 - CustomEventBus: Grafana-AWSIncidentDetectionResponse-EventBus
 - SNSSubscription: `arn:aws:sns:eu-west-1:012345678912:grafana-sns:[random_string]`
 - TransformLambdaExecutionRole: IDR-TransformLambdaExecutionRole-eu-west-1
 - TransformLambdaFunction: Grafana-AWSIncidentDetectionResponse-Lambda-Transform
 - TransformLambdaPermission: GrafanaIntegrationForIDR-TransformLambdaPermission-[random_string]

Integration testing

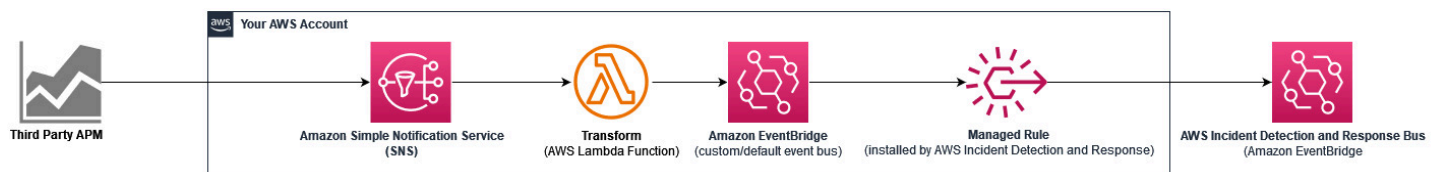
After the CloudFormation stack is deployed successfully, you can validate the integration by sending a test payload from your APM. Once the test payload is sent from your APM:

1. Navigate to the Lambda Console and select the `APMNameParameter-AWSIncidentDetectionResponse-Lambda-Transform` function. Then, choose the Monitor tab.
2. A successful invocation should be observed in the metric graphs.
3. Select View Amazon CloudWatch Logs. You can verify from the Log events in the Log streams to confirm that the test payload sent from your APM is present, or if any errors were encountered.

Sharing Your Event Bus ARN to AWS Incident Detection and Response

1. Navigate to the Amazon EventBridge Console. Select Event buses.
2. Record the ARN of the **Custom event bus** deployed as part of the CloudFormation stack, for example: `arn:aws:events:eu-west-1:012345678912:event-bus/Grafana-AWSIncidentDetectionResponse-EventBus`.
 - Provide the ARN of this Custom event bus to AWS Incident Detection and Response in the "EventBridge Event Bus ARN" field of the "Third-Party APM Alarms" section of the [Alarm ingestion questionnaire - Overview](#).
3. During the onboarding process, AWS Incident Detection and Response will create a Managed EventBridge rule on this custom event bus to ingest your APM alarms.

Option 2: Manual integration



1. Open Amazon SNS Console and [Create an SNS topic in the Amazon Simple Notification Service console](#) named `[apm_name]-sns` to receive alarms events from your APM. Note the ARN of the SNS Topic created.
2. Perform one of the following:
 - (Recommended) Create an EventBridge custom event bus named `[apm_name]-AWSIncidentDetectionResponse-EventBus`.
 - (Alternative) Use the default EventBridge event bus instead of a custom event bus.

AWS Incident Detection and Response will install a managed rule (`AWSHealthEventProcessorEventSource-DO-NOT-DELETE`) on the custom or default event bus through the `AWSServiceRoleForHealth_EventProcessor` SLR. The rule source will be the custom or default event bus, the rule destination will be AWS Incident Detection and Response, and the rule will match the pattern for ingesting 3rd party APM events.

3. Create an [Lambda](#) function named `$YourApmName-AWSIncidentDetectionResponse-LambdaFunction` to transform your SNS payloads.
 - Transformed events must meet the payload requirements as set out in [Payload Requirements For Ingesting APM Alerts with EventBridge](#)
 - Set the target of the Lambda function to either the custom event bus (Recommended) created in Step 2 or to your default event bus.

4. Set the SNS topic as a trigger for your Lambda function `$YourApmName-AWSIncidentDetectionResponse-LambdaFunction`.
 - In the "Add Triggers" page, search for "SNS".
 - Add the ARN of your dedicated SNS Topic created in Step 1.
 - Choose "Add".
5. Follow your APM documentation to set up an SNS destination for your APM payloads that need to be ingested by AWS Incident Detection and Response.

AWS Incident Detection and Response will install a managed rule (`AWSHealthEventProcessorEventSource-DO-NOT-DELETE`) on the custom or default event bus through the `AWSServiceRoleForHealth_EventProcessor` SLR. The rule source will be the custom or default event bus, the rule destination will be AWS Incident Detection and Response, and the rule will match the pattern for ingesting 3rd party APM events.

Alarm optimization and monitoring adjustments

To ensure optimal incident detection accuracy, our Incident Management Engineers continuously evaluate alarm performance against your critical workloads. We provide recommended alarm configuration changes, which you are required to make, and proactively collaborate with you and your Technical Account Managers (TAMs) to refine these settings.

When monitoring data indicates that alarms may not be aligned with your business-critical operations, such as when alerts trigger without corresponding customer impact or when alarm states fluctuate frequently, we would recommend offboarding the non-critical alarms and onboarding alarms that better reflect critical workload impact. This helps maintain the overall effectiveness of your incident response coverage.

Alarm review and feedback

AWS Incident Detection and Response conduct comprehensive reviews of your alarms prior to onboarding them for monitoring. Alarms are evaluated against a technical acceptance criteria including configuration parameters, data quality and alert effectiveness.

Based on this review, two types of feedback are provided:

- Mandatory configuration requirements - these changes must be implemented for alarm acceptance.

- **Optional improvement recommendations** - these changes enhance alarm effectiveness but are not mandatory for alarm acceptance.

After receiving this feedback, you can decide to proceed with only onboarding accepted alarms and those needing optional improvements, while working on configuration changes for alarms with mandatory configuration requirements in parallel.

Alternatively, you can implement all changes before going live. This approach extends the onboarding timeline, based on the number of alarms requiring adjustments.

Alarm testing (Gameday)

The last step in the AWS Incident Detection and Response onboarding process is to perform a Gameday for your new workload. After Alarm Ingestion steps, AWS Incident Detection and Response confirms a date and time of your choosing to start your Gameday.

Your Gameday serves two main purposes:

- **Functional Validation:** Confirms that AWS Incident Detection and Response can correctly receive your alarm events. And, functional validation confirms that your alarm events trigger the desired actions, such as automatic support case creation if you selected it during alarm ingestion.
- **Simulation:** The Gameday is an end to end simulation of what might happen during a real incident. AWS Incident Detection and Response gives you insight into how a real incident might unfold. The Gameday is an opportunity for you to ask questions or refine instructions to improve the engagement.

During the alarm test, AWS Incident Detection and Response works with you to remediate any issues identified.

CloudWatch Alarm testing

During the Gameday, Amazon CloudWatch alarms are tested by manually changing the alarm to the **Alarm** state using the AWS Command Line Interface. You can also access the AWS CLI from AWS CloudShell. AWS Incident Detection and Response provides you with a list of AWS CLI commands for you to use during testing.

Example AWS CLI command to set an alarm state:

```
aws cloudwatch set-alarm-state --alarm-name "ExampleAlarm" --state-value ALARM --state-reason "Testing AWS Incident Detection and Response" --region us-east-1
```

Note

The AWS Identity and Access Management user or role that you use for alarm testing must have `cloudwatch:SetAlarmState` permission.

To learn more about manually changing the state of CloudWatch alarms, see [SetAlarmState](#).

To learn more about the permissions required for CloudWatch API operations, see [Amazon CloudWatch permissions reference](#).

Third party APM alarms testing

Workloads that utilize a third party Application Performance Monitoring (APM) tool, such as Datadog, Splunk, New Relic, or Dynatrace, require different instructions to simulate an alarm. At the start of the Gameday, AWS Incident Detection and Response requests that you temporarily change your alarm thresholds or comparison operators to force the alarm into the **ALARM** status. This status triggers a payload to AWS Incident Detection and Response.

The Gameday validates the following points

- Alarm ingestion is successful and your alarm configuration is correct.
- Alarms are successfully created and received by AWS Incident Detection and Response.
- A support case is created for your incident and your prescribed runbook contacts are notified.
- AWS Incident Detection and Response can engage with you by your defined conference bridge method.

Alarms go live

After the Gameday has been successfully completed, AWS Incident Detection and Response sends a go live communication through your onboarding support case. From this point onward, your onboarded alarms are monitored and AWS Incident Detection and Response will engage you per the workload's contact details when your onboarded alarms enter the ALARM state.

Key outputs

- A Go-Live correspondence is sent confirming your workload is now being monitored by AWS Incident Detection and Response.

Any changes required that were identified during the Gameday, AWS Incident Detection and Response fulfills them using a [Request changes to an onboarded workload in Incident Detection and Response](#).

Workload onboarding and alarm ingestion questionnaires in Incident Detection and Response (exception path)

Note

If you are unable to use the AWS Incident Detection and Response Customer Command Line Interface to onboard your workload, use the following questionnaires for workload and alarm onboarding.

This page provides the questionnaires you need to complete when onboarding a workload to AWS Incident Detection and Response and when configuring alarms to ingest into the service. The workload onboarding questionnaire covers general information about your workload, its architecture details, and contacts for incident response. In the alarm ingestion questionnaire, you specify the critical alarms that should trigger incident creation in Incident Detection and Response for your workload, as well as runbook information on who should be contacted and what actions should be taken. Properly completing these questionnaires is a key step in setting up monitoring and incident response processes for your AWS workloads.

Download the Workload onboarding questionnaire:

- [English version](#)
- [Japanese version](#)

Download the Alarm ingestion questionnaire:

- [English version](#)
- [Japanese version](#)


Workload onboarding questionnaire - General questions





General questions


Question	Example Response
Enterprise Name	Amazon Inc.
Name of this workload (include any abbreviations)	Amazon Retail Operations (ARO)
Primary end user and the function of this workload.	This workload is an e-commerce application that allows end users to purchase various items. This workload is the primary revenue generator for our business.
Applicable compliance and/or regulatory requirements for this workload and any actions required from AWS after an incident.	The workload deals with patient health records which must be kept secured and confidential.

Workload onboarding questionnaire - Architecture questions

Architecture questions

Question	Example Response
A list of AWS resource tags used to define resources that are part of this workload. AWS uses these tags to identify this workload's resources to expedite support during incidents.	<p>appName: Optimax</p> <p>environment: Production</p>
<div data-bbox="142 1589 271 1627">  Note </div> <p>Tags are case sensitive. If you provide multiple tags, all resources used by this workload must have the same tags.</p>	

Question	Example Response
<p>A list of AWS services utilized by this workload and the AWS Account and Regions that they're in.</p> <div data-bbox="115 401 792 569"><p> Note Create a new row for each service.</p></div>	<p>Route 53: Routes internet traffic to the ALB.</p> <p>Account:123456789101</p> <p>Region: US-EAST-1, US-WEST-2</p>
<p>A list of AWS services utilized by this workload and the AWS Account and Regions that they're in.</p> <div data-bbox="115 783 792 951"><p> Note Create a new row for each service.</p></div>	<p>ALB: Routes incoming traffic to a target group of ECS containers.</p> <p>Account: 123456789101</p> <p>Region: N/A</p>
<p>A list of AWS services utilized by this workload and the AWS Account and Regions that they're in.</p> <div data-bbox="115 1167 792 1335"><p> Note Create a new row for each service.</p></div>	<p>ECS: Compute infrastructure for main business logic fleet. Responsible for handling incoming user requests and making queries to persistence layer.</p> <p>Account: 123456789101</p> <p>Region: US-EAST-1</p>
<p>A list of AWS Services utilized by this workload and the AWS Account and Regions that they're in.</p> <div data-bbox="115 1551 792 1719"><p> Note Create a new row for each service.</p></div>	<p>RDS: Amazon Aurora cluster stores user data accessed by ECS business logic layer.</p> <p>Account: 123456789101</p> <p>Region: US-EAST-1</p>

Question	Example Response
<p>A list of AWS Services utilized by this workload and the AWS Account and Regions that they're in.</p> <div data-bbox="115 401 792 569" style="border: 1px solid #add8e6; border-radius: 10px; padding: 10px; margin-top: 10px;"> <p> Note Create a new row for each service.</p> </div>	<p>S3: Stores website static assets.</p> <p>Account: 123456789101</p> <p>Region: N/A</p>
<p>Detail any upstream/downstream component s not being onboarded that could affect this workload if experiencing an outage.</p>	<p>Authentication Microservice: Will prevent users from loading their health records as they will be unauthenticated.</p>
<p>Are there any on-premise or non-AWS components for this workload? If so, what are they and what functions are performed?</p>	<p>All internet based traffic in/out of AWS is routed via our on-prem proxy service.</p>
<p>Provide details of any manual or automated failover/disaster recovery plans at the Availability Zone and regional level.</p>	<p>Warm standby. Automated failover to US-WEST-2 during sustained drop in success rate.</p>

Alarm ingestion questionnaire - Overview

In the alarm ingestion questionnaire, you specify the critical alarms for your workload that you want to engage AWS Incident Detection and Response, as well as the contacts you want an Incident Management Engineer to engage when these alarms trigger.


The Alarm Ingestion Questionnaire is divided into the following sections:

- Contact section:** First, specify the primary contact(s) to be included on the Support Case created with AWS Incident Detection and Response when an alarm triggers, as well as your preferred conferencing application for incident bridges. If no bridge preference is provided, AWS Incident Detection and Response will create an incident bridge during incidents. Next, specify escalation contacts and time intervals to engage them when primary contacts are unreachable. Finally, list any contacts who should receive regular incident status updates through the support case for the duration of an incident.

- **Alarm matrix:** List the set of alarms that will engage AWS Incident Detection and Response when triggered. See the "Critical Alarm Criteria" defined by AWS Incident Detection and Response when selecting alarms for onboarding. For more information, see [Alarm definition](#).
- **Amazon CloudWatch Alarms** (leave this section blank if you don't have Amazon CloudWatch alarms)
- **Third party APM alarms** (leave this section blank if you don't have Third party APM alarms)
 - **EventBridge EventBus ARN:** This is the ARN of the custom EventBus ARN that you created in [Ingest Alarms from APMs with direct EventBridge integration](#) or [Ingest alarms from APMs without direct integration with EventBridge](#).
 - **Alarm Identifiers:** Share the account number, region, and name of the APM alarm.

