AWS Incident Detection and Response の概念と手順

AWS Incident Detection and Response ユーザーガイド

Version May 15, 2025

Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

AWS Incident Detection and Response ユーザーガイド: AWS Incident Detection and Response の概念と手順

Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon の商標およびトレードドレスはAmazon 以外の製品およびサービスに使用することはできません。また、お客様に誤解を与える可能性がある形式で、または Amazon の信用を損なう形式で使用することもできません。Amazon が所有していないその他のすべての商標は Amazon との提携、関連、支援関係の有無にかかわらず、それら該当する所有者の資産です。

Table of Contents

AWS Incident Detection and Response とは 1
利用規約 2
アーキテクチャ 2
役割と責任 3
利用可能なリージョン 6
はじめに 8
ワークロード 8
アラーム 8
オンボーディング 9
ワークロードのオンボーディング 9
アラームの取り込み 10
オンボーディングに関するアンケート 10
ワークロードオンボーディングのアンケート - 一般的な質問 11
ワークロードオンボーディングのアンケート - アーキテクチャに関する質問 11
ワークロードオンボーディングのアンケート - AWS のサービスイベントに関する質問 14
アラームの取り込みのアンケート 14
アラームのマトリックス 16
ワークロードの検出 20
ワークロードをサブスクライブする 21
アラームを定義および設定する 23
CloudWatch アラームの作成 26
CloudFormation テンプレートを使用して CloudWatch アラームを作成する 29
CloudWatch アラームのユースケースの例 32
アラームを取り込む 34
アクセスのプロビジョニング 35
CloudWatch との統合 35
EventBridge と統合されている APM からアラームを取り込む 36
例: Datadog と Splunk からの通知の統合 37
EventBridge と統合していない APM からアラームを取り込む 47
ワークロードを管理する 48
ランブックと対応計画を作成する 48
オンボードされたワークロードをテストする 55
CloudWatch アラーム 56
サードパーティーの APM アラーム 56

重要なアウトプット 56
ワークロードへの変更をリクエストする 57
アラームを抑制 58
アラームソースでアラームを抑制 58
ワークロード変更リクエストを送信してアラームを抑制 63
チュートリアル: Metric Math 関数を使用してアラームを抑制 64
チュートリアル: Metric Math 関数を削除してアラーム抑制を解除 66
ワークロードのオフボード 67
モニタリングとオブザーバビリティ 69
オブザーバビリティの実装 70
インシデント管理 71
アプリケーションチームのアクセス権をプロビジョニングする 74
サービスイベントのインシデント管理 74
インシデント対応をリクエストする 75
AWS Support Center Console を介したリクエスト 75
AWS サポート API を介したリクエスト 76
AWS Support App in Slack を介したリクエスト 76
AWS Support App in Slack で Incident Detection and Response のサポートケースを管理する 78
Slack でのアラームによって開始されたインシデントの通知 79
Slack でインシデント対応リクエストを作成する 79
レポート作成 80
セキュリティと回復性 81
アカウントへのアクセス 82
アラームデータ 82
ドキュメント履歴 83

AWS Incident Detection and Response とは

AWS Incident Detection and Response は、対象となる AWS エンタープライズサポートのお客様に、障害の可能性を減らし、重要なワークロードの中断からの復旧を加速するための、プロアクティブなインシデント対応を提供します。Incident Detection and Response により、AWS とのコラボレーションが促進され、オンボーディングされた各ワークロードに合わせてカスタマイズされたランブックとレスポンスプランが策定できます。

インシデント検出と対応には、次の主要な機能があります。

- オブザーバビリティの向上: AWS の専門家は、ワークロードのアプリケーションレイヤーとインフラストラクチャレイヤー間のメトリクスとアラームの定義と関連付けを支援し、中断を早期に検出できるようにします。
- 5 分以内の応答時間: インシデント管理エンジニアリング (IME) は、オンボードされたお客様のワークロードを 24 時間 365 日モニタリングして、重大なインシデントを検出します。IME は、アラームがトリガーされてから 5 分以内に応答するか、お客様が Incident Detection and Response に設定したビジネスクリティカルなサポートケースに対応します。
- より迅速な解決: IME は、ワークロード用に策定された事前定義済みのカスタムランブックを使用して、5 分以内に応答し、お客様に代わってサポートケースを作成し、ワークロードのインシデントを管理します。IME は、インシデントに対する一元化された所有権を提供し、インシデントが解決されるまで適切な AWS の専門家と連携し続けます。
- AWS イベントのインシデント管理: 当社はお客様の重要なワークロード (アカウント、サービス、インスタンスなど) のコンテキストを把握しているため、AWS のサービスイベント中にワークロードへの潜在的な影響を検出して事前に通知できます。リクエストがあれば、IME は AWS のサービスイベント中にお客様と連携し、イベントの最新情報を提供します。Incident Detection and Response はサービスイベント中の復旧に優先順位を付けることはできませんが、障害軽減プランの実装に役立つサポートガイダンスを提供します。
- 障害の可能性の低減: 解決後、IME はインシデント後レビュー (リクエストに応じて) を提供します。また、AWS の専門家がお客様と協力して、インシデントレスポンスプランとランブックを改善するために学んだ教訓を適用します。また、ワークロードの回復性の継続的な追跡に AWS Resilience Hub を活用することもできます。

トピック

- Incident Detection and Response の利用規約
- Incident Detection and Response のアーキテクチャ

- Incident Detection and Response における役割と責任
- Incident Detection and Response が利用可能なリージョン

Incident Detection and Response の利用規約

次のリストは、AWS Incident Detection and Response を使用するための主要な要件と制限の概要を示しています。この情報は、サポートプランの要件、オンボーディングプロセス、最小サブスクリプション期間などの側面をカバーするため、サービスを使用する前に理解しておくことが重要です。

- AWS Incident Detection and Response は直販およびパートナーが再販したエンタープライズサポートアカウントで利用できます。
- AWS Incident Detection and Response は Partner-Led Support のアカウントでは利用できません。
- Incident Detection and Response サービスの期間中は、常に AWS エンタープライズサポートを維持する必要があります。詳細については、「AWS エンタープライズサポート」を参照してください。エンタープライズサポートを終了すると、AWS Incident Detection and Response サービスから同時に削除されます。
- AWS Incident Detection and Response のすべてのワークロードは、ワークロードのオンボーディングプロセスを経る必要があります。
- アカウントで AWS Incident Detection and Response をサブスクライブするための最小期間は 90 日です。すべてのキャンセルリクエストは、キャンセル予定日の 30 日前に提出する必要があります。
- AWS は、「AWS プライバシー通知」の説明に従ってお客様の情報を取り扱います。

> **Note**
>
> Incident Detection and Response の請求に関する質問については、AWS の請求に関連したヘルプについての記事を参照してください。

Incident Detection and Response のアーキテクチャ

次の図に示すように、AWS Incident Detection and Response は既存の環境と統合されます。このアーキテクチャには、以下のサービスが含まれます。

- Amazon EventBridge: Amazon EventBridge は、ワークロードと AWS Incident Detection and Response 間の唯一の統合ポイントとして機能します。アラームは、AWS によって管理される事前定義されたルールを使用して、Amazon EventBridge を介して Amazon CloudWatch などのモニタリングツールから取り込まれます。Incident Detection and Response が EventBridge ルールを構築および管理できるようにするには、サービスにリンクされたロールをインストールします。これらのサービスの詳細については、「Amazon EventBridge とは」、「Amazon EventBridge ルールとは」、「Amazon CloudWatch とは」、「AWS Health のサービスにリンクされたロールの使用」を参照してください。
- AWS Health: AWS Health は、リソースのパフォーマンスと、AWS のサービスのアカウントの可用性を継続的に可視化します。Incident Detection and Response では、AWS Health を使用して、ワークロードが使用する AWS のサービス のイベントを追跡し、ワークロードからアラートを受け取ったときに通知します。AWS Health の詳細については、「AWS Health とは」を参照してください。
- AWS Systems Manager: Systems Manager は、AWS リソース全体で自動化とタスク管理のための統合ユーザーインターフェイスを提供します。AWS Incident Detection and Response は、ワークロードアーキテクチャ図、アラームの詳細、対応するインシデント管理ランブックなどのワークロードに関する情報を AWS Systems Manager ドキュメントでホストします (詳細については、「AWS Systems Manager ドキュメント」を参照してください)。AWS Systems Manager の詳細については、「AWS Systems Manager とは」を参照してください。
- 特定のランブック: インシデント管理ランブックは、インシデント管理中に AWS Incident Detection and Response が実行するアクションを定義します。特定のランブックは、AWS Incident Detection and Response に、連絡先、連絡方法、共有する情報を伝えます。

Incident Detection and Response における役割と責任

AWS Incident Detection and Response RACI (Responsible = 実行責任者、Accountable = 説明責任者、Consulted = 相談先、Informed = 報告先) の表は、インシデントの検出と対応に関連するさまざまなアクティビティの役割と責任の概要を示します。この表は、データ収集、運用準備状況レビュー、アカウント設定、インシデント管理、インシデント後レビューなどのタスクに対するお客様と AWS Incident Detection and Response チームの関与を定義するのに役立ちます。

アクティビティ	お客様	Incident Detection and Response
データ収集		
カスタマーとワークロードの導入	相談先	実行責任者
アーキテクチャ	実行責任者	説明責任者
オペレーション	実行責任者	説明責任者
設定する CloudWatch アラームを決定する	実行責任者	説明責任者
インシデントレスポンスプランを定義する	実行責任者	説明責任者
オンボーディングのアンケートに入力する	実行責任者	説明責任者
運用準備状況のレビュー		
ワークロードに関する Well Architected レビュー (WAR) を実施する	相談先	実行責任者
インシデント対応を検証する	相談先	実行責任者
アラームマトリックスを検証する	相談先	実行責任者
ワークロードで使用されている主要な AWS のサービスを特定する	説明責任者	実行責任者

アクティビティ	お客様	Incident Detection and Response
アカウント設定		
カスタマーアカウントに IAM ロールを作成する	実行責任者	報告先
作成したロールを使用してマネージド EventBridge ルールをインストールする	報告先	実行責任者
CloudWatch アラームをテストする	実行責任者	説明責任者
カスタマーアラームがインシデントの検出と対応に関与していることを確認する	報告先	実行責任者
アラームを更新する	実行責任者	相談先
ランブックを更新する	相談先	実行責任者
インシデント管理		
Incident Detection and Response によって検出されたインシデントをプロアクティブに通知する	報告先	実行責任者
インシデント対応を提供する	報告先	実行責任者
インシデントの解決/インフラストラクチャの復元を提供する	実行責任者	相談先
インシデント後レビュー		
インシデント後レビューをリクエストする	実行責任者	報告先

アクティビティ	お客様	Incident Detection and Response
インシデント後レビューを提供する	報告先	実行責任者

Incident Detection and Response が利用可能なリージョン

AWS Incident Detection and Response は現在、次のいずれかの AWS リージョンでホストされているエンタープライズサポートアカウントで英語と日本語でご利用いただけます。

名前	AWS リージョン
us-east-1	米国東部 (バージニア)
us-east-2	米国東部 (オハイオ)
us-west-1	米国西部 (北カリフォルニア)
us-west-2	米国西部 (オレゴン)
ca-central-1	カナダ (中部)
ca-west-1	カナダ西部 (カルガリー)
sa-east-1	南米 (サンパウロ)
eu-central-1	欧州 (フランクフルト)
eu-west-1	欧州 (アイルランド)
eu-west-2	欧州 (ロンドン)
eu-west-3	欧州 (パリ)
eu-north-1	欧州 (ストックホルム)

名前	AWS リージョン
eu-central-2	欧州 (チューリッヒ)
eu-south-1	欧州 (ミラノ)
eu-south-2	欧州 (スペイン)
ap-south-1	アジアパシフィック (ムンバイ)
ap-northeast-1	アジアパシフィック (東京)
ap-northeast-2	アジアパシフィック (ソウル)
ap-southeast-1	アジアパシフィック (シンガポール)
ap-southeast-2	アジアパシフィック (シドニー)
ap-east-1	アジアパシフィック (香港)
ap-northeast-3	アジアパシフィック (大阪)
ap-south-2	アジアパシフィック (ハイデラバード)
ap-southeast-3	アジアパシフィック (ジャカルタ)
ap-southeast-4	アジアパシフィック (メルボルン)
ap-southeast-5	アジアパシフィック (マレーシア)
af-south-1	アフリカ (ケープタウン)
il-central-1	イスラエル (テルアビブ)
me-central-1	中東 (UAE)
me-south-1	中東 (バーレーン)