Alarm ingestion questionnaire - Runbook questions

Runbook questions

Question	Example Response
<p>AWS engages workload contacts through the Support case. Who is the primary contact when an alarm triggers for this workload?</p> <p>Specify your preferred conferencing application and AWS will request these details during an incident.</p>	<p>Application Team</p> <p>app@example.com</p> <p>+61 2 3456 7890</p>
<div data-bbox="115 1346 792 1661" style="border: 1px solid #add8e6; border-radius: 10px; padding: 10px;"> <p> Note</p> <p>If a preferred conferencing application isn't provided, then AWS will reach out during an incident and provide a Chime bridge for you to join.</p> </div>	
<p>If the primary contact is unavailable during an incident, please provide escalation contacts and timeline in the preferred communication order.</p>	<p>1. After 10 minutes, if no response from Primary Contact, engage:</p> <p>John Smith - Application Supervisor</p>

Question	Example Response
<p>AWS communicates updates through the support case at regular intervals throughout the incident. Are there additional contacts that should receive these updates?</p>	<p>john.smith@example.com</p> <p>+61 2 3456 7890</p> <p>2. After 10 minutes, if no response from John Smith, contact:</p> <p>Jane Smith - Operations Manager</p> <p>jane.smith@example.com</p> <p>+61 2 3456 7890</p> <p>john.smith@example.com, jane.smith@example.com</p>

Alarm matrix

Provide the following information to identify the set of alarms that will engage AWS Incident Detection and Response to create incidents on behalf of your workload. Once engineers from AWS Incident Detection and Response have reviewed your alarms, additional onboarding steps will be delivered.

AWS Incident Detection and Response Critical alarm criteria:

- AWS Incident Detection and Response alarms should only enter "Alarm" state upon significant business impact to the monitored workload (loss of revenue/degraded customer experience) that requires immediate operator attention.
- AWS Incident Detection and Response alarms must also engage your resolvers for the workload at the same time or prior to engagement. AWS Incident Managers collaborate with your resolvers in the mitigation process, and do not serve as a first-line responders who then escalate to you.
- AWS Incident Detection and Response alarm thresholds must be set to an appropriate threshold and duration so that any time an alarm fires an investigation must take place. If an alarm is moving between the "Alarm" and "OK" state, sufficient impact is occurring to warrant operator response and attention.

AWS Incident Detection and Response Policy for criteria violations:

These criteria can only be evaluated on a case-by-case basis as events occur. The Incident Management team works with your technical account managers (TAMs) to adjust alarms and in rare cases disable monitoring if it is suspected that customer alarms do not adhere to this criteria and is engaging the Incident Management team unnecessarily at a regular rate.

Important

Provide a group distribution email addresses when supplying contact addresses, so that you can control recipient additions and deletions without runbook updates.

Provide the contact phone number for your site reliability engineering (SRE) team if you would like the AWS Incident Detection and Response team to call them after sending an initial engagement email.

Alarm matrix table

Metric name / ARN / Threshold	Description	Notes	Actions requested
Workload volume / <i>CW Alarm ARN</i> / CallCount < 100000 for 5 datapoints within 5 minute , treat missing data as missing	<p>This metric represents the number of incoming requests coming to the workload, measured at the Application Load Balancer level.</p> <p>This alarm is important because significant drops in incoming requests may indicate issues with upstream network connectivity, or issues with our DNS implementation</p>	<p>The alarm has entered the "Alarm" state 10 times in the last week. This alarm is at risk of false positives. Threshold review is planned.</p> <p>Issues? No or Yes (if No, leave blank): This alarm flips frequently during a particular batch job execution.</p> <p>Resolvers: Site Reliability Engineers</p>	<p>Engage the Site Reliability Engineering team by sending an email to <i>SRE@example.com</i></p> <p>Create an AWS Support case for our ELB, and Amazon Route 53 services.</p> <p>If IMMEDIATE action is needed: Check EC2 Free memory/disk space and inform the <i>Example</i> team through email to restart the instance,</p>

Metric name / ARN / Threshold	Description	Notes	Actions requested
	that result in users not being able to access the workload.		or run a log flush. (if immediate action is not needed, leave blank)
<p>Workload Request Latency /</p> <p><i>CW Alarm ARN /</i></p> <p>p90 Latency > 100ms for 5 datapoints within 5 minutes , treat missing data as missing</p>	<p>This metric represents the p90 latency for HTTP requests to be fulfilled by the workload.</p> <p>This alarm represents latency (important measure of customer experience for the website).</p>	<p>The alarm has entered the "Alarm" state 0 times in the last week.</p> <p>Issues? No or Yes (if No, leave blank): This alarm flips frequently during a particular batch job execution.</p> <p>Resolvers: Site Reliability Engineers</p>	<p>Engage the Site Reliability Engineering team by sending an email to <i>SRE@example.com</i></p> <p>Create an AWS Support case for our ECW, and RDS services.</p> <p>If IMMEDIATE action is needed: Check EC2 Free memory/disk space and inform the <i>Example</i> team through email to restart the instance, or run a log flush. (if immediate action is not needed, leave blank)</p>

Metric name / ARN / Threshold	Description	Notes	Actions requested
<p>Workload Request Availability /</p> <p><i>CW Alarm ARN /</i></p> <p>Availability < 95% for 5 datapoints within 5 minutes , treat missing data as missing.</p>	<p>This metric represents the availability for HTTP requests to be fulfilled by the workload. (# of HTTP 200 / # of Requests) per period.</p> <p>This alarm represents the availability of the workload.</p>	<p>The alarm has entered the "Alarm" state 0 times in the last week.</p> <p>Issues? No or Yes (if No, leave blank): This alarm flips frequently during a particular batch job execution.</p> <p>Resolvers: Site Reliability Engineers</p>	<p>Engage the Site Reliability Engineering team by sending an email to <i>SRE@example.com</i></p> <p>Create an AWS Support case for our ELB, and Amazon Route 53 services.</p> <p>If IMMEDIATE action is needed: Check EC2 Free memory/disk space and inform the <i>Example</i> team through email to restart the instance, or run a log flush. (if immediate action is not needed, leave blank)</p>

New Relic Alarm Example

Metric name / ARN / Threshold	Description	Notes	Actions requested
<p>End to End Integration test /</p> <p><i>CW Alarm ARN /</i></p> <p>3% failure rate for 1 minute metrics over 3 minutes duration , treat missing data as missing</p> <p>Workload Identifier: End to End Test Workflow, AWS Region: US-EAST-1 , AWS account ID: 012345678910</p>	<p>This metric tests if a request can traverse each layer of the workload. If this test fails, it represents a critical failure to process business transactions.</p> <p>This alarm represents the ability to process business transactions for the workload.</p>	<p>The alarm has entered the "Alarm" state 0 times in the last week.</p> <p>Issues? No or Yes (if No, leave blank): This alarm flips frequently during a particular batch job execution.</p> <p>Resolvers: Site Reliability Engineers</p>	<p>Engage the Site Reliability Engineering team by sending an email to <i>SRE@example.com</i></p> <p>Create an AWS Support case for our Amazon Elastic Container Service, and Amazon DynamoDB services.</p> <p>If IMMEDIATE action is needed: Check EC2 Free memory/disk space and inform the <i>Example</i> team through email to restart the instance, or run a log flush. (if immediate action is not needed, leave blank)</p>

Manage workloads in Incident Detection and Response

A key part of effective incident management is having the right processes and procedures in place to onboard, test, and maintain your monitored workloads. This section covers the essential steps, including developing comprehensive runbooks and response plans to guide your teams through incidents, thoroughly testing and validating new workloads before onboarding, requesting changes to update workload monitoring, and properly offboarding workloads when required.

Topics

- [Develop runbooks and response plans for responding to an incident in Incident Detection and Response](#)
- [Test onboarded workloads in Incident Detection and Response](#)
- [Request changes to an onboarded workload in Incident Detection and Response](#)
- [Suppress alarms from engaging Incident Detection and Response](#)
- [Offboard a workload from Incident Detection and Response](#)

Develop runbooks and response plans for responding to an incident in Incident Detection and Response

Incident Detection and Response uses information captured from your AWS Incident Detection and Response Customer Command Line Interface onboarding to develop runbooks and response plans for the management of incidents affecting your workloads. Runbooks document steps Incident Managers take when responding to an incident. A response plan is mapped to at least one of your workloads. The incident management team creates these templates from the information provided by you during [workload onboarding](#). Response plans are AWS Systems Manager (SSM) document templates used to trigger incidents. To learn more about SSM documents, see [AWS Systems Manager Documents](#). To learn more about Incident Manager, see [What Is AWS Systems Manager Incident Manager?](#)

Key outputs:

- Completion of your workload definition on AWS Incident Detection and Response.
- Completion of alarms, runbooks and response plan definition on AWS Incident Detection and Response.

You can also download an AWS Incident Detection and Response Runbook example: [aws-idr-runbook-example.zip](#).

Example runbook:

Runbook template for AWS Incident Detection and Response

Description

This document is intended for [CustomerName] [WorkloadName].

[Insert short description of what the workload is intended for].

Step: Priority

****Priority actions****

1. When a case is created with Incident Detection and Response, lock the case to yourself, verify the Customer Stakeholders in the Case from *Engagement Plans - Initial Engagement*.
2. Send the first correspondence on the support case to the customer as below. If there is no support case or if it is not possible to use the support case then backup communication details are listed in the steps that follow.

...

Hello,

This is <<Engineer's name>> from AWS Incident Detection and Response. An alarm has triggered for your workload <<application name>>. I am currently investigating and will update you in a few minutes after I have finished initial investigation.

Alarm Identifier - <insert CloudWatch Alarm ARN or APM Response Identifier>

...

****Compliance and regulatory requirements for the workload****

<<e.g. The workload deals with patient health records which must be kept secured and confidential. Information not to be shared with any third parties.>>

****Actions required from Incident Detection and Response in complying****

<<e.g Incident Management Engineers must not shared data with third parties.>>

Step: Information

****Review of common information****

* This section provides a space for defining common information which may be needed through the life of the incident.

* The target user of this information is the Incident Management Engineer and Operations Engineer.

* The following steps may reference this information to complete an action (for example, execute the "Initial Engagement" plan).

Engagement plans

Describe the engagement plans applicable to this runbook. This section contains only contact details. Engagement plans will be referenced in the step by step ****Communication Plans****.

* **Initial engagement**

AWS Incident Detection and Response Team will add customer stakeholder addresses below to the Support Case. AWS Stakeholders are for additional stakeholders that may need to be made aware of any issues.

When updating customer stakeholders details in this plan also update the Backup Mailto links.

- * *****Customer Stakeholders*****: customeremail1; customeremail2; etc

- * *****AWS Stakeholders*****: aws-idr-oncall@amazon.com; tam-team-email; etc.

- * *****One Time Only Contacts*****: [These are email contacts that are included on only the first communication. Remove these contacts after the first communication has gone out. These could be customer paging email addresses such as pager-duty that must not be paged for every correspondence]

- * *****Backup Mailto Impact Template*****: <*Insert Impact Template Mailto Link here*>

- * Use the backup Mailto when communication over cases is not possible.

- * *****Backup Mailto No Impact Template*****: <*Insert No Impact Mailto Link here*>

- * Use the backup Mailto when communication over cases is not possible.

* **Engagement Escalation**

AWS Incident Detection and Response will reach out to the following contacts when the contacts from the ****Initial engagement**** plan do not respond to incidents.

For each Escalation Contact indicate if they must be added to the support case, phoned or both.

- * *****First Escalation Contact*****: [escalationEmailAddress#1] / [PhoneNumber] - Wait XX Minutes before escalating to this contact.

- * [add Contact to Case / phone] this contact.

- * *****Second Escalation Contact*****: [escalationEmailAddress#2] / [PhoneNumber] - Wait XX Minutes before escalating to this contact.

- * [add Contact to Case / phone] this contact.

- * Etc;

Communication plans

Describe how Incident Management Engineer communicates with designated stakeholders outside the incident call and communication channels.

****Impact Communication plan****

This plan is initiated when Incident Detection and Response have determined from step ****Triage**** that an alert indicates potential impact to a customer.

Incident Detection and Response will request the customer to join the predetermined bridge (Chime Bridge/Customer Provided Bridge / Customer Static Bridge) as indicated in ****Engagement plans - Incident call setup****.

All backup email templates for use when cases can't be used are in ****Engagement plans - Initial engagement****.

* 1 - Before sending the impact notification, verify then remove and/or add customer contacts from the Support Case CC based on the contacts listed in the ****Initial engagement**** Engagement plan.

* 2 - Send the engagement notification to the customer based the following Template:

(choose one and remove the rest)

****Impact Template - Chime Bridge****

...

The following alarm has engaged AWS Incident Detection and Response to an Incident bridge:

Alarm Identifier - <insert CloudWatch Alarm ARN or APM Response Identifier>

Alarm State Change Reason - <insert state change reason>

Alarm Start Time - <Example: 1 January 2023, 3:30 PM UTC>

Please join the Chime Bridge below so we can start the steps outlined in your Runbook:

<insert Chime Meeting ID>

<insert Link to Chime Bridge>

International dial-in numbers: <https://chime.aws/dialinnumbers/>

...

****Impact Template - Customer Provided Bridge****

...

The following alarm has engaged AWS Incident Detection and Response:

Alarm Identifier - <insert CloudWatch Alarm ARN or APM Response Identifier>

Alarm State Change Reason - <insert state change reason>

Alarm Start Time - <Example: 1 January 2023 3:30 PM UTC>

Please respond with your internal bridge details so we can join and start the steps outlined in your Runbook.

...

****Impact Template - Customer Static Bridge****

...

The following alarm has engaged AWS Incident Detection and Response to an Incident bridge:

Alarm Identifier - <insert CloudWatch Alarm ARN or APM Response Identifier>

Alarm State Change Reason - <insert state change reason>

Alarm Start Time - <Example: 1 January 2023, 3:30 PM UTC>

Please join the Bridge below so we can start the steps outlined in your Runbook:

Conference Number: <insert conference number>

Conference URL : <insert bridgeURL>

...

- * 3 - Set the Case to Pending Customer Action
- * 4 - Follow **Engagement Escalation** plan as mentioned above.
- * 5 - If the customer does not respond within 30 minutes, disengage and continue to monitor until the alarm recovers.

* **No Impact Communication plan**

This plan is initiated when an alarm recovers before Incident Detection and Response have completed initial **Triage**.

- * 1 - Before sending the no impact notification, verify then remove and/or add customer contacts from the Support Case CC based on the contacts listed in the **Engagement plans - Initial engagement** Engagement plan.
- * 2 - Send a no engagement notification to the customer based on the below template:

No Impact Template

...

AWS Incident Detection and Response received an alarm that has recovered for your workload.

Alarm Identifier - <insert CloudWatch Alarm ARN or APM Response Identifier>

Alarm State Change Reason - <insert state change reason>

Alarm Start Time - <Example: 1 January 2023, 3:30 PM UTC>

Alarm End Time - <Example: 1 January 2023, 3:35 PM UTC>

This may indicate a brief customer impact that is currently not ongoing.

If there is an ongoing impact to your workload, please let us know and we will engage to assist.

...

- * 3 - Put the case in to Pending Customer Action.
- * 4 - If the customer does not respond within 30 minutes Resolve the case.

* **Updates**

If AWS Incident Detection and Response is expected to provide regular updates to customer stakeholders, list those stakeholders here. Updates must be sent via the same support case.

Remove this section if not needed.

- * Update Cadence: Every XX minutes
- * External Update Stakeholders: customeremailaddress1; customeremailaddress2; etc

- * Internal Update Stakeholders: awsemailaddress1; awsemailaddress2; etc

Application architecture overview

This section provides an overview of the application/workload architecture for Incident Management Engineer and Operations Engineer awareness.

- * ****AWS Accounts and Regions with key services**** - list of AWS accounts with regions supporting this application. Assists Engineers in assessing underlying infrastructure supporting the application.

- * 123456789012

- * US-EAST-1 - brief desc as appropriate

- * EC2 - brief desc as appropriate

- * DynamoDB - brief desc as appropriate

- * etc.

- * US-WEST-1 - brief desc as appropriate

- * etc.

- * another-account-etc.

- * ****Resource identification**** - describe how engineers determine resource association with application

- * Resource groups: etc.

- * Tag key/value: AppId=123456

- * ****CloudWatch Dashboards**** - list dashboards relevant to key metrics and services

- * 123456789012

- * us-east-1

- * some-dashboard-name

- * etc.

- * some-other-dashboard-name-in-current-acct

Step: Triage

Evaluate incident and impact

This section provides instructions for triaging of the incident to determine correct impact, description, and overall correct runbook being executed.

- * ****Evaluation of initial incident information****

- * 1 - Review Incident Alarm, noting time of first detected impact as well as the alarm start time.

- * 2 - Identify which service(s) in the customer application is seeing impact.

- * 3 - Review AWS Service Health for services listed under ****AWS Accounts and Regions with key services****.

- * 4 - Review any customer provided dashboards listed under ****CloudWatch Dashboards****

* **Impact**

Impact is determined when either the customer's metrics do not recover, appear to be trending worse or if there is indication of AWS Service Impact.

- * 1 - Start **Communication plans - Impact Communication plan**
- * 2 - Start **Engagement plans - Engagement Escalation** if no response is received from the **Initial Engagement** contacts.
- * 3 - Start **Communication plans - Updates** if specified in **Communication plans**

* **No Impact**

No Impact is determined when the customer's alarm recovers before Triage is complete and there are no indications of AWS service impact or sustained impact on the customer's CloudWatch Dashboards.

- * 1 - Start **Communication plans - No Impact Communication plan**

Step: Investigate

Investigation

This section describes performing investigation of known and unknown symptoms.

Known issue

- * **List all known issues with the application and their standard actions here**

Unknown issues

- * Investigate with the customer and AWS Premium Support.
- * Escalate internally as required.

Step: Mitigation

Collaborate

- * Communicate any changes or important information from the **Investigate** step to the members of the incident call.

Implement mitigation

- * **List customer failover plans / Disaster Recovery plans / etc here for implementing mitigation.**

Step: Recovery

Monitor customer impact

- * Review metrics to confirm recovery.
- * Ensure recovery is across all Availability Zones / Regions / Services
- * Get confirmation from the customer that impact is over and the application has recovered.

****Identify action items****

- * Record key decisions and actions taken, including temporary mitigation that might have been implemented.
- * Ensure outstanding action items have assigned owners.
- * Close out any Communication plans that were opened during the incident with a final confirmation of recovery notification.

Test onboarded workloads in Incident Detection and Response

Note

The AWS Identity and Access Management user or role that you use for alarm testing must have `cloudwatch:SetAlarmState` permission.

The last step in the onboarding process is to perform a gameday for your new workload. After alarm ingestion completes, AWS Incident Detection and Response confirms a date and time of your choosing to start your gameday.

Your gameday serves two main purposes:

- **Functional Validation:** Confirms that AWS Incident Detection and Response can correctly receive your alarm events. And, functional validation confirms that your alarm events trigger the appropriate runbooks and any other desired actions, such as auto case creation if you selected it during alarm ingestion.
- **Simulation:** The gameday is an end to end simulation of what might happen during a real incident. AWS Incident Detection and Response follows your prescribed runbook steps to give you insight into how a real incident might unfold. The gameday is an opportunity for you to ask questions or refine instructions to improve the engagement.

During the alarm test, AWS Incident Detection and Response works with you to remediate any issues identified.

CloudWatch alarms

AWS Incident Detection and Response tests your Amazon CloudWatch alarms by monitoring the state change of your alarm. To do this, manually change the alarm to the **Alarm** state using the AWS Command Line Interface. You can also access the AWS CLI from AWS CloudShell. AWS

Incident Detection and Response provides you with a list of AWS CLI commands for you to use during testing.

To prevent unwanted actions, for example Amazon EC2 instance restarts, disable any CloudWatch alarm actions before you change the alarm state. You can re-enable CloudWatch alarm actions after the testing completes. To learn more about disabling or enabling alarm actions, see [DisableAlarmActions](#) and [EnableAlarmActions](#) in the *Amazon CloudWatch API Reference*.

Example AWS CLI command to set an alarm state:

```
aws cloudwatch set-alarm-state --alarm-name "ExampleAlarm" --state-value ALARM --state-reason "Testing AWS Incident Detection and Response" --region us-east-1
```

To learn more about manually changing the state of CloudWatch alarms, see [SetAlarmState](#).

To learn more about the permissions required for CloudWatch API operations, see [Amazon CloudWatch permissions reference](#).

Third party APM alarms

Workloads that utilize a third party Application Performance Monitoring (APM) tool, such as Datadog, Splunk, New Relic, or Dynatrace, require different instructions to simulate an alarm. At the start of the gameday, AWS Incident Detection and Response requests that you temporarily change your alarm thresholds or comparison operators to force the alarm into the **ALARM** status. This status triggers a payload to AWS Incident Detection and Response.

Key outputs

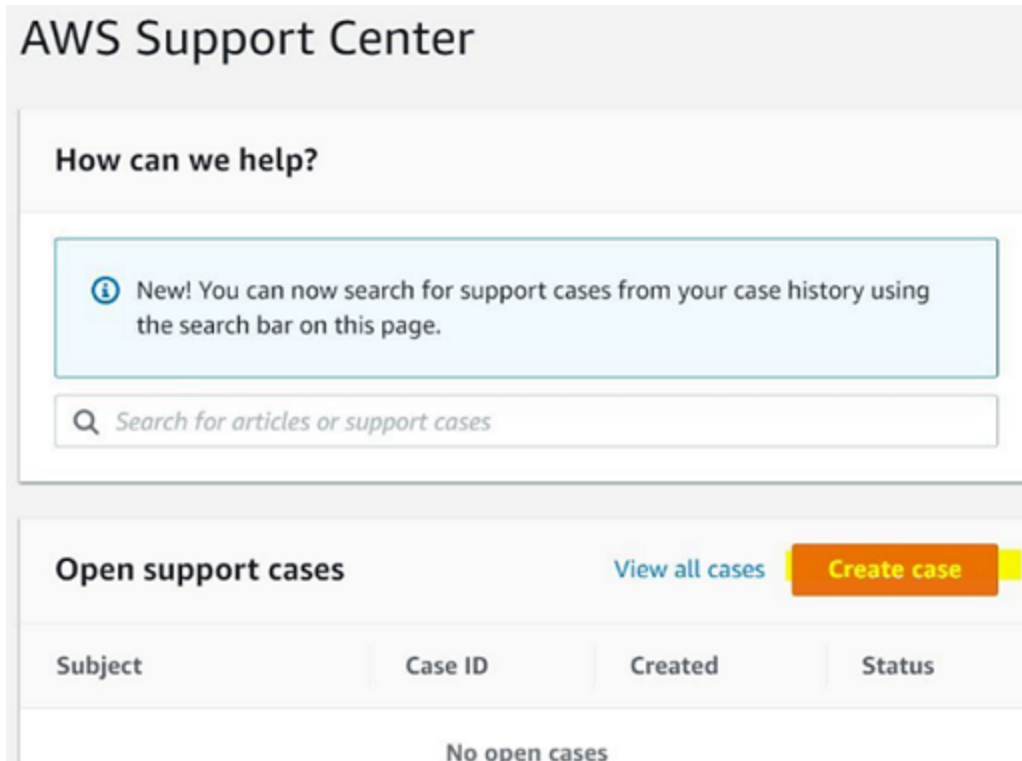
Key outputs:

- Alarm ingestion is successful and your alarm configuration is correct.
- Alarms are successfully created and received by AWS Incident Detection and Response.
- A support case is created for your engagement and your prescribed contacts are notified.
- AWS Incident Detection and Response can engage with you by your prescribed conference means.
- All alarms and support cases generated as part of the gameday are resolved.
- A Go-Live email is sent confirming your workload is now being monitored by AWS Incident Detection and Response.

Request changes to an onboarded workload in Incident Detection and Response

To request changes to an onboarded workload, complete the following steps to create a support case with AWS Incident Detection and Response.

1. Go to the [AWS Support Center](#), and then select **Create case**, as shown in the following example:



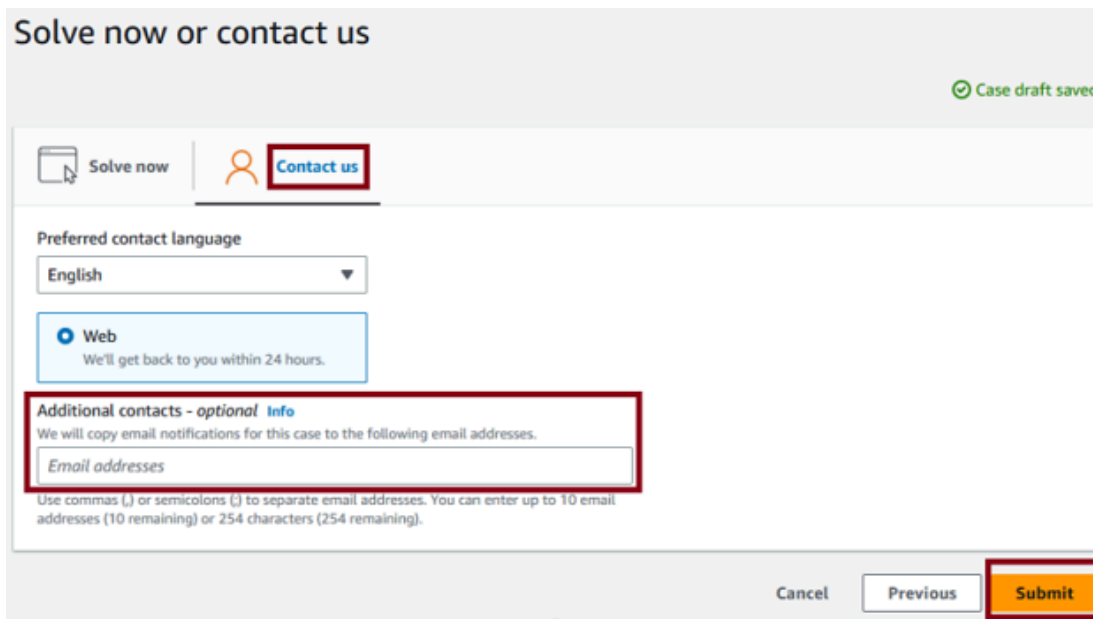
2. Choose **Technical**.
3. For **Service**, choose **Incident Detection and Response**.
4. For **Category**, choose **Workload change request**.
5. For **Severity**, choose **General Guidance**.
6. Enter a **Subject** for this change. For example:

AWS Incident Detection and Response - *workload_name*

7. Enter a **Description** for this change. For example, enter "This request is for changes to an existing workload onboarded into AWS Incident Detection and Response". Make sure that you include the following information in your request:

- **Workload name:** Your workload name.
 - **Account ID(s):** ID1, ID2, ID3, and so on.
 - **Change details:** Enter the details for your requested change.
8. In the **Additional contacts - optional** section, enter any email IDs that you want to receive correspondence about this change.

The following is an example of the **Additional contacts - optional** section.



The screenshot shows a web interface titled "Solve now or contact us" with a green status indicator "Case draft saved". There are two tabs: "Solve now" and "Contact us", with "Contact us" selected. Below the tabs, there is a "Preferred contact language" dropdown menu set to "English". A "Web" option is selected with a radio button, and a note says "We'll get back to you within 24 hours." The "Additional contacts - optional" section is highlighted with a red box and contains an "Info" icon and the text: "We will copy email notifications for this case to the following email addresses." Below this is a text input field labeled "Email addresses" with a placeholder. A note below the field states: "Use commas (,) or semicolons (;) to separate email addresses. You can enter up to 10 email addresses (10 remaining) or 254 characters (254 remaining)." At the bottom right, there are three buttons: "Cancel", "Previous", and "Submit", with "Submit" highlighted in orange.

⚠ Important

Failure to add email IDs in the **Additional contacts - optional** section might delay the change process.