Incident Detection and Response の開始方法

ワークロードとアラームは AWS Incident Detection and Response の中核です。AWS は、お客様と密接に連携して、ビジネスにとって重要な特定のワークロードを定義およびモニタリングします。AWS は、重大なパフォーマンスの問題やお客様への影響をチームに迅速に通知するアラームの設定をお手伝いします。Incident Detection and Response 内のプロアクティブなモニタリングと迅速なインシデント対応には、適切に設定されたアラームが不可欠です。

ワークロード

AWS Incident Detection and Response を使用すると、特定のワークロードを選択して、モニタリングおよび重要なインシデント管理を行うことができます。ワークロードは、リソースとコードの集合であり、連携してビジネス価値を提供します。ワークロードとは、お客様の銀行支払いポータルまたは顧客関係管理 (CRM) システムを構成するすべてのリソースとコードです。ワークロードは、1 つの AWS アカウントまたは複数の AWS アカウントでホストできます。

例えば、モノリシックアプリケーションが 1 つのアカウントでホストされている場合があります (次の図の従業員パフォーマンスアプリケーションなど)。または、アプリケーション (図のストアフロントウェブアプリケーションなど) が、マイクロサービスに分割されて、複数の異なるアカウントにまたがっている場合があります。図に示すように、ワークロードが、データベースなどのリソースを他のアプリケーションやワークロードと共有している場合があります。

ワークロードオンボーディングを開始するには、「ワークロードオンボーディング」と「ワークロードオンボーディングのアンケート」を参照してください。

アラーム

アラームは、アプリケーションおよび基盤となる AWS インフラストラクチャのパフォーマンスを可視化するという点で、Incident Detection and Response の重要な部分です。AWS はお客様と連携して、モニタリング対象のワークロードに重大な影響がある場合にのみ適切なメトリクスとアラームをトリガーするしきい値を定義します。目標は、アラームにより、指定したリゾルバーを関与させてインシデント管理チームと連携させ、問題を迅速に軽減できるようにすることです。アラームは、パフォーマンスやカスタマーエクスペリエンスの大幅な低下にすぐ対処する必要がある場合にのみ、アラーム状態に入るように設定する必要があります。アラームの主なタイプには、ビジネスへの影響を示すアラーム、Amazon CloudWatch canary、依存関係をモニタリングする集計アラームなどがあります。

アラームの取り込みを開始するには、「アラームの取り込み」と「アラームの取り込みのアンケート」を参照してください。

> **Note**
>
> ランブック、ワークロード情報、または AWS Incident Detection and Response でモニタリングするアラームを変更するには、「Incident Detection and Response でオンボードしたワークロードへの変更をリクエストする」を参照してください。

Incident Detection and Response へのオンボーディング

AWS は、ワークロードとアラームを AWS Incident Detection and Response にオンボードするためにお客様と協力して作業します。お客様は Incident Detection and Response のワークロードのオンボーディングとアラームの取り込みに関するアンケート で重要な情報を AWS に提供します。AppRegistry にもワークロードを登録するのがベストプラクティスです。詳細については、AppRegistry のユーザーガイドを参照してください。

次の図は、Incident Detection and Response におけるワークロードのオンボーディングとアラームの取り込みのフローを示しています。

ワークロードのオンボーディング

ワークロードをオンボーディングするにあたり、AWS はお客様と協力して、ワークロードと、インシデントや AWS サービスイベントが発生した際にユーザーをサポートする方法を理解します。お客様は、影響の軽減に役立つ、ワークロードに関する重要な情報を提供します。

重要なアウトプット:

- 一般的なワークロード情報
- アーキテクチャの詳細 (図を含む)
- ランブック情報
- お客様が開始したインシデント
- AWS サービスイベント

アラームの取り込み

AWS は、アラームをオンボードするためにお客様と協力して作業します。AWS Incident Detection and Response では、Amazon EventBridge を介して Amazon CloudWatch およびサードパーティのアプリケーションパフォーマンスモニタリング (APM) ツールからアラームを取り込むことができます。アラームをオンボーディングすることで、プロアクティブなインシデント検出とエンゲージメントの自動化が可能になります。詳細については、「Ingest alarms from APMs that have direct integration with Amazon EventBridge」を参照してください。

重要なアウトプット:

- アラームのマトリックス

次の表は、ワークロードを AWS Incident Detection and Response にオンボードするために必要なステップをリストにしたものです。この表は、各タスクの所要時間の例を示しています。各タスクの実際の日付は、チームの可用性とスケジュールに基づいて定義されます。

Incident Detection and Response のワークロードのオンボーディングとアラームの取り込みに関するアンケート

このページでは、ワークロードを AWS Incident Detection and Response にオンボーディングする場合と、サービスに取り込むアラームを設定する場合に、入力する必要があるアンケートについて説明します。ワークロードのオンボーディングに関するアンケートでは、ワークロード、アーキテクチャの詳細、インシデント対応の問い合わせに関する一般的な情報をカバーします。アラームの取り込みに関するアンケートでは、Incident Detection and Response でワークロードのインシデントの作成をトリガーする重要なアラームと、誰に連絡すべきか、どのようなアクションを実行すべきかに関するランブック情報を指定します。アンケートへの適切な入力は、AWS ワークロードのモニタリングおよびインシデント対応プロセスを設定する重要なステップです。

ワークロードオンボーディングのアンケートをダウンロードします。

アラームの取り込みのアンケートをダウンロードします。

ワークロードオンボーディングのアンケート - 一般的な質問

一般的な質問

質問	レスポンスの例
エンタープライズ名	Amazon Inc.
このワークロードの名前 (略語を含む)	Amazon Retail Operations (ARO)
プライマリエンドユーザーとこのワークロードの機能。	このワークロードは、エンドユーザーがさまざまなアイテムを購入できるようにする e コマースアプリケーションです。このワークロードは、弊社のビジネスの主要な収益源です。
このワークロードに適用されるコンプライアンスおよび/または規制要件、インシデント発生後に AWS から要求されるアクション。	ワークロードは、安全と機密性を保持する必要がある患者の健康記録を扱います。

ワークロードオンボーディングのアンケート - アーキテクチャに関する質問

アーキテクチャに関する質問

質問	レスポンスの例
このワークロードの一部であるリソースを定義するために使用される AWS リソースタグのリスト。AWS は、これらのタグを使用してこのワークロードのリソースを識別し、インシデント中のサポートを迅速化します。 Note タグでは、大文字と小文字が区別されます。複数のタグを指定する場合、こ	appName: Optimax environment: Production

質問	レスポンスの例
のワークロードで使用されるすべてのリソースに同じタグが必要です。	
このワークロードで利用される AWS のサービスとそれらが存在する AWS アカウントとリージョンのリスト。 Note 各サービスごとに新しい行を作成します。	Route 53: インターネットトラフィックを ALB にルーティングします。 アカウント: 123456789101 リージョン: US-EAST-1、US-WEST-2
このワークロードで利用される AWS のサービスとそれらが存在する AWS アカウントとリージョンのリスト。 Note 各サービスごとに新しい行を作成します。	ALB: ECS コンテナのターゲットグループに受信トラフィックをルーティングします。 アカウント: 123456789101 リージョン: 該当なし
このワークロードで利用される AWS のサービスとそれらが存在する AWS アカウントとリージョンのリスト。 Note 各サービスごとに新しい行を作成します。	ECS: 主要なビジネスロジックフリートのコンピューティングインフラストラクチャ。受信ユーザーリクエストを処理し、永続化レイヤーにクエリを実行する役割があります。 アカウント: 123456789101 リージョン: US-EAST-1

質問	レスポンスの例
このワークロードで利用される AWS のサービスとそれらが存在する AWS アカウントとリージョンのリスト。 Note 各サービスごとに新しい行を作成します。	RDS: Amazon Aurora クラスターは、ECS ビジネスロジックレイヤーによってアクセスされたユーザーデータを保存します。 アカウント: 123456789101 リージョン: US-EAST-1
このワークロードで利用される AWS のサービスとそれらが存在する AWS アカウントとリージョンのリスト。 Note 各サービスごとに新しい行を作成します。	S3: ウェブサイトの静的アセットを保存します。 アカウント: 123456789101 リージョン: 該当なし
停止が発生した場合にこのワークロードに影響を与える可能性のある、オンボードされていないアップストリーム/ダウンストリームコンポーネントの詳細。	認証マイクロサービス: 認証されていないため、ユーザーが健康記録を読み込めなくなります。
このワークロードにはオンプレミスコンポーネントまたは非 AWS コンポーネントがありますか。ある場合、それらはどのようなもので、どのような機能が実行されますか。	AWS に出入りするすべてのインターネットベースのトラフィックは、オンプレミスのプロキシサービスを介してルーティングされます。
アベイラビリティーゾーンおよびリージョンレベルで、手動または自動フェイルオーバー/ディザスタリカバリプランの詳細を指定します。	ウォームスタンバイ。成功率が持続的に低下している間の US-WEST-2 への自動フェイルオーバー。

ワークロードオンボーディングのアンケート - AWS のサービスイベントに関する質問

AWS のサービスイベントに関する質問

質問	レスポンスの例
社内の重大インシデント/IT 危機管理チームの連絡先の詳細 (名前/E メール/電話) を指定します。	重大インシデント管理チーム mim@example.com +61 2 3456 7890
会社が確立した静的インシデント/危機管理ブリッジの詳細を指定します。非静的ブリッジを使用する場合は、お好みのアプリケーションを指定し、AWS はインシデント中にこれらの詳細をリクエストします。 Note 指定されていない場合、インシデント中に AWS が連絡を取り、参加できる Chime ブリッジを提供します。	Amazon Chime https://chime.aws/1234567890

アラームの取り込みのアンケート

ランブックに関する質問

質問	レスポンスの例
AWS は、サポート ケースを通じてワークロードの問い合わせをエンゲージします。このワークロードでアラームがトリガーされた場合、主な連絡先は誰ですか。	アプリケーションチーム app@example.com +61 2 3456 7890

質問	レスポンスの例
優先する会議アプリケーションを指定すると、AWS はインシデント中にこれらの詳細をリクエストします。 Note 優先する会議アプリケーションが指定されていない場合、インシデント中に AWS が連絡を取り、参加できる Chime ブリッジを提供します。	
インシデント中に主な連絡先が利用できない場合は、希望する通信順序でエスカレーション連絡先とタイムラインを指定してください。	1. 10 分経過しても、主要連絡先から応答がない場合は、次の連絡先と連絡を取ります。 John Smith - アプリケーションスーパーバイザー john.smith@example.com +61 2 3456 7890 2. 10 分経過しても、John Smith から応答がない場合は、次の連絡先と連絡を取ります。 Jane Smith - オペレーションマネージャー jane.smith@example.com +61 2 3456 7890
AWS は、インシデント全体で一定の間隔で、サポートケースを通じて更新を伝達します。これらの更新を受け取る必要がある追加の連絡先はありますか。	john.smith@example.com、jane.smith@example.com

アラームのマトリックス

ワークロードに代わってインシデントを作成するために AWS Incident Detection and Response をエンゲージする一連のアラームを特定するために、次の情報を提供します。AWS Incident Detection and Response のエンジニアがアラームを確認すると、追加のオンボーディング手順が提供されます。

AWS Incident Detection and Response の重大なアラーム基準:

- AWS Incident Detection and Response のアラームは、オペレーターの即時対応を必要とするモニタリング対象のワークロードに、重大なビジネスへの影響 (収益の損失/カスタマーエクスペリエンスの低下) がある場合にのみ、「Alarm」状態に入る必要があります。
- AWS Incident Detection and Response のアラームは、ワークロードのリゾルバーを同時に、またはエンゲージメントの前に、エンゲージさせる必要もあります。AWSIncident Managers は、緩和プロセスでリゾルバーと連携しますが、エスカレーションする第一線の応答者としては機能しません。
- AWS Incident Detection and Response のアラームのしきい値は、アラームが発せられたときに調査が行われるように、適切なしきい値と期間に設定する必要があります。アラームが「Alarm」状態と「OK」状態の間で移動している場合、オペレータの応答と注意を必要とする十分な影響が発生しています。

基準違反の AWS Incident Detection and Response ポリシー:

これらの基準は、イベントが発生したときにケースバイケースでのみ評価できます。インシデント管理チームは、テクニカルアカウントマネージャー (TAM) と連携して、顧客のアラームがこの基準に準拠しておらず、一定の間隔で不必要にインシデント管理チームにエンゲージしていると疑われる場合、アラームを調整し、まれにモニタリングを無効にします。

> **Important**
>
> 連絡先アドレスを提供する際にグループ配布用の E メールアドレスを指定すると、ランブックを更新せずに受信者の追加と削除を制御できます。
> 最初のエンゲージメント E メールを送信した後に AWS Incident Detection and Response チームから電話をもらいたい場合は、サイト信頼性エンジニアリング (SRE) チームの連絡先電話番号を指定します。