9. Choose **Submit**.

After you submit the change request, you can add additional emails from your organization. To add emails, choose **Reply** in **Case details**, as shown in the following example:

AWS Support > Your support cases > Details

Case ID [redacted] [Info](#) Resolve case

Case details

Subject [redacted]	Status Customer action completed
Case ID [redacted]	Severity General guidance
Created 2023-07-09T02:30:50.234Z	Category Incident Detection and Response, Onboard New Workload
Case type Technical	Additional contacts [redacted]
Opened by [redacted]	

Correspondence Reply

Then, add the email IDs in the **Additional contacts - optional** section.

The following is an example of the **Reply** page showing where you can enter additional emails.

Reply

Do not share any sensitive information in case correspondences, such as credentials, credit cards, signed URLs, or personally identifiable information. Find more information [here](#).

Adding additional email IDs

Maximum 8000 characters (8000 remaining)

Attachments

Choose files

Up to 3 attachments, each less than 5MB.

Contact methods [Info](#)

Web

We'll respond by email and Support Center.

Additional contacts - optional [Info](#)

When we contact you via email, we will copy the correspondence to the following email addresses

[redacted]

Use commas or semicolons to separate email addresses - Maximum 10 email addresses (8 remaining) or 254 characters (213 remaining)

Suppress alarms from engaging Incident Detection and Response

Specify which of your onboarded workload alarms engage with AWS Incident Detection and Response monitoring by suppressing them temporarily or on a schedule. For example, you might temporarily suppress workload alarms during planned maintenance to prevent the alarms from engaging Incident Detection and Response. Or, you might suppress alarms on a schedule if you have daily reboot activity. You can suppress alarms at the alarm source, such as Amazon CloudWatch, or you can submit a workload change request.

Topics

- [Suppress alarms at the alarm source](#)
- [Submit a workload change request to suppress alarms](#)
- [Tutorial: Use a metric math function to suppress an alarm](#)
- [Tutorial: Remove a metric math function to un-suppress an alarm](#)

Suppress alarms at the alarm source

Specify which alarms engage with Incident Detection and Response and when they do so by suppressing alarms at the alarm source.

Topics

- [Use a metric math function to suppress a CloudWatch alarm](#)
- [Remove a metric math function to un-suppress a CloudWatch alarm](#)
- [Example metric math functions and associated use cases](#)
- [Suppress alarms from a third party APM](#)

Use a metric math function to suppress a CloudWatch alarm

To suppress Incident Detection and Response monitoring of Amazon CloudWatch alarms, use a [metric math function](#) to stop CloudWatch alarms from entering the ALARM state during a designated window.

Note

Disabling **Alarm actions** on a CloudWatch alarm doesn't suppress monitoring of your alarms by Incident Detection and Response. Alarm state changes are ingested through Amazon EventBridge, not through CloudWatch alarm actions.

To use a metric math function to suppress a CloudWatch alarm, complete the following steps:

1. Sign in to the AWS Management Console and open the CloudWatch console at <https://console.aws.amazon.com/cloudwatch/>.
2. Choose **Alarms**, and then locate the alarm that you want to add the metric math function to.
3. Choose **Actions**, then select **Edit** to change the alarm.
4. Choose **Edit metric** to modify the metric for the alarm.
5. Choose **Add math, Start with empty expression**.
6. Enter your math expression, then choose **Apply**.
7. Deselect the existing metric that the alarm monitored.
8. Select the expression that you just created, and then choose **Select metric**.
9. Choose **Skip to Preview and create**.
10. Review your changes to make sure that your metric math function is applied as expected, and then choose **Update alarm**.

For a step by step example of suppressing a CloudWatch alarm with a metric math function, see [Tutorial: Use a metric math function to suppress an alarm](#).

For more information on syntax and available functions, see [Metric math syntax and functions](#) in the *Amazon CloudWatch User Guide*.

Remove a metric math function to un-suppress a CloudWatch alarm

Un-suppress a CloudWatch alarm by removing the metric math function. To remove a metric math function from an alarm, complete the following steps:

1. Sign in to the AWS Management Console and open the CloudWatch console at <https://console.aws.amazon.com/cloudwatch/>.

2. Choose **Alarms**, and then locate the alarm or alarms that you want to remove the metric math expression from.
3. In the metric math section, choose **Edit**.
4. To remove the metric from the alarm, choose **Edit** on the metric, and then choose the **x** button next to the metric math expression.
5. Select the original metric, then choose **Select metric**.
6. Choose **Skip to Preview and create**.
7. Review your changes to make sure that your metric math function is applied as expected, then choose **Update alarm**.

Example metric math functions and associated use cases

The following table contains metric math function examples, along with associated use cases and an explanation of each metric component.

Metric math function	Use case	Explanation
IF((DAY(m1) == 2 && HOUR(m1) >= 1 && HOUR(m1) < 3), 0, m1)	Suppress alarm between 1:00 to 3:00 AM UTC every Tuesday by replacing real data points with 0 during this window.	<ul style="list-style-type: none"> • DAY(m1) == 2: Ensures it's Tuesday (Monday = 1, Sunday = 7). • HOUR(m1) >= 1 && HOUR(m1) > 3: Specifies the time range from 1 AM to 3 AM UTC. • IF(condition, value_if_true, value_if_false): If the conditions are true, then replace the metric value with 0. Otherwise, return the original value (m1)
IF((HOUR(m1) >= 23 HOUR(m1) < 4), 0, m1)	Suppress alarm between 11:00 PM to 4:00 AM UTC, daily by replacing real data	<ul style="list-style-type: none"> • HOUR(m1) >= 23: Captures the hours starting at 23:00 UTC.

Metric math function	Use case	Explanation
	points with 0 during this window.	<ul style="list-style-type: none"> • HOURL(m1) < 4: Captures the hours up to (but not including) 04:00 UTC. • : Logical OR ensures the condition applies across two ranges—late-night hours and early-morning hours. • IF(condition, value_if_true, value_if_false): Returns 0 during the specified time range. Retains the original metric value m1 outside that range.
<pre>IF((HOURL(m1) >= 11 && HOURL(m1) < 13), 0, m1)</pre>	<p>Suppress alarm between 11:00 AM to 1:00 PM UTC daily by replacing real data points with 0 during this window.</p>	<ul style="list-style-type: none"> • HOURL(m1) >= 11 && HOURL(m1) < 13: Captures the time range from 11:00 to 13:00 UTC. • IF(condition, value_if_true, value_if_false): If the condition is true (for example, the time is between 11:00 and 13:00 UTC), return 0, If the condition is false, retain the original metric value (m1).

Metric math function	Use case	Explanation
<pre>IF((DAY(m1) == 2 && HOUR(m1) >= 1 && HOUR(m1) < 3), 99, m1)</pre>	<p>Suppress alarm between 1:00 to 3:00 AM UTC every Tuesday by replacing real data points with 99 during this window.</p>	<ul style="list-style-type: none"> • DAY(m1) == 2:: Ensures it's Tuesday (Monday = 1, Sunday = 7). • HOUR(m1) >= 1 && HOUR(m1) < 3: Specifies the time range from 1 AM to 3 AM UTC. • IF(condition, value_if_true, value_if_false): If the conditions are true, replace the metric value with 99. Otherwise, return the original value (m1).
<pre>IF((HOUR(m1) >= 23 HOUR(m1) < 4), 100, m1)</pre>	<p>Suppress alarm between 11:00 PM to 4:00 AM UTC, daily by replacing real data points with 100 during this window.</p>	<ul style="list-style-type: none"> • HOUR(m1) >= 23: Captures the hours starting at 23:00 UTC. • HOUR(m1) < 4: Captures the hours up to (but not including) 04:00 UTC. • : Logical OR ensures the condition applies across two ranges—late-night hours and early-morning hours. • IF(condition, value_if_true, value_if_false): Returns 100 during the specified time range. Retains the original metric value m1 outside that range.

Metric math function	Use case	Explanation
<code>IF((HOUR(m1) >= 11 && HOUR(m1) < 13), 99, m1)</code>	Suppress alarm between 11:00 AM to 1:00 PM UTC daily by replacing real data points with 99 during this window.	<ul style="list-style-type: none"> • HOUR(m1) >= 11 && HOUR(m1) < 13: Captures the time range from 11:00 to 13:00 UTC. • IF(condition, value_if_true, value_if_false): If the condition is true (for example, the time is between 11:00 and 13:00 UTC), return 99. If the condition is false, retain the original metric value (m1).

Suppress alarms from a third party APM

Refer to your third party APM vendor's documentation for instructions on how to suppress alarms. Examples of third party APM vendors are New Relic, Splunk, Dynatrace, Datadog, and SumoLogic.

Submit a workload change request to suppress alarms

If you can't suppress alarms at the source as described in the previous section, then submit a Workload Change Request to instruct Incident Detection and Response to manually suppress monitoring of some or all of your workload's alarms.

For detailed instructions on how to create a Workload Change Request, see [Request changes to an onboarded workload in Incident Detection and Response](#). When raising a Workload Change Request to request suppression of your alarms, make sure that you provide the following required information

- **Workload name:** Your workload name.
- **Account ID(s):** ID1, ID2, ID3, and so on.
- **Change details:** Alarm Suppression
- **Suppression start time:** Date, time, and time zone.
- **Suppression end time:** Date, time, and time zone.

- **Alarms to suppress:** A list of CloudWatch alarm ARNs or third party APM event identifiers to suppress.

After you create the alarm suppression Workload Change Request, you receive the following notifications from Incident Detection and Response:

- Acknowledgement of your Workload Change Request.
- Notification when alarms are suppressed.
- Notification when alarms are re-enabled for monitoring.

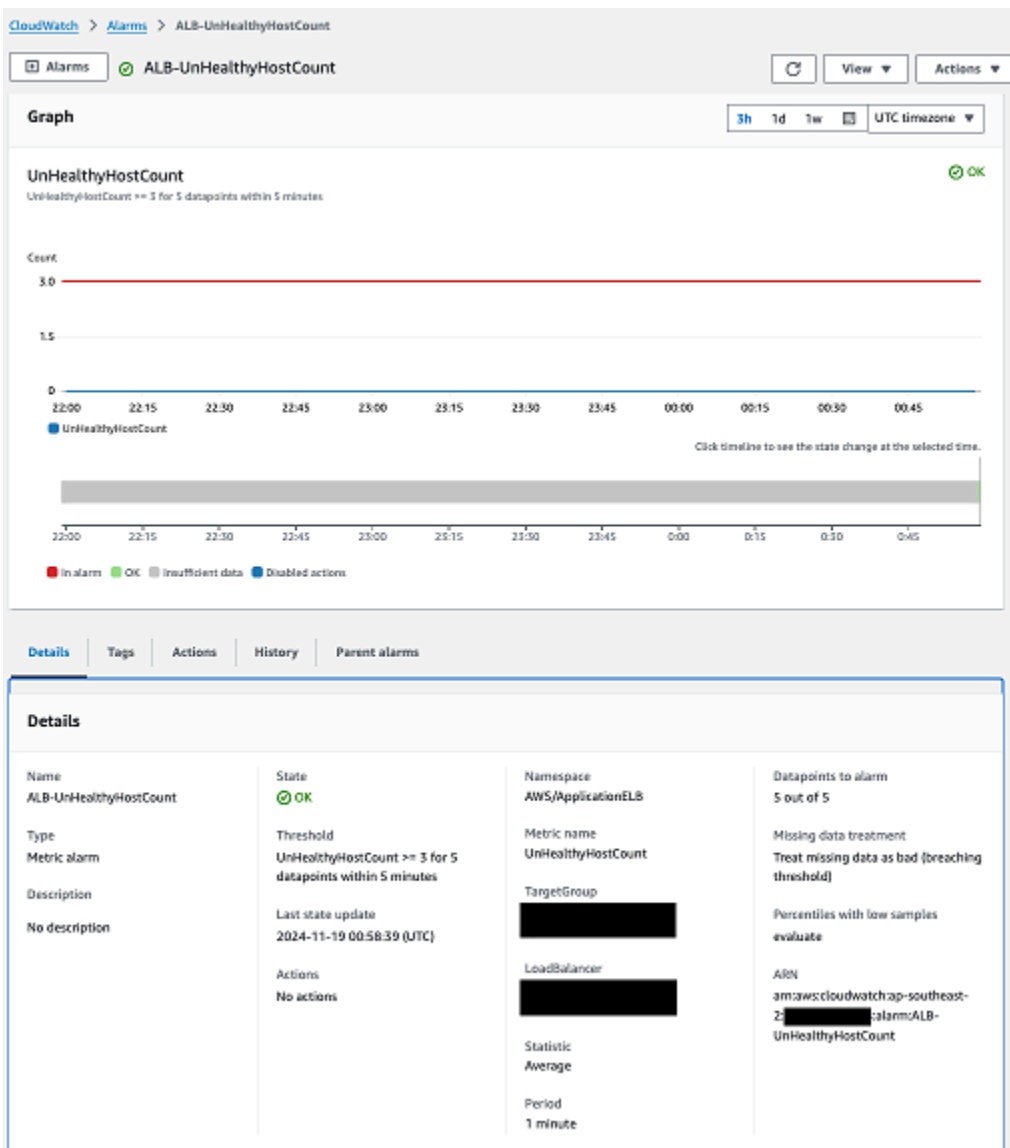
Tutorial: Use a metric math function to suppress an alarm

The following tutorial walks you through how to suppress a CloudWatch alarm using metric math.

Example scenario

There's a planned activity that takes place between 1:00 to 3:00 AM UTC on the upcoming Tuesday. You want to create a CloudWatch metric math function that replaces the real data points during this time, with 0 (a data point that falls below the set threshold).

1. Assess the criteria that causes your alarm to trigger. The following screenshot provides an example of alarm criteria:



The alarm shown in the preceding screenshot monitors the `UnHealthyHostCount` metric for an Application Load Balancer target group. This alarm enters the ALARM state when the `UnHealthyHostCount` metric is greater than or equal to 3 for 5 out of 5 data points. The alarm treats missing data as bad (breaching the configured threshold).

2. Create the metric math function.

In this example, the planned activity takes place between 1:00 to 3:00 AM UTC on the upcoming Tuesday. So, create a CloudWatch metric math function that replaces the real data points during this time, with 0 (a data point that falls below the set threshold).

Note that the replacement data point that you must configure differs depending on your alarm configuration. For example, if you have an alarm that monitors HTTP success rate, with a

threshold of less than 98, then replace your real data points during the planned activity with a value above the configured threshold, 100. The following is an example metric math function for this scenario.

```
IF((DAY(m1) == 2 && HOUR(m1) >= 1 && HOUR(m1) < 3), 0, m1)
```

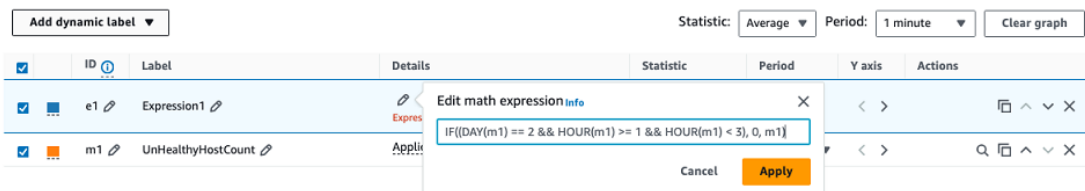
The preceding metric math function contains the following elements:

- **DAY(m1) == 2**: Ensures that it's Tuesday (Monday = 1, Sunday = 7).
- **HOUR(m1) >= 1 && HOUR(m1) < 3**: Specifies the time range from 1 AM to 3 AM UTC.
- **IF(condition, value_if_true, value_if_false)**: If the conditions are true, the function replaces the metric value with 0. Otherwise, the original value (m1) is returned.

For additional information on syntax and available functions, see [Metric math syntax and functions](#) in the *Amazon CloudWatch User Guide*

3. Sign in to the AWS Management Console and open the CloudWatch console at <https://console.aws.amazon.com/cloudwatch/>.
4. Choose **Alarms**, and then locate the alarm that you want to add the metric math function to.
5. In the metric math section, choose **Edit**.
6. Choose **Add math, Start with empty expression**.
7. Enter your math expression, and then choose **Apply**.

The existing metric that the alarm monitors automatically becomes **m1** and your math expression is **e1**, as shown in the following example:



8. (Optional) Edit the label of the metric math expression to help others understand it's function and why it was created, as shown in the following example:

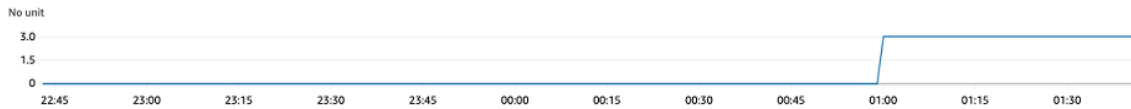
ID	Label	Details	Statistic	Period	Y axis	Actions
e1	Suppress alarm 1-3AM UTC planned maintenance	IF((DAY(m1) == 2 && HOUR(m1) >= 1 && HOUR(m1) < 3), 0, m1)				< > [] ^ v X
m1	UnHealthyHostCount	ApplicationELB - UnHealthyHostCount * 1	Average	1 minute		< > [] ^ v X

9. Deselect **m1**, select **e1**, and then choose **Select metric**. This sets the alarm to monitor the math expression instead of the underlying metric directly.

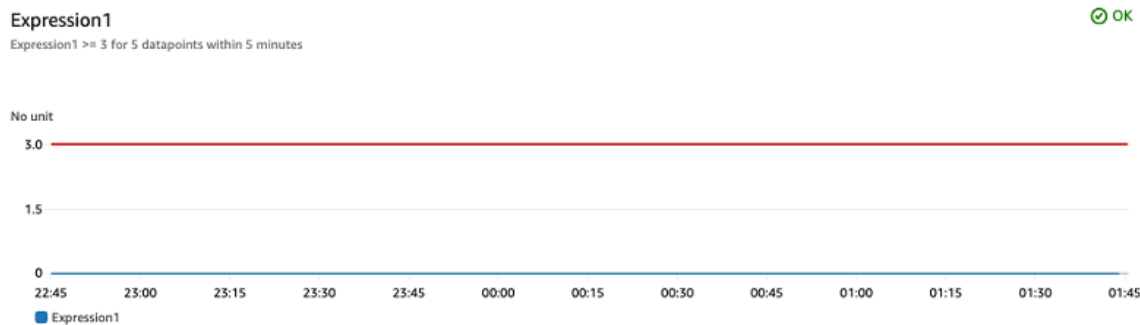
10. Choose **Skip to Preview and create**.

11. Validate that the alarm is configured as expected, then choose **Update alarm to save the change**.

In the preceding example, without the metric math function applied, the real `UnHealthyHostCount` metric would have been reported during the planned activity. This would have resulted in the CloudWatch alarm entering the ALARM state and engaging Incident Detection and Response, as shown in the following example:



With the metric math function in place, the real data points are replaced with 0 during the activity, and the alarm remains in the OK state, suppressing Incident Detection and Response engagement.



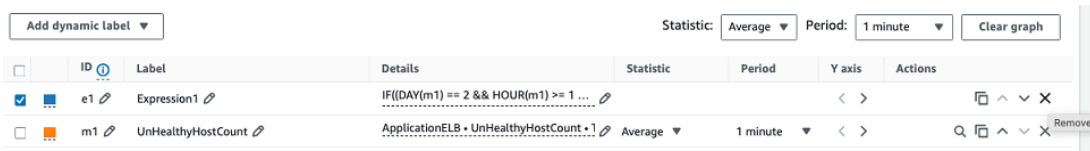
Tutorial: Remove a metric math function to un-suppress an alarm

If you suppress a CloudWatch alarm for a one-time activity, remove the metric math function from the alarm after the activity completes to resume regular monitoring of the alarm. To suppress the alarm on a regular schedule, for example, if you have a scheduled weekly patching routine that results in instance reboots on the same day and time each week, then leave the metric math function in place.

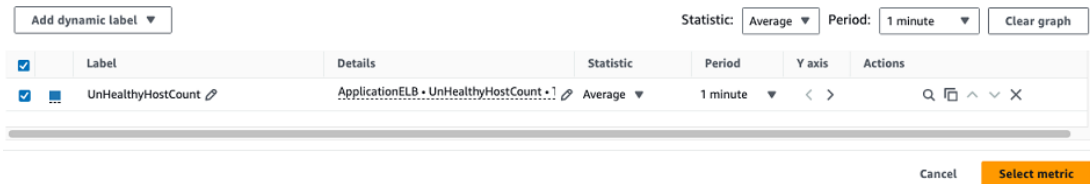
The following tutorial walks you through how to remove a metric math function to un-suppress a CloudWatch alarm

1. Sign in to the AWS Management Console and open the CloudWatch console at <https://console.aws.amazon.com/cloudwatch/>.
2. Choose **Alarms**, and then locate the alarm that you want to add the metric math function to.
3. In the metric math section, choose **Edit**.

- To remove the suppression from the alarm, select the **x** button next to the metric math expression.



- Select the metric to resume monitoring of the real metric. then choose **Select metric**.



- Choose **Skip to Preview and create**.
- Validate that the alarm is configured as expected, then choose **Update alarm to save the change**.

Offboard a workload from Incident Detection and Response

To offboard a workload from AWS Incident Detection and Response, create a new support case for each workload. When you create the support case, keep the following in mind:

- To offboard a workload that's in a single AWS account, create the support case either from the workload's account or from your payer account.
- To offboard a workload that spans multiple AWS accounts, then create the support case from your **payer account**. In the body of the support case, list all account IDs to offboard.

⚠ Important

If you create a support case to offboard a workload from the incorrect account, you might experience delays and requests for additional information before your workloads can be offloaded.

Request to offboard a workload

- Go to the [AWS Support Center](#), and then select **Create case**.

2. Choose **Technical**.
3. For **Service**, choose **Incident Detection and Response**.
4. For **Category**, choose **Workload Offboarding**.
5. For **Severity**, choose **General Guidance**.
6. Enter a **Subject** for this change. For example:

[Offboard] AWS Incident Detection and Response - *workload_name*

7. Enter a **Description** for this change. For example, enter "This request is for offboarding an existing workload onboarded into AWS Incident Detection and Response". Make sure that you include the following information in your request:
 - **Workload name:** Your workload name.
 - **Account ID(s):** ID1, ID2, ID3, and so on.
 - **Reason for offboarding:** Provide a reason for offboarding the workload.
8. In the **Additional contacts - optional** section, enter any email IDs that you want to receive correspondence about this offboarding request.
9. Choose **Submit**.

AWS Incident Detection and Response monitoring and observability

AWS Incident Detection and Response offers you expert guidance on defining observability across your workloads from the application layer to the underlying infrastructure. Monitoring tells you that something is wrong. Observability uses data collection to tell you what is wrong and why it happened.

The Incident Detection and Response system monitors your AWS workloads for failures and performance degradation by leveraging native AWS services such as Amazon CloudWatch and Amazon EventBridge to detect events that may impact your workload. Monitoring provides you notification of imminent, on-going, receding, or potential failures or of performance degradation. When you onboard your account to Incident Detection and Response, you select which alarms in your account should be monitored by the Incident Detection and Response monitoring system and you associate those alarms with an application and a runbook used during incident management.

Incident Detection and Response uses Amazon CloudWatch and other AWS services to build your observability solution. AWS Incident Detection and Response helps you with observability in two ways:

- **Business Outcome metrics:** Observability on AWS Incident Detection and Response starts with defining the key metrics that monitor the outcomes of your workloads or end-user experience. AWS experts work with you to understand the objectives of your workload, the key outputs or factors that may impact user-experience, and to define the metrics and alerts that capture any degradation in those key metrics. For example a key business metric for a mobile calling application is the *Call Setup Success Rate* (monitors the success rate of user call attempts), and a key metric for a website is *page speed*. Incident engagement is triggered based on business outcome metrics.
- **Infrastructure level metrics:** At this stage, we identify the underlying AWS services and infrastructure supporting your application and define metrics and alarms to track the performance of these infrastructure services. These may include metrics such as `ApplicationLoadBalancerErrorCount` for Application Load Balancer instances. This starts after the workload has been onboarded and monitoring set up.

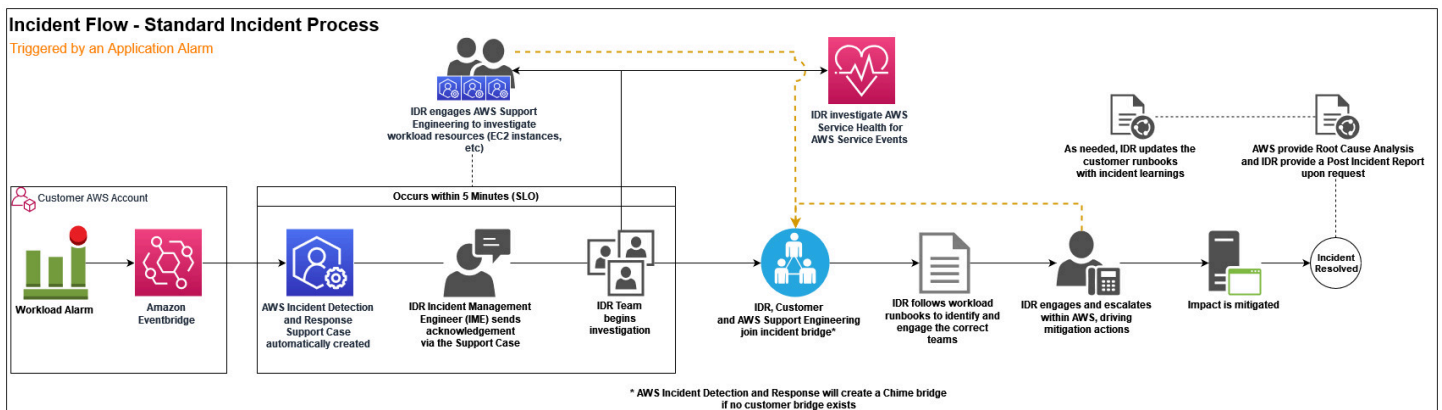
Implementing observability on AWS Incident Detection and Response

Because observability is a continuous process that may not be completed in one exercise or time frame, AWS Incident Detection and Response implements observability in two phases:

- **Onboarding phase:** Observability during onboarding is focused on detecting when the business outcomes of your application are impaired. To this end, observability during the onboarding phase is focused on defining the key business outcome metrics at the application layer to notify AWS of disruptions to your workloads. This way AWS can promptly respond to these disruption and provide you help toward recovery. To learn more about using the AWS Incident Detection and Response Customer Command Line Interface to help automate these steps, see [CLI for AWS Incident Detection and Response](#).
- **Post-onboarding phase:** AWS Incident Detection and Response offers a number of proactive services for observability including the definition of infrastructure level metrics, metric tuning, and setting up traces and logs depending, on the maturity level of the customer. The implementation of these services may span several months and involve multiple teams. AWS Incident Detection and Response provides guidance on observability setup and customers are required to implement the required changes in their workload environment. For help with hands-on implementation of observability features, raise a request to your technical account managers (TAMs).

Incident management with Incident Detection and Response

AWS Incident Detection and Response offers you 24 hours a day, 7 days a week proactive monitoring and incident management delivered by a designated team of incident managers. The following diagram outlines the standard incident management process when an application alarm triggers an incident, including alarm generation, AWS Incident Manager engagement, incident resolution, and post-incident review.



- 1. Alarm generation:** Alarms triggered on your workloads are pushed through Amazon EventBridge to AWS Incident Detection and Response. AWS Incident Detection and Response automatically pulls up the runbook associated with your alarm and notifies an incident manager. If a critical incident occurs on your workload that isn't detected by alarms monitored by AWS Incident Detection and Response, then you can create a support case to request an Incident Response. For more information on requesting an Incident Response, see [Request an Incident Response](#).
- 2. AWS Incident Manager engagement:** The incident manager responds to the alarm and engages you on a conference call or as otherwise specified in the runbook. The incident manager verifies the health of the AWS services to determine if the alarm is related to issues with AWS services used by the workload and advises on the status of the underlying services. If required, the incident manager then creates a case on your behalf and engages the right AWS experts for support. Because AWS Incident Detection and Response monitors AWS services specifically for your applications, AWS Incident Detection and Response might determine that the incident is related to an AWS service issue before an AWS service event is declared. In this scenario, the incident manager advises you on the status of the AWS service, triggers the AWS service event incident management workflow, and follows up with the service team on resolution.