アラームのマトリックスの表

メトリクス名/ARN/しきい値	説明	メモ	リクエストされたアクション
ワークロードボリューム/ *CW Alarm ARN*/ 5 分以内に 5 つのデータポイントの CallCount が 100,000 未満の場合、欠落したデータを欠落として処理します。	このメトリクスは、Application Load Balancer レベルで測定された、ワークロードへの受信リクエストの数を表します。 このアラームは重要です。受信リクエストの大幅な減少は、アップストリームネットワーク接続に問題があるか、DNS 実装に問題があり、その結果ユーザーがワークロードにアクセスできない可能性があるためです。	アラームは先週 10 回「Alarm」状態になりました。このアラームは誤検出のリスクがあります。しきい値の見直しが計画されています。 問題は? いいえまたははい (いいえの場合は空白のまま): このアラームは、特定のバッチジョブの実行中に頻繁に入れ替わります。 リゾルバー: サイト信頼性エンジニア	*SRE@xyz.com* に E メールを送信して、サイト信頼性エンジニアリングチームを関与させます。 ELB および Route 53 サービス用の AWS プレミアムサポートケースを作成します。 即時のアクションが必要な場合: EC2 の空きメモリ/ディスク容量を確認し、*XYZ* チームに E メールで通知してインスタンスを再起動するか、ログフラッシュを実行します。(即時のアクションが必要ない場合は空白のままにします)
ワークロードリクエストのレイテンシー/ *CW Alarm ARN*/ 5 分以内に 5 つのデータポイントの p90 レイテンシーが 100 ミリ秒超の場合、欠落	このメトリクスは、ワークロードによって実行される HTTP リクエストの p90 レイテンシーを表します。	アラームは先週 0 回「Alarm」状態になりました。 問題は? いいえまたははい (いいえの場合は空白のまま): このアラームは、特定の	*SRE@xyz.com* に E メールを送信して、サイト信頼性エンジニアリングチームを関与させます。 ECW および RDS サービスの AWS プレ

メトリクス名/ARN/しきい値	説明	メモ	リクエストされたアクション
したデータを欠落として処理します	このアラームは、レイテンシー (ウェブサイトのカスタマーエクスペリエンスの重要な指標) を表します。	バッチジョブの実行中に頻繁に入れ替わります。 リゾルバー: サイト信頼性エンジニア	ミアムサポートケースを作成します。 即時のアクションが必要な場合: EC2 の空きメモリ/ディスク容量を確認し、*XYZ* チームに E メールで通知してインスタンスを再起動するか、ログフラッシュを実行します。(即時のアクションが必要ない場合は空白のままにします)

メトリクス名/ARN/しきい値	説明	メモ	リクエストされたアクション
ワークロードリクエストの可用性/ *CW Alarm ARN*/ 5 分以内に 5 つのデータポイントの可用性が 95% 未満の場合、欠落したデータを欠落として処理します。	このメトリクスは、ワークロードが満たすべき HTTP リクエストの可用性を表します。(HTTP 200 の数/リクエストの数) 期間あたり。 このアラームは、ワークロードの可用性を表します。	アラームは先週 0 回「Alarm」状態になりました。 問題は? いいえまたははい (いいえの場合は空白のまま): このアラームは、特定のバッチジョブの実行中に頻繁に入れ替わります。 リゾルバー: サイト信頼性エンジニア	*SRE@xyz.com* に E メールを送信して、サイト信頼性エンジニアリングチームを関与させます。 ELB および Route 53 サービス用の AWS プレミアムサポートケースを作成します。 即時のアクションが必要な場合: EC2 の空きメモリ/ディスク容量を確認し、*XYZ* チームに E メールで通知してインスタンスを再起動するか、ログフラッシュを実行します。(即時のアクションが必要ない場合は空白のままにします)
New Relic アラームの例			

メトリクス名/ARN/しきい値	説明	メモ	リクエストされたアクション
エンドツーエンドの統合テスト/ *CW Alarm ARN*/ 3 分間の期間で 1 分間のメトリクスの失敗率が 3% の場合、欠落しているデータを欠落として処理します。 ワークロード識別子: エンドツーエンドのテストワークフロー、AWS リージョン: US-EAST-1、AWS アカウント ID: 012345678910	このメトリクスは、リクエストがワークロードの各レイヤーを通過できるかどうかをテストします。このテストが失敗した場合、ビジネストランザクションの処理に重大な障害があることを示しています。 このアラームは、ワークロードのビジネストランザクションを処理する能力を表します。	アラームは先週 0 回「Alarm」状態になりました。 問題は? いいえまたははい (いいえの場合は空白のまま): このアラームは、特定のバッチジョブの実行中に頻繁に入れ替わります。 リゾルバー: サイト信頼性エンジニア	*SRE@xyz.com* に E メールを送信して、サイト信頼性エンジニアリングチームを関与させます。 ECS および DynamoDB サービスの AWS プレミアムサポートケースを作成します。 即時のアクションが必要な場合: EC2 の空きメモリ/ディスク容量を確認し、*XYZ* チームに E メールで通知してインスタンスを再起動するか、ログフラッシュを実行します。(即時のアクションが必要ない場合は空白のままにします)

Incident Detection and Response でのワークロードの検出

AWS は、ワークロードに関するコンテキストをできるだけ理解するためにお客様と協力して作業します。AWS Incident Detection and Response は、この情報を使用して、インシデントや AWS サービスイベント中にお客様をサポートするランブックを作成します。必要な情報は Incident Detection and Response のワークロードのオンボーディングとアラームの取り込みに関するアンケート で取得されます。AppRegistry にもワークロードを登録するのがベストプラクティスです。詳細については、AppRegistry のユーザーガイドを参照してください。

重要なアウトプット:

- ワークロードの情報: ワークロードの説明、アーキテクチャ図、連絡先、エスカレーションの詳細など。
- ワークロードが各 AWS リージョンで AWS サービスをどのように採用しているかの詳細。
- AWS がサービスイベント中にお客様をサポートする方法に関する具体的な情報。
- ワークロードへの重大な影響を検出するチームが使用するアラーム。

ワークロードを Incident Detection and Response にサブスクライブする

AWS Incident Detection and Response にワークロードをサブスクライブするには、ワークロードごとに新しいサポートケースを作成します。サポートケースを作成する際は、次の点に注意してください。

- 1 つの AWS アカウントにあるワークロードをオンボードするには、ワークロードのアカウントまたは支払者アカウントからサポートケースを作成します。
- 複数の AWS アカウントにまたがるワークロードをオンボードするには、[支払者アカウント] からサポートケースを作成します。サポートケースの本文で、オンボードするすべてのアカウント ID を記載します。

> **Important**
>
> Incident Detection and Response にサブスクライブするワークロードのサポートケースを作成するアカウントを間違えると、ワークロードをサブスクライブするまでに遅延が発生したり、追加情報が要求されたりする場合があります。

ワークロードをサブスクライブするには

1. 次の例に示すように、AWS サポート センターに移動し、[ケースの作成] を選択します。ワークロードは、エンタープライズサポートに登録されているアカウントからのみサブスクライブできます。

2. サポートケースのフォームに入力します。

- [テクニカルサポート] を選択します。
- [サービス] で、[Incident Detection and Response] を選択します。
- [カテゴリ] で、[新しいワークロードをオンボード] を選択します。
- [重要度] で、[一般的なガイダンス] を選択します。

3. この変更の [件名] を入力します。例：

   [オンボード] AWS Incident Detection and Response - *workload_name*

4. この変更の [説明] を入力します。例えば、「このリクエストは、ワークロードを AWS Incident Detection and Response にオンボードするためのものです」と入力します。リクエストには、次の情報が含まれていることを確認してください。

   - ワークロード名: ワークロードの名前。
   - アカウント ID: ID1、ID2、ID3 など。これらは、AWS Incident Detection and Response にオンボードするアカウントです。
   - 言語: 英語または日本語。
   - サブスクリプション開始日: AWS Incident Detection and Response のサブスクリプションを開始する日付。

5. [追加の連絡先 - オプション] セクションに、リクエストの回答を受け取る E メール ID を入力します。

   以下は、[追加の連絡先 - オプション] セクションの例です。

   > **Important**
   > [追加の連絡先 - オプション] セクションで E メール ID を追加しなかった場合、AWS Incident Detection and Response のオンボーディングプロセスが遅れる可能性があります。

6. [送信] を選択します。

   リクエストを送信したら、組織の E メールを追加できます。E メールを追加するには、ケースに返信し、[追加の連絡先 - オプション] セクションで E メール ID を追加します。

   以下は、[追加の連絡先 - オプション] セクションの例です。

サブスクリプションをリクエストするサポートケースを作成したら、ワークロードのオンボーディングプロセスに進むため、次の 2 つのドキュメントを準備します。

- AWS ワークロードのアーキテクチャ図。
- Incident Detection and Response のワークロードのオンボーディングとアラームの取り込みに関するアンケート: オンボーディングするワークロードに関連するすべての情報をアンケートに入力します。オンボーディングするワークロードが複数ある場合は、ワークロードごとに新しいオンボーディングアンケートを作成します。オンボーディングアンケートの完了についてご質問がある場合は、テクニカルアカウントマネージャー (TAM) にお問い合わせください。

> **Note**
>
> これら 2 つのドキュメントをケースに添付する際、[ファイルを添付] オプションを使用しないでください。AWS Incident Detection and Response チームは、ケースに返信する際、お客様がドキュメントをアップロードできるよう、Amazon Simple Storage Service アップローダーのリンクを含めます。

AWS Incident Detection and Response を使用してケースを作成し、既存のオンボーディング済みワークロードへの変更をリクエストする方法については、「Incident Detection and Response でオンボードしたワークロードへの変更をリクエストする」を参照してください。ワークロードをオフボードする方法については、「Incident Detection and Response からのワークロードのオフボード」を参照してください。

Incident Detection and Response でアラームを定義および設定する

AWS は、アプリケーションとその基盤となる AWS インフラストラクチャのパフォーマンスを可視化するため、お客様と協力してメトリクスとアラームを定義します。しきい値を定義および設定する際は、アラームが次の基準に準拠する必要があります。

- アラームは、モニタリング対象のワークロードに重大な影響 (収益の損失またはパフォーマンスを大幅に低下させるカスタマーエクスペリエンスの低下) があり、オペレーターによる即時の注意が必要な場合にのみ「Alarm」状態になります。

- また、アラームは、インシデント管理チームを関与させると同時に、または関与させる前に、ワークロード向けに指定したリゾルバーを関与させる必要があります。インシデント管理エンジニアは、緩和プロセスでお客様が指定したリゾルバーと連携しますが、エスカレーションする第一線の応答者としては機能しません。
- アラームのしきい値は、アラームが発生したときに調査が行われるように、適切なしきい値と期間に設定する必要があります。アラームが「Alarm」状態と「OK」状態の間でフラッピングしている場合、オペレーターの応答と注意を必要とする十分な影響が発生しています。

アラームのタイプ:

- ビジネスへの影響のレベルを示し、単純な障害検出のために関連情報を渡すアラーム。
- Amazon CloudWatch canary。詳細については、「Canary と X-Ray のトレース」および「X-Ray」を参照してください。
- アラームの集計 (依存関係のモニタリング)

次の表に、CloudWatch モニタリングシステムを使用した、アラームの例を示します。

メトリクス名/アラームしきい値	アラームの ARN またはリソース ID	このアラームが発生した場合	関与が認められる場合、プレミアムサポートケースを発行するサービス
API エラー/ エラー数 >= 10 個のデータポイントで 10 回	arn:aws:cloudwatch:us-west-2:000000000000:alarm:E2MPmimLambda-Errors	データベース管理者 (DBA) チームにチケットを提出	Lambda、API Gateway

メトリクス名/アラームしきい値	アラームの ARN またはリソース ID	このアラームが発生した場合	関与が認められる場合、プレミアムサポートケースを発行するサービス
ServiceUnavailable (Http ステータスコード 503) エラー数 >= 5 分間で 10 個のデータポイントで 3 回 (異なるクライアント)	arn:aws:cloudwatch:us-west-2:xxxxx:alarm:http errorcode503	サービスチームにチケットを提出	Lambda、API Gateway
ThrottlingException (Http ステータスコード 400) エラー数 >= 5 分間で 10 個のデータポイントで 3 回 (異なるクライアント)	arn:aws:cloudwatch:us-west-2:xxxxx:alarm:http errorcode400	サービスチームにチケットを提出	EC2、Amazon Aurora

詳細については、「AWS Incident Detection and Response のモニタリングとオブザーバビリティ」を参照してください。

重要なアウトプット:

- ワークロードのアラームの定義と設定。
- オンボーディングアンケートにアラームの詳細を入力します。

トピック

- Incident Detection and Response でビジネスニーズに合った CloudWatch アラームを作成する

- CloudFormation テンプレートを使用して Incident Detection and Response で CloudWatch アラームを作成する
- Incident Detection and Response における CloudWatch アラームのユースケースの例

Incident Detection and Response でビジネスニーズに合った CloudWatch アラームを作成する

Amazon CloudWatch アラームを作成する場合、アラームがビジネスニーズに最も適していることを確認するために実行できるいくつかのステップがあります。

> **Note**
>
> AWS のサービス が Incident Detection and Response にオンボードするための推奨される CloudWatch アラームの例については、「Incident Detection and Response Alarm Best Practices on AWS re:Post」を参照してください。

提案された CloudWatch アラームを確認する

提案されたアラームを確認して、モニタリング対象のワークロードに重大な影響 (収益の損失またはパフォーマンスを大幅に低下させるカスタマーエクスペリエンスの低下) がある場合にのみ「Alarm」状態になることを確認します。例えば、このアラームは、「Alarm」状態になった場合にすぐに対応する必要があるほど重大なものですか?