The information provided gives you the opportunity to implement your recovery plans or workarounds early to mitigate the impact of the AWS service event.

Sometimes alarms trigger and quickly recover. In this scenario, the incident manager sends a case correspondence stating the alarm has recovered, but doesn't engage you. However, if an alarm triggers more than once within 15 minutes, the incident manager engages you per your runbook instructions, even if the alarm recovers.

- 3. Incident resolution:** The incident manager coordinates the incident across the required AWS teams and makes sure that you remain engaged with the right AWS experts until the incident is mitigated or resolved.
- 4. Post Incident Review (if requested):** After an incident, AWS Incident Detection and Response can perform a post incident review at your request and generate a Post Incident Report. The Post Incident Report includes a description of the issue, the impact, which teams were engaged, and workarounds or actions taken to mitigate or resolve the incident. The Post Incident Report might contain information that can be used to reduce the likelihood of incident recurrence, or to improve the management of a future occurrence of a similar incident. The Post Incident Report isn't a Root Cause Analysis (RCA). You can request a RCA in addition to the Post Incident Report. An example of a Post Incident Report is provided in the following section.

Important

The following report template is an example only.

Post ** Incident ** Report ** Template

Post Incident Report - 0000000123

Customer: Example Customer

AWS Support case ID(s): 0000000000

Customer internal case ID (if provided): 1234567890

Incident start: 2023-02-04T03:25:00 UTC

Incident resolved: 2023-02-04T04:27:00 UTC

Total Incident time: 1:02:00 s

Source Alarm ARN: arn:aws:cloudwatch:us-east-1:000000000000:alarm:alarm-prod-workload-impaired-useast1-P95

Problem Statement:

Outlines impact to end users and operational infrastructure impact.

Starting at 2023-02-04T03:25:00 UTC, the customer experienced a large scale outage of their workload that lasted one hour and two minutes and spanning across all Availability Zones where the application is deployed. During impact, end users were unable to connect to the workload's Application Load Balancers (ALBs) which service inbound communications to the application.

Incident Summary:

Summary of the incident in chronological order and steps taken by AWS Incident Managers to direct the incident to a path to mitigation.

At 2023-02-04T03:25:00 UTC, the workload impairments alarm triggered a critical incident for the workload. AWS Incident Detection and Response Managers responded to the alarm, checking AWS service health and steps outlined in the workload's runbook.

At 2023-02-04T03:28:00 UTC, ** per the runbook, the alarm had not recovered and the Incident Management team sent the engagement email to the customer's Site Reliability Team (SRE) team, created a troubleshooting bridge, and an Support support case on behalf of the customer.

At 2023-02-04T03:32:00 UTC, ** the customer's SRE team, and Support Engineering joined the bridge. The Incident Manager confirmed there was no on-going AWS impact to services the workload depends on. The investigation shifted to the specific resources in the customer account.

At 2023-02-04T03:45:00 UTC, the Cloud Support Engineer discovered a sudden increase in traffic volume was causing a drop in connections. The customer confirmed this ALB was newly provisioned to handle an increase in workload traffic for an on-going promotional event.

At 2023-02-04T03:56:00 UTC, the customer instituted back off and retry logic. The Incident Manager worked with the Cloud Support Engineer to raise an escalation a higher support level to quickly scale the ALB per the runbook.

At 2023-02-04T04:05:00 UTC, ALB support team initiates scaling activities. The back-off/retry logic yields mild recovery but timeouts are still being seen for some clients.

By 2023-02-04T04:15:00 UTC, scaling activities complete and metrics/alarms return to pre-incident levels. Connection timeouts subside.

At 2023-02-04T04:27:00 UTC, per the runbook the call was spun down, after 10 minutes of recovery monitoring. Full mitigation is agreed upon between AWS and the customer.

Mitigation:

Describes what was done to mitigate the issue. NOTE: this is not a Root Cause Analysis (RCA).

Back-off and retries yielded mild recovery. Full mitigation happened after escalation to ALB support team (per runbook) to scale the newly provisioned ALB.

Follow up action items (if any):

Action items to be reviewed with your Technical Account Manager (TAM), if required.

Review alarm thresholds to engage AWS Incident Detection and Response closer to the time of impact.

Work with AWS Support and TAM team to ensure newly created ALBs are pre-scaled to accommodate expected spikes in workload traffic.

Topics

- [Provision access to AWS Support Center Console for application teams](#)
- [Request an Incident Response](#)
- [Manage Incident Detection and Response support cases with the AWS Support App in Slack](#)

Provision access to AWS Support Center Console for application teams

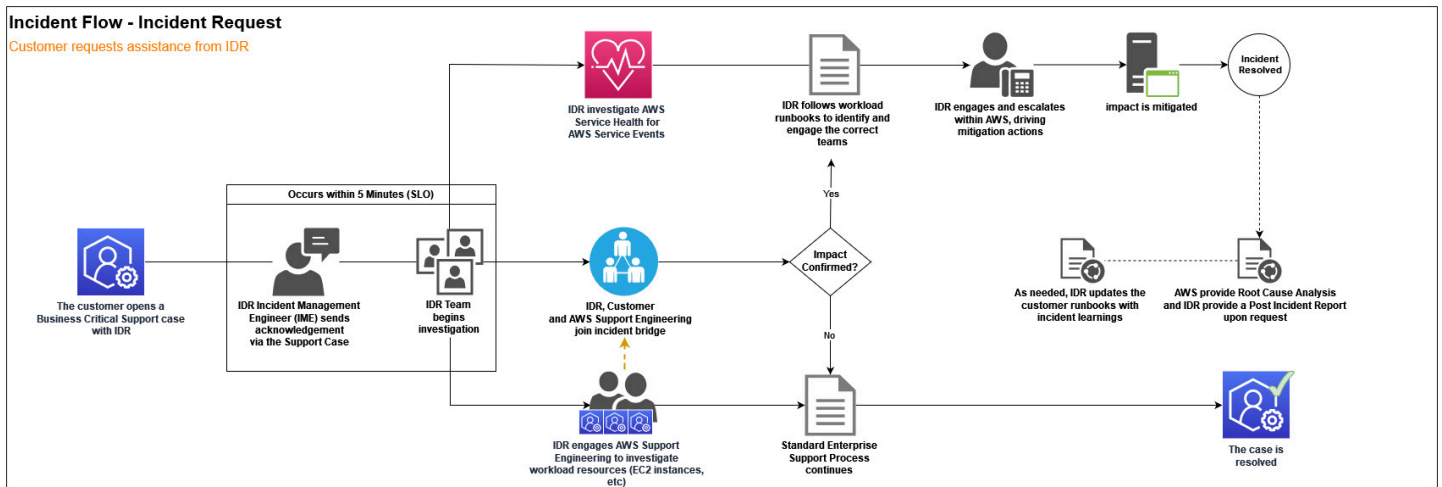
AWS Incident Detection and Response communicates with you through Support cases during the lifecycle of an incident. To correspond with Incident Managers, your teams must have access to the Support Center.

For more information on provisioning access, see [Manage access to Support Center](#) in the *Support User Guide*.

Request an Incident Response

If a critical incident occurs on your workload that isn't detected by alarms monitored by AWS Incident Detection and Response, you can create a support case to request an Incident Response. You can request an Incident Response for any workload that's subscribed to AWS Incident Detection and Response, including workloads in the process of onboarding, using the AWS Support Center Console, AWS Support API, or AWS Support App in Slack.

The following diagram illustrates the end-to-end workflow for an AWS customer requesting incident assistance from the Incident Detection and Response team, detailing the steps from the initial request through investigation, mitigation, and resolution.



To request an Incident Response for an incident that's actively impacting your workload, create an Support case. After the support case is raised, AWS Incident Detection and Response engages you on a conference bridge with the AWS experts required to accelerate the recovery of your workload.

Request an Incident Response using the AWS Support Center Console

To request an incident response, complete the following steps:

1. Open the [AWS Support Center Console](#) to create a new support case.
2. For **Subject**, enter a brief summary of the incident. For example, AWS Incident Detection and Response - Active Incident - workload_name.
3. For **Description**, enter the details of the incident. We recommend that you include the following details in your support case:
 - Affected AWS resource ARN(s), workload name and its function
 - Description of impact to the business
 - (Optional) Your preferred conference bridge URL. If you don't provide bridge details, AWS Incident Detection and Response creates an AWS conference bridge and sends you an invitation with the bridge URL.
4. (Optional) Attach files that can help describe the incident, such as screenshots or log excerpts.
5. Configure the following case classification fields:
 - **Case type: Technical**
 - **Service: Incident Detection and Response**
 - **Category: Active Incident**

- **Severity: Business-critical system down**
6. Provide additional context to help AWS Incident Detection and Response engage AWS experts faster, such as the impacted AWS service, impacted AWS Region, business impact, impact start time, and affected resources.
 7. Choose **Submit**.
 8. AWS Incident Detection and Response acknowledges your case within five minutes and engages you on a conference bridge with the appropriate AWS experts.

Request an Incident Response using the AWS Support API

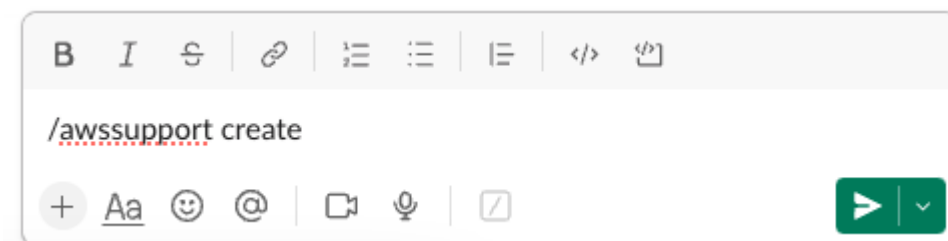
You can use the AWS Support API to programmatically create support cases. For more information, see [About the AWS Support API](#) in the *AWS Support User Guide*.

Request an Incident Response using the AWS Support App in Slack

To use the AWS Support App in Slack to request an Incident Response, complete the following steps:

1. Open the Slack channel that you configured the AWS Support App in Slack in.
2. Enter the following command:

```
/awssupport create
```



3. Enter a **Subject** for this incident. For example, enter **AWS Incident Detection and Response - Active Incident - workload_name**.
4. Enter the **Problem Description** for this incident. Add the following details:

Technical Information:

Affected Service(s):

Affected Resource(s):

Affected Region(s):

Workload Name:

Business Information:

Description of impact to the business:

[Optional] Customer Bridge Details:

5. Choose **Next**.

The screenshot shows the 'Create a support case' dialog box in the AWS Support App. It is titled 'Step 1 of 3'. The main heading is 'Create a support case'. Below the heading, there is a sub-heading 'Step 1 of 3'. The text reads: 'You can create a case with AWS Support for technical and account-related issues.' The 'Subject' field contains the text 'AWS Incident Detection and Response - Active Incident - workl'. The 'Description' field is a large text area with a blue border, containing the following text: 'Technical Information: 2302', 'Affected Service(s):', 'Affected Resource(s):', 'Affected Region(s):', 'Workload Name:', 'Business Information:', 'Description of impact to the business:', and '[Optional] Customer Bridge Details:'. Below the description field, there is a note: 'Note: You can add attachments after step 3 when you confirm the case.' At the bottom of the dialog, there are two buttons: 'Cancel' and 'Next'.

aws **Create a support case**

Step 1 of 3

You can create a case with AWS Support for technical and account-related issues.

Subject

AWS Incident Detection and Response - Active Incident - workl

Description

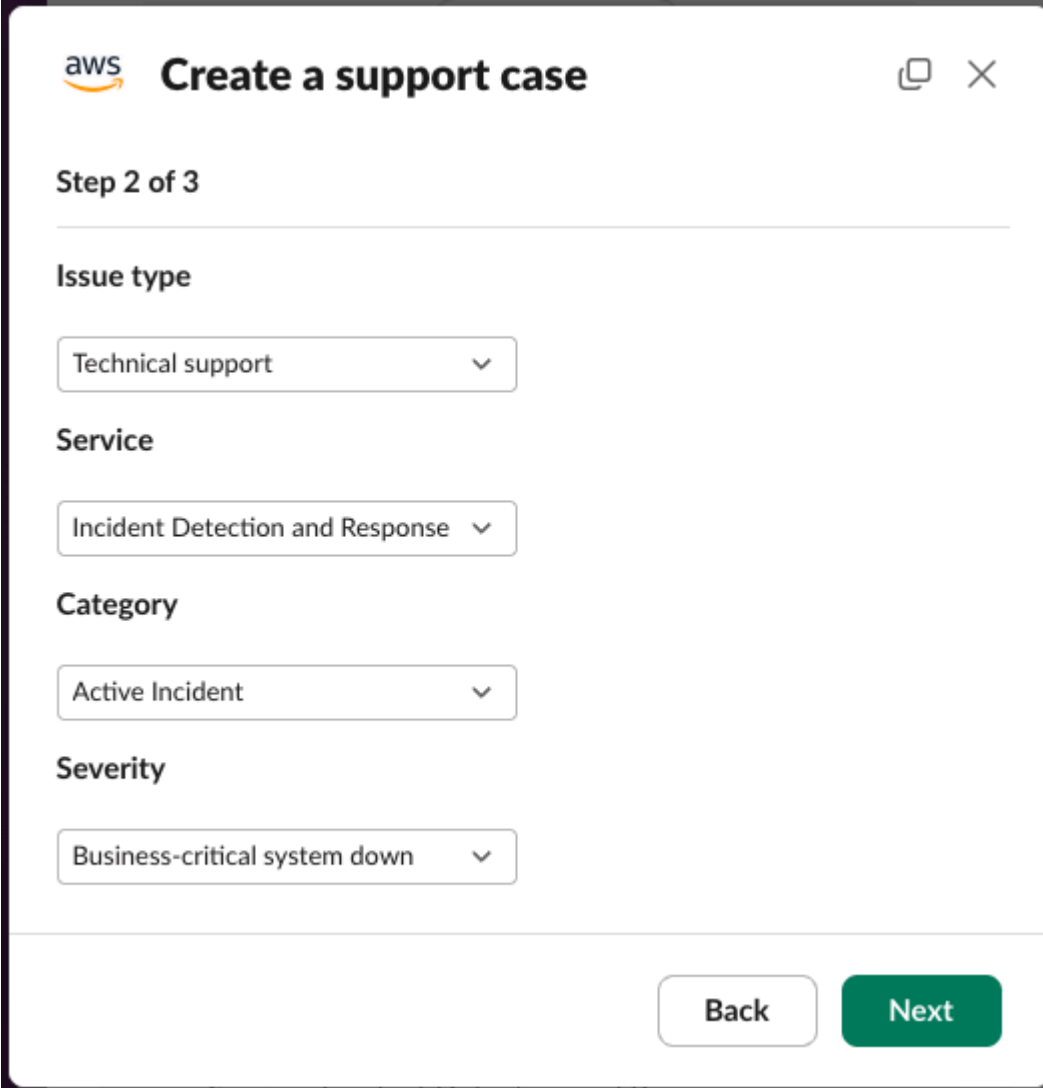
Technical Information: 2302
Affected Service(s):
Affected Resource(s):
Affected Region(s):
Workload Name:

Business Information:
Description of impact to the business:
[Optional] Customer Bridge Details:

Note: You can add attachments after step 3 when you confirm the case.