以下は、エンドユーザーのアプリケーションエクスペリエンスに影響を与えるなど、ビジネスに重大な影響を与える可能性のある推奨メトリクスです。

- CloudFront: 詳細については、「CloudFront 関数およびエッジ関数のメトリクスの表示」を参照してください。
- Application Load Balancers: 可能であれば、Application Load Balancer に対して次のアラームを作成することがベストプラクティスです。
  - HTTPCode_ELB_5XX_Count
  - HTTPCode_Target_5XX_Count

  上記のアラームにより、Application Load Balancer の背後にあるターゲットからのレスポンス、または他のリソースの背後にあるターゲットからのレスポンスをモニタリングできます。これに

より、5XX エラーの原因を簡単に特定できるようになります。詳細については、「CloudWatch metrics for your Application Load Balancer」を参照してください。

- Amazon API Gateway: Elastic Beanstalk で WebSocket API を使用している場合、次のメトリクスの使用を検討してください。
  - 統合エラー率 (5XX エラーにフィルタリング)
  - 統合のレイテンシー
  - 実行エラー

  詳細については、「CloudWatch メトリクスを使用した WebSocket API の実行のモニタリング」を参照してください。
- Amazon Route 53: EndPointUnhealthyENICount メトリクスをモニタリングします。このメトリクスは、[自動復旧] ステータスの Elastic Network Interface の数です。このステータスは、エンドポイント ([EndpointId] で指定) に関連付けられている 1 つ以上の Amazon Virtual Private Cloud ネットワークインターフェイスをリゾルバーが復旧しようとしたことを示します。復旧プロセスでは、エンドポイントは限られた容量で機能します。エンドポイントは、完全に復旧するまで DNS クエリを処理できません。詳細については、「Monitoring Route 53 Resolver endpoints with Amazon CloudWatch」を参照してください。

アラーム設定を検証する

提案されたアラームがビジネスニーズに合っていることを確認したら、アラームの設定と履歴を検証します。

- メトリクスの [しきい値] を検証して、メトリクスのグラフのトレンドに対する「Alarm」状態を入力します。
- データポイントのポーリングに使用される [期間] を検証します。データポイントを 60 秒でポーリングすると、インシデントの早期検出に役立ちます。
- [DatapointToAlarm] 設定を検証します。ほとんどの場合、これを 3 個中 3 個または 5 個中 5 個に設定するのがベストプラクティスです。インシデントでは、[60 second metrics with 3 out of 3 DatapointToAlarm] に設定すると 3 分後にアラームがトリガーされ、[60 second metrics with 5 out of 5 DatapointToAlarm] に設定すると 5 分後にアラームがトリガーされます。この組み合わせを使用して、ノイズの多いアラームを排除します。

> **Note**
>
> 上記の推奨事項は、サービスの使用方法によって異なる場合があります。AWS のサービスごとにワークロード内での動作は異なります。また、同じサービスを複数の場所で使用した場合、動作が異なる場合があります。ワークロードがアラームを供給するリソースをどのように使用するか、およびアップストリームとダウンストリームの効果を理解する必要があります。

アラームが欠落データを処理する方法を検証する

一部のメトリクスソースは、データを CloudWatch に定期的に送信しません。これらのメトリクスについては、欠落データを [notBreaching] として扱うことがベストプラクティスです。詳細については、「CloudWatch アラームの欠落データの処理の設定」および「アラーム状態への早期移行の回避」を参照してください。

例えば、メトリクスでエラー率をモニタリングし、エラーがない場合、メトリクスはデータなし (nil) のデータポイントを報告します。欠落データを [欠落] として扱うようにアラームを設定すると、1 つの違反データポイントに続いて 2 つのデータなし (nil) データポイントがあると、メトリクスは「Alarm」状態になります (データポイント 3 個中 3 個)。これは、欠落データの設定が評価期間内の最後の既知のデータポイントを評価するためです。

メトリクスでエラー率をモニタリングする場合、サービスの低下がない限り、データがないのは良いことだと考えることができます。欠落データを [notBreaching] として扱うことがベストプラクティスです。これにより、欠落データは「OK」として扱われ、メトリクスが単一のデータポイントで「Alarm」状態になることはありません。

各アラームの履歴を確認する

アラームの履歴が、頻繁に「Alarm」状態になるものの、すぐに回復していることを示す場合、アラームが問題になっている可能性があります。ノイズや誤アラームを防ぐために、アラームを調整してください。

基盤となるリソースのメトリクスを検証する

メトリクスが、基盤となる有効なリソースを参照し、正しい統計情報を使用していることを確認します。無効なリソース名を確認するようにアラームが設定されている場合、アラームは基盤となるデータを追跡できない可能性があります。これにより、アラームが「Alarm」状態になる場合があります。

複合アラームを作成する

Incident Detection and Response オペレーションにオンボーディング用のアラームを多数提供すると、複合アラームを作成するように求められる場合があります。複合アラームにより、オンボードする必要があるアラームの総数を減らすことができます。

CloudFormation テンプレートを使用して Incident Detection and Response で CloudWatch アラームを作成する

AWS Incident Detection and Response へのオンボーディングを高速化し、アラームの作成に必要な労力を削減するために、AWS は AWS CloudFormation テンプレートを提供しています。テンプレートには、Application Load Balancer、Network Load Balancer、Amazon CloudFront など、一般的にオンボーディングされるサービス用に最適化されたアラーム設定が含まれています。

CloudFormation テンプレートを使用して CloudWatch アラームを作成する

1. 提供されたリンクを使用してテンプレートをダウンロードします。

NameSpac	メトリクス	ComparisonOperator (しきい値)	期間	DatapointsToAlarm	TreatMissingData	統計	テンプレートへのリンク
Application Elastic Load Balancer	(m1+m2)/(m1+m2+mm4)*100 m1=HTTPCode_Target_2XX_Count m2=HTTPCode_Target_3XX_Count m3=HTTPCode_Target_4XX_Cou	LessThanThreshold(95)	60	3 個中 3 個	missing	合計	テンプレート

NameSpac	メトリクス	ComparisonOperator (しきい値)	期間	DatapointsToAlarm	TreatMissingData	統計	テンプレートへのリンク
	t m4=HTTP(de_Target _5XX_Cou t						
Amazon CloudFront	TotalErrorRate	GreaterThanThreshold(5)	60	3 個中 3 個	notBreaching	平均	テンプレート
Application Elastic Load Balancer	UnHealthyHostCount	GreaterThanOrEqualToThreshold(2)	60	3 個中 3 個	notBreaching	最大値	テンプレート
Network Elastic Load Balancer	UnHealthyHostCount	GreaterThanOrEqualToThreshold(2)	60	3 個中 3 個	notBreaching	最大値	テンプレート

2. ダウンロードした JSON ファイルを確認し、組織の運用およびセキュリティのプロセスを満たしていることを確認します。
3. CloudFormation スタックを作成します。

> **Note**
>
> 次の手順では、標準の CloudFormation スタック作成プロセスを使用します。詳細な手順については、「AWS CloudFormation コンソールでのスタックの作成」を参照してください。

a. https://console.aws.amazon.com/cloudformation で AWS CloudFormation コンソール を開きます。

b. [Create stack] を選択します。

c. [テンプレートの準備完了] を選択し、ローカルフォルダからテンプレートファイルをアップロードします。

   [スタックを作成] 画面の例を次に示します。

d. [次へ] を選択します。

e. 以下の必須情報を入力します。

   - [AlarmNameConfig] と [AlarmDescriptionConfig]: アラームの名前と説明を入力します。

   - [ThresholdConfig]: アプリケーションの要件を満たすようにしきい値を変更します。

   - [DistributionIDConfig]: ディストリビューション ID が、AWS CloudFormation スタックを作成するアカウントの正しいリソースを指していることを確認します。

f. [次へ] を選択します。

g. [PeriodConfig]、[EvalutionPeriodConfig]、[DatapointsToAlarmConfig] の各フィールドのデフォルト値を確認します。これらのフィールドにはデフォルト値を使用するのがベストプラクティスです。必要に応じて、アプリケーションの要件に合わせて調整できます。

h. 必要に応じて、タグと SNS 通知情報を入力します。アラームが誤って削除されないように、[終了の保護] を有効にするのがベストプラクティスです。[終了の保護] を有効にするには、次の例に示すように、[アクティブ化] ラジオボタンを選択します。

i. [次へ] を選択します。

j. スタックの設定を確認し、[スタックを作成] を選択します。

k. スタックを作成すると、次の例に示すように、Amazon CloudWatch の [アラーム] リストにアラームが表示されます。

4. すべてのアラームを正しいアカウントおよび AWS リージョンで作成したら、テクニカルアカウントマネージャー (TAM) に通知します。AWS Incident Detection and Response チームは、新しいアラームのステータスを確認し、オンボーディングを続行します。

Incident Detection and Response における CloudWatch アラームのユースケースの例

Incident Detection and Response で Amazon CloudWatch アラームを使用する方法の例については、次のユースケースを参照してください。以下の例では、AWS のさまざまなサービスにわたって主要なメトリクスとしきい値をモニタリングするように CloudWatch アラームを設定し、アプリケーションやワークロードの可用性とパフォーマンスに影響を与える可能性がある潜在的な問題を特定して対応できるようにする方法を示します。

ユースケース A の例: Application Load Balancer

ワークロードへの潜在的な影響を通知する次の CloudWatch アラームを作成できます。これを行うには、正常な接続が特定のしきい値を下回ったときにアラームを発するメトリクス数式を作成します。使用可能な CloudWatch メトリクスについては、「CloudWatch metrics for your Application Load Balancer」を参照してください。

メトリクス:`HTTPCode_Target_3XX_Count;HTTPCode_Target_4XX_Count;HTTPCode_Target_5XX_Count.(m1+m2)/(m1+m2+m3+m4)*100 m1 = HTTP Code 2xx || m2 = HTTP Code 3xx || m3 = HTTP Code 4xx || m4 = HTTP Code 5xx`

名前空間: AWS/ApplicationELB

ComparisonOperator (しきい値): x 未満 (x = お客様のしきい値)。

期間: 60 秒

DatapointsToAlarm: 3 個中 3 個

欠落データの処理: 欠落データをしきい値を超過として処理します。

統計: Sum

次の図は、ユースケース A のフローを示しています。

ユースケース B の例: Amazon API Gateway

ワークロードへの潜在的な影響を通知する次の CloudWatch アラームを作成できます。これを行うには、API Gateway でレイテンシーが高いか、4XX エラーの数が平均して多い場合に、アラームを発

する複合メトリクスを作成します。使用可能なメトリクスについては、「Amazon API Gateway のディメンションとメトリクス」を参照してください。

メトリクス:`compositeAlarmAPI Gateway (ALARM(error4XXMetricApiGatewayAlarm)) OR (AALARM(latencyMetricApiGatewayAlarm))`

名前空間: AWS/API Gateway

ComparisonOperator (しきい値): x または y より大きい (x または y = お客様のしきい値)。

期間: 60 秒

DatapointsToAlarm: 1 個中 1 個

欠落データの処理: 欠落データをしきい値内として処理します。

統計:

次の図は、ユースケース B のフローを示しています。

ユースケース C の例: Amazon Route 53

Route 53 のヘルスチェックを作成すると、リソースをモニタリングできます。ヘルスチェックでは、CloudWatch を使用して未加工データを収集し、読み取り可能なほぼリアルタイムのメトリクスに加工します。ワークロードへの潜在的な影響を通知する次の CloudWatch アラームを作成できます。CloudWatch メトリクスを使用して、確立されたしきい値を超えたときにトリガーするアラームを作成できます。利用可能な CloudWatch メトリクスについては、「Route 53 ヘルスチェックの CloudWatch メトリクス」を参照してください。

メトリクス:`R53-HC-Success`

名前空間: AWS/Route 53

HealthCheckStatus のしきい値: HealthCheckStatus が 3 分以内に 3 個のデータポイントで x 未満 (x = お客様のしきい値)

期間: 1 分

DatapointsToAlarm: 3 個中 3 個

欠落データの処理: 欠落データをしきい値を超過として処理します。

統計: Minimum

次の図は、ユースケース C のフローを示しています。

ユースケース D の例: カスタムアプリケーションでワークロードをモニタリングする

このシナリオでは、時間をかけて適切なヘルスチェックを定義することが重要です。アプリケーションのポートが開いていることのみを検証する場合、アプリケーションが動作していることは検証しません。さらに、アプリケーションのホームページを呼び出すことは、アプリケーションが動作しているかどうかを判断する正しい方法とは限りません。例えば、アプリケーションがデータベースと Amazon Simple Storage Service (Amazon S3) の両方に依存している場合、ヘルスチェックではすべての要素を検証する必要があります。そのための 1 つの方法は、/monitor などのモニタリングウェブページを作成することです。モニタリングウェブページは、データベースを呼び出して、データを接続および取得できることを確認します。さらに、モニタリングウェブページは Amazon S3 を呼び出します。次に、ロードバランサーのヘルスチェックを /monitor ページに指定します。

次の図は、ユースケース D のフローを示しています。

AWS Incident Detection and Response にアラートを取り込む

AWS Incident Detection and Response は、Amazon EventBridge を介したアラームの取り込みをサポートしています。このセクションでは、AWS Incident Detection and Response をさまざまなアプリケーションパフォーマンスモニタリング (APM) ツールと統合する方法について説明します。これには、Amazon CloudWatch、Amazon EventBridge と直接統合する APM (Datadog や New Relic など)、Amazon EventBridge と直接統合しない APM が含まれます。Amazon EventBridge と直接統合している APM の詳細なリストについては、「Amazon EventBridge の統合」を参照してください。

トピック

- Incident Detection and Response にアラートを取り込むためのアクセスをプロビジョニングする
- Incident Detection and Response を Amazon CloudWatch と統合する
- Amazon EventBridge と直接統合されている APM からアラームを取り込む
- 例: Datadog と Splunk からの通知を統合する
- ウェブフックを使用して Amazon EventBridge と直接統合していない APM からアラームを取り込む

Incident Detection and Response にアラートを取り込むためのアクセスをプロビジョニングする

AWS Incident Detection and Response がアカウントからアラームを取り込むことができるようにするには、`AWSServiceRoleForHealth_EventProcessor` サービスにリンクされたロール (SLR) をインストールします。AWS は、SLR を引き受けて Amazon EventBridge マネージドルールを作成します。マネージドルールは、アカウントから AWS Incident Detection and Response に通知を送信します。関連する AWS マネージドポリシーを含むこの SLR の詳細については、「AWS Health ユーザーガイド」の「Using service-linked roles」を参照してください。

このサービスにリンクされたロールをアカウントにインストールするには、「AWS Identity and Access Management ユーザーガイド」の「サービスにリンクされたロールの作成」の手順に従います。または、AWS コマンドラインインターフェイス (AWS CLI) を使用することもできます。

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

重要なアウトプット

- サービスリンクロールがアカウントに正常にインストールされました。

関連情報

詳細については、以下の各トピックを参照してください。

- AWS Health でのサービスリンクロールの使用
- サービスにリンクされたロールの作成
- AWS マネージドポリシー: AWSHealth_EventProcessorServiceRolePolicy

Incident Detection and Response を Amazon CloudWatch と統合する

AWS Incident Detection and Response は、アクセスプロビジョニング中にオンにしたサービスにリンクされたロール (SLR) を使用して、`AWSHealthEventProcessor-DO-NOT-DELETE` という名前の AWS アカウントに Amazon EventBridge マネージドルールを作成します。Incident Detection and Response では、このルールを使用して、アカウントから Amazon CloudWatch アラームを取り込みます。CloudWatch からアラームを取り込む場合は、追加の手順は必要ありません。

Amazon EventBridge と直接統合されている APM からアラームを取り込む

次の図は、Datadog や Splunk のような Amazon EventBridge と直接統合されているアプリケーションパフォーマンスモニタリング (APM) ツールから AWS Incident Detection and Response に通知を送信するプロセスを示しています。EventBridge と直接統合している APM の詳細なリストについては、「Amazon EventBridge の統合」を参照してください。

次の手順を使用して、AWS Incident Detection and Response との統合を設定します。手順を実行する前に、AWS サービスにリンクされたロール (SLR) `AWSServiceRoleForHealth_EventProcessor` がアカウントにインストールされていることを確認します。

AWS Incident Detection and Response との統合を設定する

各 AWS アカウントおよび各 AWS リージョンで次の手順を実行する必要があります。アラートは、アプリケーションリソースが存在する AWS アカウントと AWS リージョンから送信する必要があります。

1. 各 APM を Amazon EventBridge パートナーイベントソースとして設定します (例: `aws.partner/my_apm/integrationName`)。APM をイベントソースとして設定するガイドラインについては、「Receiving events from a SaaS partner with Amazon EventBridge」を参照してください。これにより、アカウントにパートナーイベントバスが作成されます。
2. 次のいずれかを行います：
   - (推奨メソッド) カスタムの EventBridge イベントバスを作成します。AWS Incident Detection and Response は、`AWSServiceRoleForHealth_EventProcessor` SLR を介してマネージドルール (`AWSHealthEventProcessorEventSource-DO-NOT-DELETE`) バスをインストールします。ルールソースはカスタムのイベントバスです。ルールの送信先は AWS Incident Detection and Response です。このルールは、サードパーティーの APM イベントを取り込むためのパターンと一致します。
   - (代替方法) カスタムイベントバスの代わりにデフォルトのイベントバスを使用します。デフォルトのイベントバスでは、マネージドルールが AWS Incident Detection and Response に APM アラートを送信する必要があります。
3. パートナーイベントバスイベントを変換する AWS Lambda 関数 (例: `My_APM-AWSIncidentDetectionResponse-LambdaFunction`) を作成します。変換されたイベントは、マネージドルール `AWSHealthEventProcessorEventSource-DO-NOT-DELETE` と一致します。

a. 変換されたイベントには一意の AWS Incident Detection and Response の識別子が含まれ、イベントのソースタイプと詳細タイプを必要な値に設定します。このパターンはマネージドルールと一致します。

b. Lambda 関数のターゲットを、ステップ 2 (推奨メソッド) で作成したカスタムのイベントバスまたはデフォルトのイベントバスに設定します。

4. EventBridge ルールを作成し、AWS Incident Detection and Response にプッシュするイベントのリストと一致するイベントパターンを定義します。ルールのソースは、ステップ 1 で定義したパートナーイベントバスです (例えば、aws.partner/my_apm/integrationName)。ルールのターゲットは、ステップ 3 で定義した Lambda 関数です (例: `My_APM-AWSIncidentDetectionResponse-LambdaFunction`)。EventBridge ルールの定義に関するガイドラインについては、「Amazon EventBridge ルール」を参照してください。

AWS Incident Detection and Response で使用するパートナーイベントバスの統合を設定する方法の例については、「例: Datadog と Splunk からの通知を統合する」を参照してください。

例: Datadog と Splunk からの通知を統合する

この例では、Datadog と Splunk からの通知を AWS Incident Detection and Response に統合するための詳細な手順を示します。

トピック

- ステップ 1: APM を Amazon EventBridge のイベントソースとしてセットアップする
- ステップ 2: カスタムイベントバスを作成する
- ステップ 3: 変換用の AWS Lambda 関数を作成する
- ステップ 4: カスタムの Amazon EventBridge ルールを作成する

ステップ 1: APM を Amazon EventBridge のイベントソースとしてセットアップする

AWS アカウントの Amazon EventBridge で、各 APM をイベントソースとしてセットアップします。APM をイベントソースとして設定する手順については、「event source set up instructions for your tool in Amazon EventBridge partners」を参照してください。

APM をイベントソースとして設定することで、APM から AWS アカウントのイベントバスに通知を取り込むことができます。セットアップ後にイベントバスがイベントを受信すると、AWS Incident Detection and Response がインシデント管理プロセスを開始できます。このプロセスにより、Amazon EventBridge が APM の送信先として追加されます。

ステップ 2: カスタムイベントバスを作成する

カスタムイベントバスを使用するのがベストプラクティスです。AWS Incident Detection and Response は、カスタムイベントバスを使用して、変換されたイベントを取り込みます。AWS Lambda 関数はパートナーイベントバスのイベントを変換し、カスタムイベントバスに送信します。AWS Incident Detection and Response は、カスタムイベントバスからイベントを取り込むためのマネージドルールをインストールします。

カスタムイベントバスの代わりにデフォルトのイベントバスを使用することもできます。AWS Incident Detection and Response では、カスタムイベントバスの代わりに、デフォルトのイベントバスから取り込むようにマネージドルールを変更します。

AWS アカウントにカスタムイベントバスを作成する方法:

1. Amazon EventBridge コンソール (https://console.aws.amazon.com/events/) を開きます。
2. [バス]、[イベントバス] の順に選択します。
3. [カスタムイベントバス] で、[作成] を選択します。
4. [名前] でイベントバスの名前を指定します。推奨される形式は、[APMName-AWSIncidentDetectionResponse-EventBus] です。

   例として、Datadog または Splunk を使用する場合は、次のいずれかを使用します。

   - [Datadog]: Datadog-AWSIncidentDetectionResponse-EventBus
   - [Splunk]: Splunk-AWSIncidentDetectionResponse-EventBus

ステップ 3: 変換用の AWS Lambda 関数を作成する

Lambda 関数は、ステップ 1 のパートナーイベントバスとステップ 2 のカスタム (またはデフォルト) イベントバスの間でイベントを変換します。Lambda 関数による変換は、AWS Incident Detection and Response マネージドルールと一致します。

AWS アカウントに AWS Lambda 関数を作成する

1. AWS Lambda コンソールの [Functions] (関数) ページを開きます。
2. [関数の作成] を選択してください。
3. [一から作成] タブを選択します。
4. [関数名] には、`APMName-AWSIncidentDetectionResponse-LambdaFunction` の形式を使って名前を入力します。

Datadog と Splunk の例を次に示します。

- [Datadog]: Datadog-AWSIncidentDetectionResponse-LambdaFunction
- [Splunk]: Splunk-AWSIncidentDetectionResponse-LambdaFunction

5. [ランタイム] には、「Python 3.10」と入力します。
6. 残りのフィールドはデフォルト値のままにします。[関数の作成] を選択してください。
7. [コード編集] ページで、デフォルトの Lambda 関数コンテンツを次のコード例の関数に置き換えます。

   次のコード例の # で始まるコメントに注意してください。これらのコメントは、変更する値を示します。

   Datadog 変換コードテンプレート:

```
import logging
import json
import boto3

logger = logging.getLogger()
logger.setLevel(logging.INFO)

# Change the EventBusName to the custom event bus name you created previously or
 use your default event bus which is called 'default'.
# Example 'Datadog-AWSIncidentDetectionResponse-EventBus'
EventBusName = "Datadog-AWSIncidentDetectionResponse-EventBus"

def lambda_handler(event, context):
    # Set the event["detail"]["incident-detection-response-identifier"] value to
 the name of your alert that is coming from your APM. Each APM is different and
 each unique alert will have a different name.
    # Replace the dictionary path, event["detail"]["meta"]["monitor"]["name"], with
 the path to your alert name based on your APM payload.
    # This example is for finding the alert name for Datadog.
    event["detail"]["incident-detection-response-identifier"] = event["detail"]
["meta"]["monitor"]["name"]
    logger.info(f"We got: {json.dumps(event, indent=2)}")

    client = boto3.client('events')
    response = client.put_events(
     Entries=[
              {
```

```
                'Detail': json.dumps(event["detail"], indent=2),
                'DetailType': 'ams.monitoring/generic-apm', # Do not modify. This
 DetailType value is required.
                'Source': 'GenericAPMEvent', # Do not modify. This Source value is
 required.
                'EventBusName': EventBusName # Do not modify. This variable is set
 at the top of this code as a global variable. Change the variable value for your
 eventbus name at the top of this code.
            }
     ]
    )
    print(response['Entries'])
```