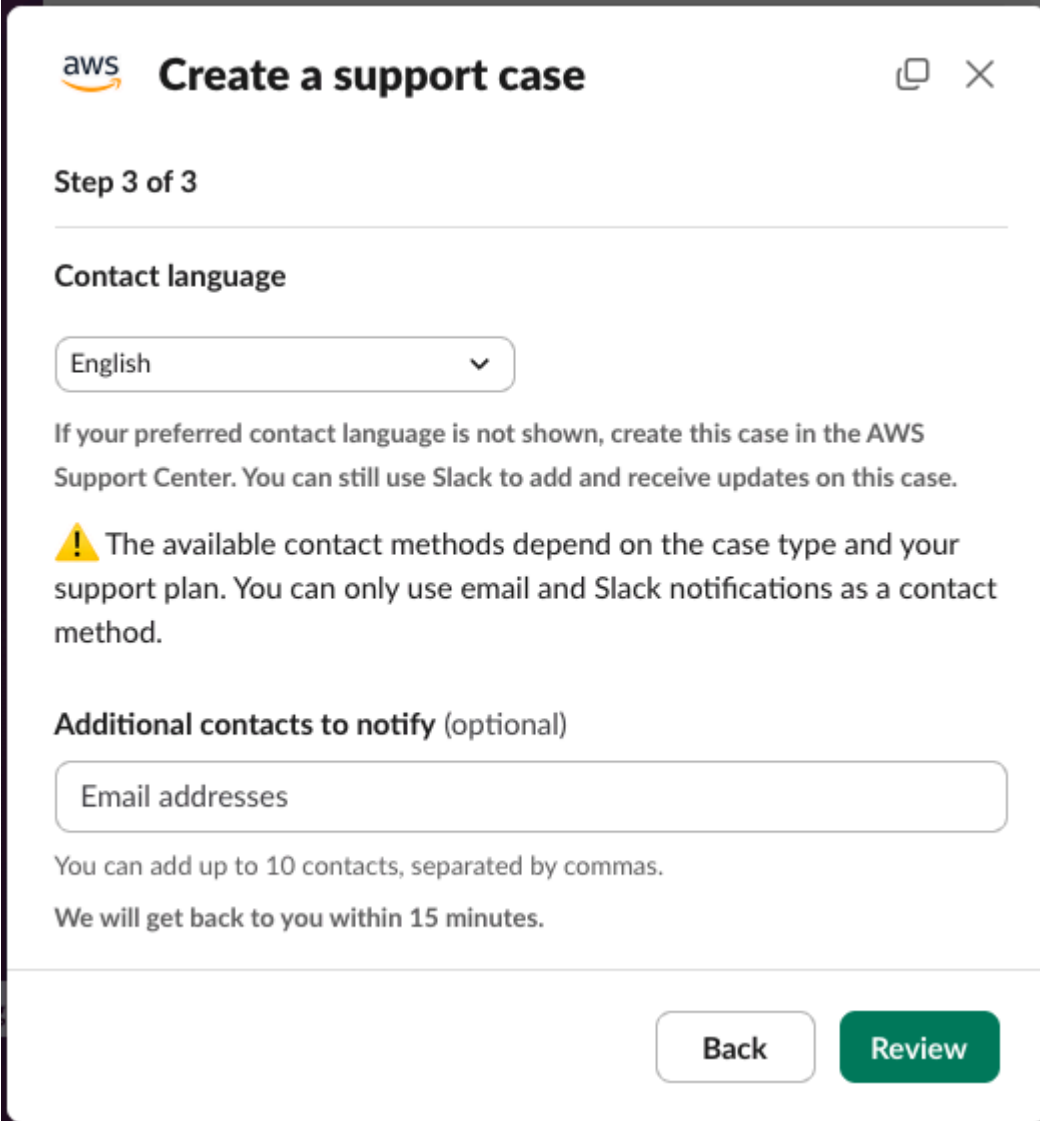
Cancel Next

6. For **Issue Type**, choose **Technical support**.
7. For **Service**, choose **Incident Detection and Response**.
8. For **Category**, choose **Active Incident**.
9. For **Severity**, choose **Business-critical system down**.



The screenshot shows the 'Create a support case' interface in the AWS Support App. It is titled 'Step 2 of 3'. The form contains four dropdown menus: 'Issue type' set to 'Technical support', 'Service' set to 'Incident Detection and Response', 'Category' set to 'Active Incident', and 'Severity' set to 'Business-critical system down'. At the bottom right, there are two buttons: a white 'Back' button and a green 'Next' button.

10Optionally enter up to 10 additional contacts in the **Additional contacts to notify** field, separated by commas. These additional contacts receive copies of email correspondence about this incident.



aws Create a support case

Step 3 of 3

Contact language

English

If your preferred contact language is not shown, create this case in the AWS Support Center. You can still use Slack to add and receive updates on this case.

! The available contact methods depend on the case type and your support plan. You can only use email and Slack notifications as a contact method.

Additional contacts to notify (optional)

Email addresses

You can add up to 10 contacts, separated by commas.

We will get back to you within 15 minutes.


Back Review

11. Choose **Review**.

12. A new message that is only visible to you appears in the Slack channel. Review the case details, then choose **Create case**.

Today ▾ New

👁 Only visible to you

 **AWS Support** APP 15:35

Review your case details and then choose **Create case**.

Case subject: AWS Incident Detection and Response - Active Incident - workload_name

Description: Technical Information:

Affected Service(s):

Affected Resource(s):

Affected Region(s):

Workload Name:

Business Information:

Description of impact to the business:

[Optional] Customer Bridge Details:

- **AWS account:**
- **Issue type:** Technical support
- **Service:** Incident Detection and Response
- **Category:** Active Incident
- **Severity:** Business-critical system down
- **Language:** English

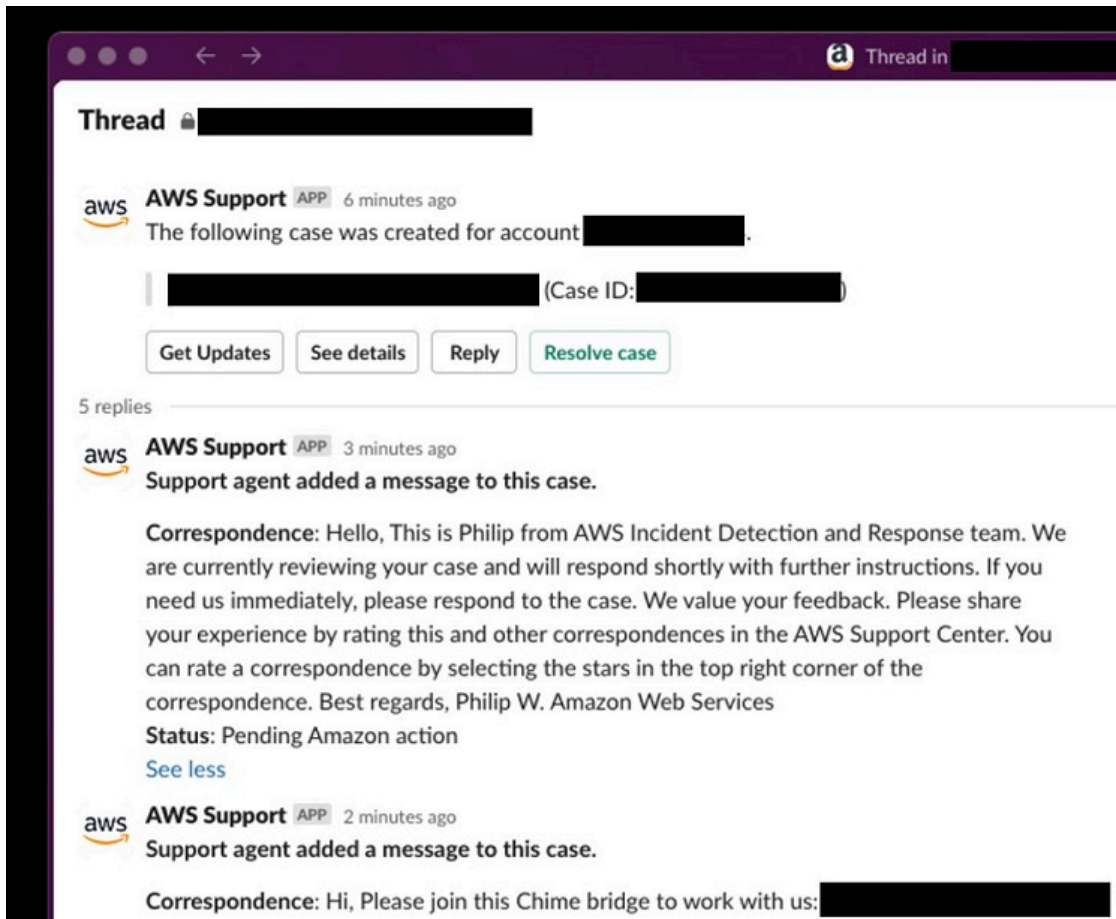
We will contact you by email and Slack notifications within 15 minutes.

Additional contacts to notify: None

13>Your Case ID is provided in a new message from the AWS Support App in Slack.

14 Incident Detection and Response acknowledges your case within 5 minutes and engages you on a conference bridge with the appropriate AWS experts.

15 Correspondence from Incident Detection and Response is updated in the case thread.



Manage Incident Detection and Response support cases with the AWS Support App in Slack

With the [AWS Support App in Slack](#), you can manage your Support cases in Slack, receive notifications about new [alarm initiated incidents](#) on your AWS Incident Detection and Response workload, and create [Incident Response Requests](#).

To configure the AWS Support App in Slack, follow the instructions provided in the [Support User Guide](#).

Important

- To receive notifications in Slack for all alarm initiated incidents on your workload, you must configure the AWS Support App in Slack for all your workload's accounts that are onboarded to AWS Incident Detection and Response. Support cases are created in the account that the workload alarm originated in.

- Multiple high-severity support cases can be opened on your behalf during an incident to engage Support resolvers. You receive notifications in Slack for all support cases that are opened during an incident that match your [notification configuration for the Slack channel](#).
- Notifications that you receive through the AWS Support App in Slack don't replace your workload's initial and escalation contacts that are engaged via email or phone call by AWS Incident Detection and Response during an incident.

Topics

- [Alarm-initiated incident notifications in Slack](#)
- [Create an Incident Response Request in Slack](#)

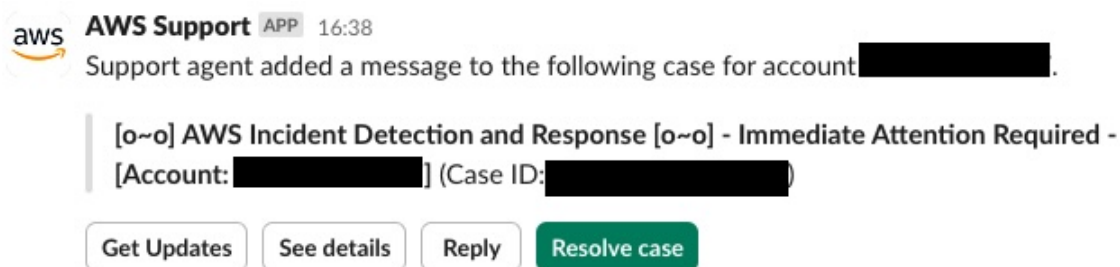
Alarm-initiated incident notifications in Slack

After you configure the AWS Support App in Slack in your Slack channel, you receive notifications about alarm initiated incidents on your AWS Incident Detection and Response monitored workload.

The following example shows how notifications for alarm initiated incidents appear in Slack.


Example notification

When your alarm initiated incident is acknowledged by AWS Incident Detection and Response, a notification similar to the following generates in Slack:



To view the full correspondence added by AWS Incident Detection and Response, choose **See details**.

Only visible to you

 **AWS Support** APP 16:38

This case was created on [REDACTED].

Case subject: [o~o] AWS Incident Detection and Response [o~o] - Immediate Attention Required - [Account: [REDACTED]] (Case ID: [REDACTED])

Description: Hello,

This is Michelle from AWS Incident Detection and Response. An alarm has triggered for your workload Example Workload Name. I am currently investigating and will update you in a few minutes once I have finished initial investigation.

Alarm Identifier - arn:aws:cloudwatch:us-east-1:1234567891012:alarm:ExampleAlarm

We value your feedback. Please share your experience by rating this and other correspondences in the AWS Support Center. You can rate a correspondence by selecting the stars in the top right corner of the correspondence.


Best regards,
Michelle K.
Amazon Web Services

- **Status:** Pending Amazon action
- **AWS account:** [REDACTED]
- **Issue type:** Technical support
- **Service:** -
- **Category:** -
- **Severity:** Business-critical system down
- **Language:** English

Correspondence:
No correspondences have been added yet.

[Share to channel](#) [Reply](#) [Resolve case](#)


Further updates from AWS Incident Detection and Response appear in the case's thread.