Splunk 変換コードテンプレート:

```
import logging
import json
import boto3

logger = logging.getLogger()
logger.setLevel(logging.INFO)

# Change the EventBusName to the custom event bus name you created previously or
 use your default event bus which is called 'default'.
# Example Splunk-AWSIncidentDetectionResponse-EventBus
EventBusName = "Splunk-AWSIncidentDetectionResponse-EventBus"

def lambda_handler(event, context):
    # Set the event["detail"]["incident-detection-response-identifier"] value to
 the name of your alert that is coming from your APM. Each APM is different and
 each unique alert will have a different name.
    # replace the dictionary path event["detail"]["ruleName"] with the path to your
 alert name based on your APM payload.
    # This example is for finding the alert name in Splunk.
    event["detail"]["incident-detection-response-identifier"] = event["detail"]
["ruleName"]
    logger.info(f"We got: {json.dumps(event, indent=2)}")

    client = boto3.client('events')
    response = client.put_events(
    Entries=[
            {
             'Detail': json.dumps(event["detail"], indent=2),
```

```
            'DetailType': 'ams.monitoring/generic-apm', # Do not modify. This
DetailType value is required.
            'Source': 'GenericAPMEvent', # Do not modify. This Source value is
required.
            'EventBusName': EventBusName # Do not modify. This variable is set
at the top of this code as a global variable. Change the variable value for your
eventbus name at the top of this code.
          }
      ]
   )
   print(response['Entries'])
```

8. [デプロイ] を選択します。
9. 変換されたデータの送信先となるイベントバスの Lambda 実行ロールに [PutEvents] のアクセス許可を追加します。
   a. AWS Lambda コンソールの [Functions] (関数) ページを開きます。
   b. 関数を選択してから、[設定] タブで [アクセス許可] を選択します。
   c. [実行ロール] で [ロール名] を選択して、AWS Identity and Access Management コンソールで実行ロールを開きます。
   d. [アクセス許可ポリシー] で、既存のポリシー名を選択してポリシーを開きます。
   e. [このポリシーで定義されている許可] で、[編集] を選択します。
   f. [ポリシーエディタ] ページで、[新しいステートメントを追加] を選択します。
   g. [ポリシーエディタ] により、次のような新しい空白のステートメントが追加されます。
   h. 自動生成された新しいステートメントを以下に置き換えます。

```
{
  "Sid": "AWSIncidentDetectionResponseEventBus0",
  "Effect": "Allow",
  "Action": "events:PutEvents",
  "Resource": "arn:aws:events:{region}:{accountId}:event-bus/{custom-eventbus-
name}"
}
```

   i. [リソース] は、ステップ 2: カスタムイベントバスを作成する で作成したカスタムイベントバスの ARN、または Lambda コードでデフォルトのイベントバスを使用している場合はデフォルトのイベントバスの ARN です。

10. 必要なアクセス許可がロールに追加されていることを確認します。
11. [この新しいバージョンをデフォルトとして設定] を選択し、[変更を保存] を選択します。

ペイロード変換には何が必要ですか?

AWS Incident Detection and Response によって取り込まれるイベントバスのイベントには、次の JSON キーと値のペアが必要です。

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

次の例は、変換前と変換後のパートナーイベントバスからのイベントを示しています。

```
{
    "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}
 \u003c\u003d 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"
      ]
    }
}
```

イベントが変換される前は、`detail-type` はアラートが発生した APM、ソースはパートナー APM、そして `incident-detection-response-identifier` キーが存在していないことに注意してください。

Lambda 関数は上記のイベントを変換し、ターゲットのカスタムイベントバスまたはデフォルトのイベントバスに配置します。変換されたペイロードに、必要なキーと値のペアが含まれるようになりました。

```
{
    "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}
 \u003c\u003d 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"
      ]
    }
}
```

`detail-type` は `ams.monitoring/generic-apm`、ソースは `GenericAPMEvent` になり、詳細には新しいキーと値のペアである `incident-detection-response-identifier` が存在するようになりました。

前の例では、`incident-detection-response-identifier` 値はパス `$.detail.meta.monitor.name` の下のアラート名から取得されています。APM アラート名のパスは、APM ごとに異なります。Lambda 関数を変更して、正しいパートナーイベント JSON パスからアラーム名を取得し、`incident-detection-response-identifier` 値に使用する必要があります。

`incident-detection-response-identifier` で設定される一意の名前はそれぞれ、オンボーディング中に AWS Incident Detection and Response チームに提供されます。`incident-detection-response-identifier` の名前が不明なイベントは処理されません。

ステップ 4: カスタムの Amazon EventBridge ルールを作成する

ステップ 1 で作成したパートナーイベントバスには、お客様が作成する EventBridge ルールが必要です。このルールは、パートナーイベントバスから、ステップ 3 で作成した Lambda 関数に目的のイベントを送信します。

EventBridge ルールの定義に関するガイドラインについては、「Amazon EventBridge ルール」を参照してください。

1. Amazon EventBridge コンソール (https://console.aws.amazon.com/events/) を開きます。

2. [ルール] を選択し、APM に関連付けられたパートナーイベントバスを選択します。パートナーイベントバスの例を次に示します。

   - [Datadog]: aws.partner/datadog.com/eventbus-name
   - [Splunk]: aws.partner/signalfx.com/RandomString
3. [ルールを作成] を選択して、新しい EventBridge ルールを作成します。
4. ルール名には、次の形式 `APMName-AWS Incident Detection and Response-EventBridgeRule` で名前を入力し、[次へ] を選択します。以下は名前の例です。

   - [Datadog]: Datadog-AWSIncidentDetectionResponse-EventBridgeRule
   - [Splunk]: Splunk-AWSIncidentDetectionResponse-EventBridgeRule
5. [イベントソース] で、[AWS イベントまたは EventBridge パートナーイベント] を選択します。
6. [サンプルイベント] と [作成方法] をデフォルト値のままにします。
7. [イベントパターン] の場合は、次のいずれかを実行します。

   a. [イベントソース]: EventBridge パートナー。
   b. [パートナー]: APM パートナーを選択します。
   c. [イベントタイプ]: すべてのイベント。

   イベントパターンの例を次に示します。

   Datadog イベントパターンの例

   Splunk イベントパターンの例

8. [ターゲット] で、以下を選択します。

   a. [ターゲットタイプ]: AWS サービス
   b. [ターゲットを選択]: Lambda 関数を選択します。
   c. [関数]: ステップ 2 で作成した Lambda 関数の名前。
9. [次へ]、[ルールを保存] の順に選択します。

ウェブフックを使用して Amazon EventBridge と直接統合していない APM からアラームを取り込む

AWS Incident Detection and Response では、Amazon EventBridge と直接統合していないサードパーティーの APM からアラームを取り込む場合のウェブフックの使用をサポートしています。

Amazon EventBridge と直接統合している APM のリストについては、「Amazon EventBridge の統合」を参照してください。

次の手順を使用して、AWS Incident Detection and Response との統合を設定します。ステップを実行する前に、AWS マネージドルール、AWSHealthEventProcessorEventSource-DO-NOT-DELETE がアカウントにインストールされていることを確認します。

ウェブフックを使用したイベントの取り込み

1. APM からのペイロードを受け入れる Amazon API Gateway を定義します。
2. 前の図に示すように、認証トークンを使用して認可用の AWS Lambda 関数を定義します。
3. 2 番目の Lambda 関数を定義して、AWS Incident Detection and Response 識別子を変換し、ペイロードに追加します。この関数を使用すると、AWS Incident Detection and Response に送信するイベントをフィルタリングすることもできます。
4. API Gateway で生成された URL に通知を送信するよう APM を設定します。

Incident Detection and Response でワークロードを管理する

効果的なインシデント管理で重要な部分は、モニタリング対象のワークロードのオンボーディング、テスト、維持に適したプロセスと手順を設定することです。このセクションでは、インシデント中のチームを導くための包括的なランブックと対応計画の作成、オンボーディング前の新しいワークロードの徹底したテストと検証、ワークロードのモニタリングを更新する変更のリクエスト、必要に応じたワークロードの適切なオフボーディングなど、重要なステップについて説明します。

トピック

- Incident Detection and Response でインシデントに対応するためのランブックと対応計画を作成する
- Incident Detection and Response でオンボードしたワークロードをテストする
- Incident Detection and Response でオンボードしたワークロードへの変更をリクエストする
- Incident Detection and Response との連動によるアラームの抑制
- Incident Detection and Response からのワークロードのオフボード

Incident Detection and Response でインシデントに対応するためのランブックと対応計画を作成する

Incident Detection and Response では、オンボーディングアンケートから取得した情報を使用して、ワークロードに影響するインシデントを管理するためのランブックと対応計画を作成します。ランブックは、Incident Manager がインシデントに対応するときに実行するステップを文書化したものです。対応計画は、少なくとも 1 つのワークロードにマッピングされます。インシデント管理チームは、ワークロードの検出で提供された情報から、これらのテンプレートを作成します。対応計画は、インシデントのトリガーに使用される AWS Systems Manager (SSM) ドキュメントテンプレートです。SSM ドキュメントの詳細については、「AWS Systems Manager ドキュメント」を参照してください。Incident Manager の詳細については、「AWS Systems Manager Incident Manager とは」を参照してください。

重要なアウトプット:

- AWS Incident Detection and Response に関するワークロードの定義を入力します。
- AWS Incident Detection and Response に関するアラーム、ランブック、対応計画の定義を入力します。

AWS Incident Detection and Response ランブックの例、aws-idr-runbook-example.zip をダウンロードすることもできます。

ランブックの例:

```
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.
```

Incident Detection and Response でオンボードしたワークロードをテストする

> **Note**
> アラームテストに使用する AWS Identity and Access Management ユーザーまたはロールには `cloudwatch:SetAlarmState` の権限が必要です。

オンボーディングプロセスの最後のステップは、新しいワークロードのゲームデーを実行することです。アラームの取り込みが完了すると、AWS Incident Detection and Response は、ゲームデーを開始するために選択した日時を確認します。

ゲームデーには主に次の 2 つの目的があります。

- 機能検証: AWS Incident Detection and Response がアラームイベントを正しく受信できることを確認します。また、機能検証では、アラームイベントが適切なランブックをトリガーし、自動ケース作成など (アラームの取り込み中に選択した場合)、その他の必要なアクションがトリガーされることを確認します。
- シミュレーション: ゲームデーは、実際にインシデントが発生した場合に起きる可能性があることをエンドツーエンドでシミュレートします。AWS Incident Detection and Response は、実際のインシデントがどのように展開されるかに関するインサイトを提供するために、規定のランブックのステップに従います。ゲームデーは、エンゲージメントを向上させるために質問したり、指示を改良したりする機会です。

アラームテスト中、AWS Incident Detection and Response はお客様と協力して、特定された問題を修正します。

CloudWatch アラーム

AWS Incident Detection and Response は、アラームの状態の変化をモニタリングすることで、Amazon CloudWatch アラームをテストします。これを行うには、AWS Command Line Interface を使用してアラームを [Alarm] 状態に手動で変更します。AWS CLI は AWS CloudShell からアクセスできます。AWS Incident Detection and Response には、テスト中に使用できる AWS CLI コマンドのリストが用意されています。

アラームの状態を設定する AWS CLI コマンドの例:

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

CloudWatch アラームの状態を手動で変更する方法の詳細については、「SetAlarmState」を参照してください。

CloudWatch API オペレーションに必要なアクセス許可の詳細については、「Amazon CloudWatch の許可リファレンス」を参照してください。

サードパーティーの APM アラーム

Datadog、Splunk、New Relic、Dynatrace などのサードパーティのアプリケーションパフォーマンスモニタリング (APM) ツールを利用するワークロードでは、アラームをシミュレートするためのさまざまな手順が必要です。ゲームデーの開始時に、AWS Incident Detection and Response は、アラームのしきい値または比較演算子を一時的に変更して、アラームを [ALARM] ステータスに強制するようリクエストします。このステータスは、AWS Incident Detection and Response へのペイロードをトリガーします。

重要なアウトプット

重要なアウトプット:

- アラームの取り込みが成功し、アラームも正しく設定されています。
- アラームは AWS Incident Detection and Response によって正常に作成され、受信されます。
- サポートケースがエンゲージメント用に作成され、所定の連絡先に通知されます。
- AWS Incident Detection and Response は、所定の会議手段で利用できます。
- ゲームデーの一部として生成されたすべてのアラームとサポートケースが解決されます。

- 本番稼働 E メールは、ワークロードが AWS Incident Detection and Response によってモニタリングされていることを確認するために送信されます。

Incident Detection and Response でオンボードしたワークロードへの変更をリクエストする

オンボーディングされたワークロードの変更をリクエストするには、次の手順を実行して、AWS Incident Detection and Response でサポートケースを作成します。

1. 次の例に示すように、AWS サポート センターに移動し、[ケースの作成] を選択します。
2. [技術] を選択します。
3. [サービス] で、[Incident Detection and Response] を選択します。
4. [カテゴリ] で、[ワークロードの変更リクエスト] を選択します。
5. [重要度] で、[一般的なガイダンス] を選択します。
6. この変更の [件名] を入力します。例：

   AWS Incident Detection and Response - *workload_name*
7. この変更の [説明] を入力します。例えば、「このリクエストは、AWS Incident Detection and Response にオンボーディングされた既存のワークロードを変更するためのものです」と入力します。リクエストには、次の情報が含まれていることを確認してください。

   - ワークロード名: ワークロードの名前。
   - アカウント ID: ID1、ID2、ID3 など。
   - 変更の詳細: リクエストした変更の詳細を入力します。
8. [追加の連絡先 - オプション] セクションに、この変更に関する連絡を受け取る E メール ID を入力します。

   次に示すのは、[追加の連絡先 - オプション] セクションの例です。

> **Important**
>
> [追加の連絡先 - オプション] セクションに E メール ID を追加しなかった場合、変更プロセスが遅れる可能性があります。

9. [Submit] を選択してください。

   変更リクエストを送信したら、組織から E メールを追加することができます。E メールを追加するには、次の例に示すように、[ケースの詳細] で [返信] を選択します。

   次に、[追加の連絡先 - オプション] セクションで、E メール ID を追加します。

   以下は、追加の E メールを入力できる場所を示す [返信] ページの例です。

Incident Detection and Response との連動によるアラームの抑制

オンボードされたワークロードアラームのうち、AWS Incident Detection and Response モニタリングと連動するものを指定し、一時的またはスケジュールに従って抑制します。例えば、計画的なメンテナンス中にワークロードアラームを一時的に抑制して、アラームが Incident Detection and Response と連動しないようにすることができます。または、毎日再起動アクティビティがある場合は、スケジュールに従ってアラームを抑制することもできます。Amazon CloudWatch などのアラームソースでアラームを抑制したり、ワークロード変更リクエストを送信したりできます。

トピック

- アラームソースでアラームを抑制
- ワークロード変更リクエストを送信してアラームを抑制
- チュートリアル: Metric Math 関数を使用してアラームを抑制
- チュートリアル: Metric Math 関数を削除してアラーム抑制を解除

アラームソースでアラームを抑制

アラームソースでアラームを抑制することで、Incident Detection and Response に連動するアラームと、連動するタイミングを指定します。

トピック

- Metric Math 関数を使用して CloudWatch アラームを抑制
- Metric Math 関数を削除して CloudWatch アラーム抑制を解除
- Metric Math 関数と関連するユースケースの例
- サードパーティー APM からのアラームを抑制

Metric Math 関数を使用して CloudWatch アラームを抑制

Amazon CloudWatch アラームの Incident Detection and Response モニタリングを抑制するには、Metric Math 関数を使用して、指定されたウィンドウ中に CloudWatch アラームが ALARM 状態に入らないようにします。

> **Note**
>
> CloudWatch のアラームで [アラームアクション] を無効にしても、Incident Detection and Response によるアラームのモニタリングは抑制されません。アラーム状態の変更は、CloudWatch のアラームアクションではなく Amazon EventBridge を介して取り込まれます。

Metric Math 関数を使用して CloudWatch アラームを抑制するには、次の手順を実行します。

1. AWS Management Console にサインインして、CloudWatch コンソール (https://console.aws.amazon.com/cloudwatch/) を開きます。
2. [アラーム] を選択し、Metric Math 関数を追加するアラームを見つけます。
3. Metric Math セクションで、[編集] を選択します。
4. [数式の追加]、[空の式から開始] の順に選択します。
5. 数式を入力し、[適用] を選択します。
6. アラームがモニタリングした既存のメトリクスの選択を解除します。
7. 先ほど作成した式を選択し、その後 [メトリクスの選択] を選択します。
8. [プレビューと作成にスキップ] を選択します。
9. 変更内容を確認して、Metric Math 関数が期待どおりに適用されていることを確認し、[アラームの更新] を選択します。

Metric Math 関数を使用して CloudWatch アラームを抑制するステップバイステップの例については、「チュートリアル: Metric Math 関数を使用してアラームを抑制」を参照してください。

構文と利用可能な関数の詳細については、「Amazon CloudWatch ユーザーガイド」の「Metric Math 構文と関数」を参照してください。

Metric Math 関数を削除して CloudWatch アラーム抑制を解除

Metric Math 関数を削除して CloudWatch アラームの抑制を解除します。アラームから Metric Math 関数を削除するには、次の手順を実行します。

1. AWS Management Console にサインインして、CloudWatch コンソール (https://console.aws.amazon.com/cloudwatch/) を開きます。
2. [アラーム] を選択し、メトリクス数式を削除するアラームを見つけます。
3. Metric Math セクションで、[編集] を選択します。
4. アラームからメトリクスを削除するには、メトリクスの [編集] を選択し、メトリクス数式の横にある [x] ボタンを選択します。
5. 元のメトリクスを選択し、[メトリクスの選択] を選択します。
6. [プレビューと作成にスキップ] を選択します。
7. 変更内容を確認して、Metric Math 関数が期待どおりに適用されていることを確認し、[アラームの更新] を選択します。

Metric Math 関数と関連するユースケースの例

次の表は、Metric Math 関数の例と、関連するユースケース、各メトリクスコンポーネントの説明を示しています。

Metric Math 関数	ユースケース	説明
`IF((DAY(m1) == 2 && HOUR(m1) >= 1 && HOUR(m1) < 3), 0, m1)`	毎週火曜日の午前 1 時から午前 3 時 (UTC) までの期間中、実際のデータポイントを 0 に置き換えることでアラームを抑制します。	• DAY(m1) == 2: 火曜日 (月曜日 = 1、日曜日 = 7) であることを確認します。 • HOUR(m1) >= 1 && HOUR(m1) > 3: 午前 1 時から午前 3 時 (UTC) の時間範囲を指定します。 • IF(condition, value_if_true, value_if_false): 条件が true の場合は、メトリクス値を 0 に置き換えます。それ以

Metric Math 関数	ユースケース	説明
		外の場合は、元の値 (m1) を返します。
`IF((HOUR(m1) >= 23 || HOUR(m1) < 4), 0, m1)`	毎日午後 11 時から午前 4 時 (UTC) までの期間中、実際のデータポイントを 0 に置き換えることでアラームを抑制します。	• HOUR(m1) >= 23: 23:00 (UTC) から始まる時間をキャプチャします。 • HOUR(m1) < 4: 04:00 (UTC) までの時間をキャプチャします (ただし、04:00 (UTC) は含まない)。 • ||: 論理 OR により、条件が深夜と早朝の 2 つの範囲で適用されるようにします。 • IF(condition, value_if_true, value_if_false): 指定された時間範囲内に 0 を返します。範囲外では元のメトリクス値 m1 を保持します。
`IF((HOUR(m1) >= 11 && HOUR(m1) < 13), 0, m1)`	毎日午前 11 時から午後 1 時 (UTC) までの期間中、実際のデータポイントを 0 に置き換えることでアラームを抑制します。	• HOUR(m1) >= 11 && HOUR(m1) < 13: 11:00 ~ 13:00 (UTC) の時間範囲をキャプチャします。 • IF(condition, value_if_true, value_if_false): 条件が true の場合 (例えば、時刻が 11:00 から 13:00 (UTC) の間)、0 を返します。条件が false の場合、元のメトリクス値 (m1) を保持します。

Metric Math 関数	ユースケース	説明
`IF((DAY(m1) == 2 && HOUR(m1) >= 1 && HOUR(m1) < 3), 99, m1)`	毎週火曜日の午前 1 時から午前 3 時 (UTC) までの期間中、実際のデータポイントを 99 に置き換えることでアラームを抑制します。	• DAY(m1) == 2: 火曜日 (月曜日 = 1、日曜日 = 7) であることを確認します。 • HOUR(m1) >= 1 && HOUR(m1) < 3: 午前 1 時から午前 3 時 (UTC) の時間範囲を指定します。 • IF(condition, value_if_true, value_if_false): 条件が true の場合は、メトリクス値を 99 に置き換えます。それ以外の場合は、元の値 (m1) を返します。
`IF((HOUR(m1) >= 23 || HOUR(m1) < 4), 100, m1)`	毎日午後 11 時から午前 4 時 (UTC) までの期間中、実際のデータポイントを 100 に置き換えることでアラームを抑制します。	• HOUR(m1) >= 23: 23:00 (UTC) から始まる時間をキャプチャします。 • HOUR(m1) < 4: 04:00 (UTC) までの時間をキャプチャします (ただし、04:00 (UTC) は含まない)。 • ||: 論理 OR により、条件が深夜と早朝の 2 つの範囲で適用されるようにします。 • IF(condition, value_if_true, value_if_false): 指定された時間範囲内に 100 を返します。範囲外では元のメトリクス値 m1 を保持します。

Metric Math 関数	ユースケース	説明
`IF((HOUR(m1) >= 11 && HOUR(m1) < 13), 99, m1)`	毎日午前 11 時から午後 1 時 (UTC) までの期間中、実際のデータポイントを 99 に置き換えることでアラームを抑制します。	• HOUR(m1) >= 11 && HOUR(m1) < 13: 11:00 ~ 13:00 (UTC) の時間範囲をキャプチャします。 • IF(condition, value_if_true, value_if_false): 条件が true の場合 (例えば、時刻が 11:00 から 13:00 (UTC) の間)、99 を返します。条件が false の場合、元のメトリクス値 (m1) を保持します。

サードパーティー APM からのアラームを抑制

アラームを抑制する方法については、サードパーティーの APM ベンダーのドキュメントを参照してください。サードパーティーの APM ベンダーの例としては、New Relic、Splunk、Dynatrace、Datadog、SumoLogic などがあります。

ワークロード変更リクエストを送信してアラームを抑制

前のセクションで説明したようにソースでアラームを抑制できない場合は、ワークロード変更リクエストを送信して、ワークロードのアラームの一部またはすべてのモニタリングを手動で抑制するように Incident Detection and Response に指示します。

ワークロード変更リクエストの作成方法の詳細については、「Incident Detection and Response でオンボードしたワークロードへの変更をリクエストする」を参照してください。ワークロード変更リクエストを発行してアラームの抑制をリクエストするときは、次の必須情報を必ず提供してください。

- ワークロード名: ワークロードの名前。
- アカウント ID: ID1、ID2、ID3 など。
- 変更の詳細: アラームの抑制
- 抑制開始時刻: 日付、時刻、タイムゾーン。
- 抑制終了時刻: 日付、時刻、タイムゾーン。

- 抑制するアラーム: 抑制する CloudWatch アラーム ARN またはサードパーティー APM イベント識別子のリスト。

アラーム抑制ワークロード変更リクエストを作成すると、Incident Detection and Response から次の通知を受け取ります。

- ワークロード変更リクエストの確認。
- アラームが抑制されたときの通知。
- モニタリングのためにアラームが再び有効になったときの通知。

チュートリアル: Metric Math 関数を使用してアラームを抑制

次のチュートリアルでは、Metric Math を使用して CloudWatch アラームを抑制する方法について説明します。

シナリオの例

次の火曜日の午前 1 時から午前 3 時 (UTC) までの間に予定されているアクティビティがあります。この時間帯の実際のデータポイントを 0 (設定されたしきい値を下回るデータポイント) に置き換える CloudWatch Metric Math 関数を作成します。

1. アラームをトリガーする基準を評価します。次のスクリーンショットは、アラーム基準の例を示しています。

   前のスクリーンショットに示したアラームは、Application Load Balancer ターゲットグループの `UnHealthyHostCount` メトリクスをモニタリングします。このアラームは、5 つのデータポイントのうち 5 つについて、`UnHealthyHostCount` メトリクスが 3 以上になると `ALARM` 状態になります。アラームは、欠落しているデータを不良 (設定されたしきい値に違反している) として扱います。
2. Metric Math 関数を作成します。

   この例では、予定されているアクティビティは、次の火曜日の午前 1 時から午前 3 時 (UTC) までの間に行われます。したがって、この時間帯の実際のデータポイントを 0 (設定されたしきい値を下回るデータポイント) に置き換える CloudWatch Metric Math 関数を作成します。

   設定する必要がある置換データポイントは、アラーム設定によって異なります。例えば、HTTP 成功率をモニタリングするアラームでしきい値が 98 未満の場合は、計画されたアクティビティ

中の実際のデータポイントを、設定されたしきい値である 100 を超える値に置き換えます。このシナリオの Metric Math 関数の例を次に示します。

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

上述の Metric Math 関数には、次の要素が含まれています。

- DAY(m1) == 2: 火曜日 (月曜日 = 1、日曜日 = 7) であることを確認します。
- HOUR(m1) >= 1 && HOUR(m1) < 3: 午前 1 時から午前 3 時 (UTC) の時間範囲を指定します。
- IF(condition, value_if_true, value_if_false): 条件が true の場合、関数はメトリクス値を 0 に置き換えます。それ以外の場合は、元の値 (m1) が返されます。

構文と利用可能な関数の詳細については、「Amazon CloudWatch ユーザーガイド」の「Metric Math 構文と関数」を参照してください。

3. AWS Management Console にサインインして、CloudWatch コンソール (https://console.aws.amazon.com/cloudwatch/) を開きます。
4. [アラーム] を選択し、Metric Math 関数を追加するアラームを見つけます。
5. Metric Math セクションで、[編集] を選択します。
6. [数式の追加]、[空の式から開始] の順に選択します。
7. 数式を入力し、[適用] を選択します。