 **AWS Support** APP 2 minutes ago
Support agent added a message to the following case for account [REDACTED]

[o~o] AWS Incident Detection and Response [o~o] - Immediate Attention Required -
[Account: [REDACTED]] (Case ID: [REDACTED])

[Get Updates](#) [See details](#) [Reply](#) [Resolve case](#)

1 reply

 **AWS Support** APP Just now
Support agent added a message to this case.

Correspondence: The following alarm has engaged AWS Incident Detection and Response to an Incident bridge: Alarm Identifier - arn:aws:cloudwatch:us-east-1:1234567891012:alarm:ExampleAlarm Alarm State Change Reason - Threshold Crossed: 3 out of the last 5 datapoints [642.4 (26/09/24 04:51:00), 504.0 (26/09/24 04:52:00), 203.8 (26/09/24 04:55:00)] were greater than the threshold (150.0) (minimum 3 datapoints for OK -> ALARM transition). Alarm Start Time - 26 September 2024 04:55 AM UTC Please join the Chime Br...

Status: Pending customer action

[See less](#)

Choose **See details** to view the full correspondence added by AWS Incident Detection and Response.

Correspondence:**Amazon Web Services,** [REDACTED]

The following alarm has engaged AWS Incident Detection and Response to an Incident bridge:

Alarm Identifier - arn:aws:cloudwatch:us-east-1:1234567891012:alarm:ExampleAlarm
Alarm State Change Reason - Threshold Crossed: 3 out of the last 5 datapoints [642.4 (26/09/24 04:51:00), 504.0 (26/09/24 04:52:00), 203.8 (26/09/24 04:55:00)] were greater than the threshold (150.0) (minimum 3 datapoints for OK -> ALARM transition).
Alarm Start Time - 26 September 2024 04:55 AM UTC

Please join the Chime Bridge below so we can start the steps outlined in your Runbook:

Meeting ID: 1234567891012

Chime Bridge: <https://chime.aws/1234567891012>

International dial-in numbers: <https://chime.aws/dialinnumbers/>

We value your feedback. Please share your experience by rating this and other correspondences in the AWS Support Center. You can rate a correspondence by selecting the stars in the top right corner of the correspondence.

Best regards,

Michelle K.

Amazon Web Services

Share to channel

Reply

Resolve case

Create an Incident Response Request in Slack

For instructions on how to create an Incident Response Request through the AWS Support App in Slack, see [Request an Incident Response](#).

Reporting in Incident Detection and Response

AWS Incident Detection and Response provides operational and performance data to help you understand how the service is configured, the history of your incidents, and the performance of the Incident Detection and Response service. This page covers the types of data available, including configuration data, incident data, and performance data.

Configuration data

- All accounts onboarded
- Names of all applications
- The alarms, runbooks, and support profiles associated with each application

Incident data

- The dates, number, and duration of incidents for each application
- The dates, number, and duration of incidents associated with a specific alarm
- Post Incident Report

Performance data

- Service Level Objective (SLO) performance

Reach out to your Technical Account Manager for operational and performance data you may need.

Incident Detection and Response security and resiliency

The [AWS Shared Responsibility Model](#) applies to data protection in Support. As described in this model, AWS is responsible for protecting the global infrastructure that runs all of the AWS Cloud. You are responsible for maintaining control over your content that is hosted on this infrastructure. This content includes the security configuration and management tasks for the AWS services that you use.

For more information about data privacy, see the [Data Privacy FAQ](#).

For information about data protection in Europe, see the [AWS Shared Responsibility Model and GDPR](#) blog post on the AWS Security Blog.

For data protection purposes, we recommend that you protect AWS account credentials and set up individual user accounts with AWS Identity and Access Management (IAM). That way each user is given only the permissions necessary to fulfill their job duties. We also recommend that you secure your data in the following ways:

- Use multi-factor authentication (MFA) with each account.
- Use Secure Sockets Layer/Transport Layer Security (SSL/TLS) certificates to communicate with AWS resources. We recommend TLS 1.2 or later. For information, see [What Is An SSL/TLS Certificate?](#)
- Set up API and user activity logging with AWS CloudTrail. For information, see [AWS CloudTrail](#).
- Use AWS encryption solutions, along with all default security controls within AWS services.
- Use advanced managed security services such as Amazon Macie, which assists in discovering and securing personal data that is stored in Amazon S3. For information about Amazon Macie, see [Amazon Macie](#).
- If you require FIPS 140-2 validated cryptographic modules when accessing AWS through a command line interface or an API, use a FIPS endpoint. For information about the available FIPS endpoints, see [Federal Information Processing Standard \(FIPS\) 140-2](#).

We strongly recommend that you never put confidential or sensitive information, such as your customers' email addresses, into tags or free-form fields such as a **Name** field. This includes when you work with Support or other AWS services using the console, API, AWS CLI, or AWS SDKs. Any data that you enter into tags or free-form fields used for names may be used for billing or

diagnostic logs. If you provide a URL to an external server, we strongly recommend that you do not include credentials information in the URL to validate your request to that server.

AWS Incident Detection and Response access to your accounts

AWS Identity and Access Management (IAM) is a web service that helps you securely control access to AWS resources. You use IAM to control who is authenticated (signed in) and authorized (has permissions) to use resources.

AWS Incident Detection and Response and your alarm data

By default, Incident Detection and Response receives the Amazon resource name (ARN) and state of every CloudWatch alarm in your account and then starts the incident detection and response process when your onboarded alarm changes into the ALARM state. If you would like to customize what information incident detection and response receives about alarms from your account, contact your Technical Account Manager.

Document history

The following table describes the important changes to the documentation since the last release of the IDR guide.

Change	Description	Date
Updated Request an Incident Response procedure	<p>Updated the Request an Incident Response procedure to match the current AWS Support Center Console UI, added bridge URL guidance, and removed outdated screenshots.</p> <p>For more information, see Request an Incident Response using the AWS Support Center Console.</p>	May 12, 2026
Updated onboarding to CLI-first approach	<p>Updated the Get started chapter to promote the AWS Incident Detection and Response Customer Command Line Interface as the primary onboarding method and deprecated the Workload Onboarding Questionnaire and Alarm Ingestion Questionnaire as the default onboarding path. Questionnaires remain available as an exception-only option for customers who can't use the IDR CLI.</p> <p>For more information, see Onboard workloads to Incident Detection and Response and Alarm Ingestion.</p>	May 12, 2026
Added Japanese questionnaire links	<p>Added Japanese-language download links for the Workload onboarding questionnaire and Alarm ingestion questionnaire.</p> <p>For more information, see Workload onboarding and alarm ingestion questionn</p>	April 20, 2026

Change	Description	Date
	aires in Incident Detection and Response (exception path) .	
Updated architecture references	Removed references to architecture diagrams and replaced with architecture details. For more information, see Architecture of Incident Detection and Response and About workloads in Incident Detection and Response .	March 31, 2026
Updated Test onboarded workloads in Incident Detection and Response	Added information about disabling CloudWatch alarm actions before changing alarm state during testing. For more information, see Test onboarded workloads in Incident Detection and Response .	March 2, 2026
Updated Incident management with Incident Detection and Response	Added information about repeating alarm behavior and incident manager engagement. For more information, see Incident management with Incident Detection and Response .	March 2, 2026
Updated steps in the Use a metric math function to suppress a CloudWatch alarm section	Updated steps in the Use a metric math function to suppress a CloudWatch alarm section. For more information, see Suppress alarms at the alarm source .	February 3, 2026
Added Korean as a supported language	Added Korean as a supported language. For more information, see Region availability for Incident Detection and Response .	January 22 2026

Change	Description	Date
Added Mandarin as a supported language	<p>Added Mandarin as a supported language.</p> <p>For more information, see Region availability for Incident Detection and Response.</p>	January 13, 2026
Added a new section: AWS Incident Detection and Response Customer Command Line Interface	<p>Added the IDR CLI section and updated the Get started chapter to include information about the AWS Incident Detection and Response Customer Command Line Interface.</p> <p>For more information, see CLI for AWS Incident Detection and Response.</p>	December 8, 2025
Updated multiple sections: <i>Workload onboarding and alarm ingestion questionnaires in Incident Detection and Response</i> and <i>Get started in Incident Detection and Response</i>	<p>The AWS service events handling process is no longer part of AWS Incident Detection and Response. Sections of this user guide were updated to remove references to this process. You will continue to receive service event notifications through the AWS Service Health Dashboard. AWS Incident Detection and Response customers can use an Incident Response request to receive help during service events as needed. For more information, see Request an Incident Response.</p>	October 14, 2025
Deleted section: <i>Incident management for service events</i>	<p>The AWS service events handling process is no longer part of AWS Incident Detection and Response. This section of the user guide was removed to reflect this change. You will continue to receive service event notifications through the AWS Service Health Dashboard. AWS Incident Detection and Response customers can use an Incident Response request to receive help during service events as needed. For more information, see Request an Incident Response.</p>	October 14, 2025

Change	Description	Date
Updated section: <i>Region availability for Incident Detection and Response</i>	AWS Incident Detection and Response is now available in AWS GovCloud (US-East) and AWS GovCloud (US-West). For more information, see Region availability for Incident Detection and Response	October 05, 2025
Updated section: <i>Workload onboarding and alarm ingestion questionnaires in Incident Detection and Response</i>	Updated example email address for Alarm matrix table.	August 26, 2025
Updated section: <i>Subscribe a workload to AWS Incident Detection and Response</i>	<p>Removed reference to the Subscription start date field in the Description section of the Create case window.</p> <p>Updated section: <i>Subscribe a workload to AWS Incident Detection and Response</i></p>	August 4, 2025
New function: Suppress alarms from engaging Incident Detection and Response	<p>Added new sections to Managed workloads that provide information on how to suppress alarms temporarily or on a schedule</p> <p>New section: Suppress alarms from engaging Incident Detection and Response</p>	April 9, 2025
Updated instructions for Request an Incident Response using the AWS Support Center Console	<p>Added details on what information to enter in the Problem description field.</p> <p>Updated section: Request an Incident Response</p>	February 6, 2025

Change	Description	Date
Additional AWS Regions added	<p>Additional AWS Regions have been added to the Incident Detection and Response availability section.</p> <p>Updated section: Region availability for Incident Detection and Response</p>	November 1, 2024
Updates to Manage Incident Detection and Response support cases with the AWS Support App in Slack page	<p>Moved page under Incident Management, revised text, and replaced screenshots.</p> <p>Updated section: Manage Incident Detection and Response support cases with the AWS Support App in Slack</p>	October 10, 2024
<p>Added a new page AWS Support App in Slack</p> <p>Updated Incident management with AWS Incident Detection and Response</p>	<p>Added a new page for AWS Support App in Slack</p> <p>Updated Incident management with AWS Incident Detection and Response to add a new section, "Request an Incident Response using the AWS Support App in Slack".</p>	September 10, 2024
Updated Account subscription	<p>Updated the Account subscription section to include details on where to open a support case when you request to subscribe an account.</p> <p>Updated section: Subscribe a workload to AWS Incident Detection and Response</p>	June 12, 2024
Added a new section: Offboard a workload	<p>Added the Offload a workload section in Getting started to include information about offboarding workloads</p> <p>For more information, see Offboard a workload from Incident Detection and Response.</p>	March 28, 2024

Change	Description	Date
Updated Account subscription	<p>Updated the Account subscription section to include information about offboarding workloads</p> <p>For more information, see Subscribe a workload to AWS Incident Detection and Response</p>	March 28, 2024
Updated Testing	<p>Updated the Testing section to include information on gameday testing as the last step in the onboarding process.</p> <p>Updated section: Test onboarded workloads in Incident Detection and Response</p>	February 29, 2024
Updated What is AWS Incident Detection and Response	<p>Updated the What is AWS Incident Detection and Response section.</p> <p>Updated section: What is AWS Incident Detection and Response?</p>	February 19, 2024
Updated Questionnaire section	<p>Updated the Workload onboarding questionnaire and added Alarm ingestion questionnaire. Renamed the section from Onboarding questionnaire to Workload onboarding and Alarm ingestion questionnaires.</p>	February 2, 2024

Change	Description	Date
Updated AWS Service Event and onboarding information	<p>Updated several sections with new information for onboarding.</p> <p>Updated sections:</p> <ul style="list-style-type: none"> • Onboard workloads to Incident Detection and Response • Subscribe a workload to AWS Incident Detection and Response <p>New sections</p> <ul style="list-style-type: none"> • Provision access to AWS Support Center Console for application teams 	January 31, 2024
Added a Related information section	<p>Added a Related information section in Access provisioning.</p> <p>Updated section: Provision access for alarm ingestion to Incident Detection and Response</p>	January 17, 2024
Updated example steps	<p>Updated the procedure for steps 2,3, and 4 in Example: Integrating notifications from Datadog and Splunk.</p> <p>Updated section: Example: Integrating notifications from Datadog and Splunk</p>	December 21, 2023
Updated introduction graphic and text	<p>Updated graphic in Ingest alarms from APMs that have direct integration with Amazon EventBridge.</p> <p>Updated section: Develop runbooks and response plans for responding to an incident in Incident Detection and Response</p>	December 21, 2023

Change	Description	Date
Updated runbook template	<p>Updated the runbook template in Developing runbooks for AWS Incident Detection and Response.</p> <p>Updated section: Develop runbooks and response plans for responding to an incident in Incident Detection and Response</p>	December 4, 2023
Updated Alarm Configurations	<p>Updated Alarm Configurations with detailed information on CloudWatch alarm configuration.</p> <p>New section: Create CloudWatch alarms that fit your business needs in Incident Detection and Response</p> <p>New section: Build CloudWatch alarms in Incident Detection and Response with CloudFormation templates</p> <p>New section: Example use cases for CloudWatch alarms in Incident Detection and Response</p>	September 28, 2023
Updated Getting Started	<p>Updated Getting Started with information on Workload change requests.</p> <p>New section: Request changes to an onboarded workload in Incident Detection and Response</p> <p>Updated section: Subscribe a workload to AWS Incident Detection and Response</p>	September 05, 2023
New section in Getting Started	Added Ingesting alerts into AWS Incident Detection and Response.	June 30, 2023

Change	Description	Date
Original document	AWS Incident Detection and Response first published	March 15, 2023