   次の例に示すように、アラームがモニタリングする既存のメトリクスは自動的に [m1] になり、数式は [e1] になります。

8. (オプション) 次の例に示すように、メトリクス数式のラベルを編集して、その機能と作成理由を他のユーザーが理解できるようにします。

9. [m1] の選択を解除し、[e1] を選択してから、[メトリクスの選択] を選択します。これにより、基になるメトリクスを直接モニタリングする代わりに、数式をモニタリングするようにアラームが設定されます。
10. [プレビューと作成にスキップ] を選択します。
11. アラームが想定どおりに設定されていることを検証し、[アラームを更新して変更を保存] を選択します。

前の例では、Metric Math 関数が適用されていなければ、実際の `UnHealthyHostCount` メトリクスは計画されたアクティビティ中に報告されていたはずです。この結果、次の例に示すように、CloudWatch アラームが `ALARM` 状態になり、Incident Detection and Response が連動します。

Metric Math 関数を使用すると、実際のデータポイントがアクティビティ中は 0 に置き換えられ、アラームは `OK` 状態のままになり、Incident Detection and Response エンゲージメントの連動が抑制されます。

チュートリアル: Metric Math 関数を削除してアラーム抑制を解除

1 回限りのアクティビティに対して CloudWatch アラームを抑制する場合は、アクティビティの完了後にアラームから Metric Math 関数を削除して、アラームの定期的なモニタリングを再開します。例えば、毎週同じ曜日と時刻にインスタンスを再起動するパッチ適用ルーチンがスケジュールされている場合など、定期的なスケジュールでアラームを抑制するには、Metric Math 関数をそのままにしておきます。

次のチュートリアルでは、Metric Math 関数を削除して CloudWatch アラームの抑制を解除する方法について説明します。

1. AWS Management Console にサインインして、CloudWatch コンソール (https://console.aws.amazon.com/cloudwatch/) を開きます。
2. [アラーム] を選択し、Metric Math 関数を追加するアラームを見つけます。
3. Metric Math セクションで、[編集] を選択します。
4. アラームから抑制を削除するには、メトリクス数式の横にある [x] ボタンを選択します。
5. メトリクスを選択して実際のメトリクスのモニタリングを再開し、[メトリクスの選択] を選択します。
6. [プレビューと作成にスキップ] を選択します。
7. アラームが想定どおりに設定されていることを検証し、[アラームを更新して変更を保存] を選択します。

Incident Detection and Response からのワークロードのオフボード

AWS Incident Detection and Response からワークロードをオフボードするには、ワークロードごとに新しいサポートケースを作成します。サポートケースを作成する際は、次の点に注意してください。

- 1 つの AWS アカウントにあるワークロードをオフボードするには、ワークロードのアカウントまたは支払者アカウントからサポートケースを作成します。
- 複数の AWS アカウントにまたがるワークロードをオフボードするには、[支払者アカウント] からサポートケースを作成します。サポートケースの本文で、オフボードするすべてのアカウント ID を記載します。

> **Important**
>
> ワークロードをオフボードするサポートケースを作成するアカウントを間違えると、ワークロードをオフロードするまでに遅延が発生したり、追加情報が要求されたりする場合があります。

ワークロードをオフボードするリクエスト

1. AWS サポート センターに移動し、[ケースの作成] を選択します。
2. [技術] を選択します。
3. [サービス] で、[Incident Detection and Response] を選択します。
4. [カテゴリ] で、[ワークロードのオフボーディング] を選択します。
5. [重要度] で、[一般的なガイダンス] を選択します。
6. この変更の [件名] を入力します。例：

   [オフボード] AWS Incident Detection and Response - *workload_name*
7. この変更の [説明] を入力します。例えば、「このリクエストは AWS インシデント検出とレスポンスにオンボードされた既存のワークロードをオフボーディングするためのものです」と入力します。リクエストには、次の情報が含まれていることを確認してください。

   - ワークロード名: ワークロードの名前。
   - アカウント ID: ID1、ID2、ID3 など。
   - オフボーディングの理由: ワークロードをオフボーディングする理由を入力します。

8. [追加の連絡先 - オプション] セクションに、このオフボーディングのリクエストに関する連絡を受け取る E メール ID を入力します。
9. [Submit] を選択します。

AWS Incident Detection and Response のモニタリングとオブザーバビリティ

AWS Incident Detection and Response は、アプリケーションレイヤーから基盤となるインフラストラクチャまで、ワークロード全体のオブザーバビリティを定義するための専門的なガイダンスを提供します。モニタリングにより、何か問題があることがわかります。オブザーバビリティは、データ収集を使用して、何が問題で、なぜそれが発生したかを知らせます。

Incident Detection and Response システムは、Amazon CloudWatch や Amazon EventBridge などのネイティブ AWS のサービスを活用してワークロードに影響を与える可能性のあるイベントを検出することで、AWS ワークロードの障害やパフォーマンスの低下をモニタリングします。モニタリングは、差し迫った障害、進行中の障害、減少中の障害、潜在的な障害、またはパフォーマンスの低下を通知します。アカウントを Incident Detection and Response にオンボードするときは、Incident Detection and Response モニタリングシステムでモニタリングするアカウント内のアラームを選択し、それらのアラームをインシデント管理中に使用されるアプリケーションとランブックに関連付けます。

Incident Detection and Response では、Amazon CloudWatch やその他の AWS のサービスを使用してオブザーバビリティソリューションを構築します。AWS Incident Detection and Response は、次の 2 つの方法でオブザーバビリティをサポートします。

- ビジネス成果メトリクス: AWS Incident Detection and Response におけるオブザーバビリティは、ワークロードまたはエンドユーザーエクスペリエンスの成果をモニターする主要なメトリクスを定義することから始まります。AWS の専門家がお客様と協力し、ワークロードの目的、ユーザーエクスペリエンスに影響を与える可能性のある主要な出力または要因を理解し、これらの主要なメトリクスの低下をキャプチャするメトリクスとアラートを定義します。例えば、モバイル通話アプリケーションの主要なビジネスメトリクスは、通話セットアップの成功率 (ユーザー通話の成功率をモニタリング) であり、ウェブサイトの主要なメトリクスはページ速度です。インシデントエンゲージメントは、ビジネス成果メトリクスに基づいてトリガーされます。
- インフラストラクチャレベルのメトリクス: この段階では、アプリケーションをサポートする基盤となる AWS のサービスとインフラストラクチャを特定し、これらのインフラストラクチャサービスのパフォーマンスを追跡するためのメトリクスとアラームを定義します。これには、Application Load Balancer インスタンスの `ApplicationLoadBalancerErrorCount` などのメトリクスが含まれる場合があります。これは、ワークロードがオンボーディングされ、モニタリングがセットアップされた後に開始されます。

AWS Incident Detection and Response のオブザーバビリティの実装

オブザーバビリティは継続的なプロセスで、1 つの演習や時間枠では完了しない可能性があるため、AWS Incident Detection and Response では、次の 2 つのフェーズでオブザーバビリティを実装します。

- オンボーディングフェーズ: オンボーディング中のオブザーバビリティは、アプリケーションのビジネス成果が損なわれたときにそれを検出することに重点を置いています。このため、オンボーディングフェーズのオブザーバビリティは、アプリケーションレイヤーで主要なビジネス成果メトリクスを定義して、ワークロードの中断を AWS に通知することに重点を置いています。これにより、AWS はこのような中断に迅速に対応でき、復旧に役立ちます。
- オンボーディング後フェーズ: AWS Incident Detection and Response には、インフラストラクチャレベルのメトリクスの定義、メトリクスの調整、お客様の成熟度に応じたトレースとログの設定など、オブザーバビリティのためのプロアクティブサービスが多数用意されています。これらのサービスの実装には数か月かかる場合があり、複数のチームが関与する可能性があります。AWS Incident Detection and Response では、オブザーバビリティの設定に関するガイダンスを提供され、お客様はワークロード環境に必要な変更を実装する必要があります。オブザーバビリティ機能の実装に関する実践的なサポートが必要な場合は、テクニカルアカウントマネージャー (TAM) にリクエストしてください。

Incident Detection and Response によるインシデント管理

AWS Incident Detection and Response では、指定された Incident Manager のチームが提供する 24 時間 365 日のプロアクティブモニタリングとインシデント管理を利用できます。次の図は、アプリケーションアラームがインシデントをトリガーする際の標準インシデント管理プロセスの概要を示しています。アラーム生成、AWS Incident Manager エンゲージメント、インシデント解決、インシデント後レビューなどが含まれています。

1. アラーム生成: ワークロードでトリガーされたアラームは、Amazon EventBridge を介して AWS Incident Detection and Response にプッシュされます。AWS Incident Detection and Response は、アラームに関連付けられたランブックを自動的にプルし、Incident Manager に通知します。AWS Incident Detection and Response がモニタリングするアラームによって検出されない重大なインシデントがワークロードで発生した場合は、サポートケースを作成してインシデントへの対応をリクエストできます。インシデントへの対応のリクエストの詳細については、「インシデント対応をリクエストする」を参照してください。
2. AWS Incident Manager エンゲージメント: Incident Manager はアラームに応答し、カンファレンスコール、またはランブックで指定されているとおりにユーザーをエンゲージします。Incident Manager は、AWS のサービスの正常性を検証して、アラームがワークロードで使用される AWS のサービスの問題に関連しているかどうかを判断し、基盤となるサービスのステータスについてアドバイスします。必要に応じて、Incident Manager がユーザーに代わってケースを作成し、適切な AWS の専門家にサポートを依頼します。

   AWS Incident Detection and Response はアプリケーションに特化して AWS のサービスをモニタリングするため、AWS Incident Detection and Response は、AWS のサービスのイベントが宣言される前であっても、インシデントが AWS のサービスの問題に関連していると判断する場合があります。このシナリオでは、Incident Manager は AWS のサービスのステータスをアドバイスし、AWS のサービスのイベントインシデント管理フローをトリガーし、解決についてサービスチームにフォローアップします。提供された情報により、復旧計画や回避策を早期に実装して、AWS のサービスのイベントの影響を軽減できます。詳細については、「サービスイベントのインシデント管理」を参照してください。
3. インシデント解決: Incident Manager は、必要な AWS チーム全体でインシデントを調整し、インシデントが軽減または解決されるまで、適切な AWS の専門家と連携していることを確認します。
4. インシデント後レビュー (リクエストした場合): インシデント後、AWS Incident Detection and Response はリクエストに応じてインシデント後レビューを実行し、インシデント後レポートを

生成できます。インシデント後レポートには、問題の説明、影響、エンゲージメントしたチーム、およびインシデントを軽減または解決するために取られた回避策またはアクションが含まれます。インシデント後レポートには、インシデントの再発の可能性を減らすため、または同様のインシデントの将来の発生の管理を改善するために使用できる情報が含まれている場合があります。インシデント後レポートは根本原因分析 (RCA) ではありません。インシデント後レポートに加えて RCA をリクエストできます。インシデント後レポートの例を次のセクションに示します。

> **Important**
>
> 以下のレポートテンプレートは一例です。

```
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 case on behalf
 of the customer.
  At 2023-02-04T03:32:00 UTC, ** the customer’s SRE team, and #### 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 #### and TAM team to ensure newly created ALBs are pre-scaled to
 accommodate expected spikes in workload traffic.
```

トピック

- アプリケーションチームの AWS Support Center Consoleへのアクセス権をプロビジョニングする
- サービスイベントのインシデント管理
- インシデント対応をリクエストする
- AWS Support App in Slack で Incident Detection and Response のサポートケースを管理する

アプリケーションチームの AWS Support Center Consoleへのアクセス権をプロビジョニングする

AWS Incident Detection and Response は、インシデントのライフサイクル中に サポート ケースを通じて通信します。Incident Manager に対応するには、チームは サポート センターにアクセスできる必要があります。

アクセスのプロビジョニングの詳細については、「サポート ユーザーガイド」の「サポート センターへのアクセスの管理」を参照してください。

サービスイベントのインシデント管理

AWS Incident Detection and Response は、複数の顧客に影響を与える問題や、影響を受けた AWS リージョンまたはアベイラビリティーゾーン内でワークロードが使用している AWS のサービスの問題など、AWS のサービスの中断が広範囲の顧客に影響を与える場合に通知します。リクエストに応じて、AWS Incident Detection and Response インシデントマネージャーは電話会議ブリッジに参加して以下を行うことができます。

- 復旧計画の実装をガイドする
- 可能な回避策を伝達する
- インシデントと影響に関する情報を収集する
- お客様のために AWS内で問題を提唱およびエスカレーションする

お客様は AWS Health を通じてサービス中断通知を受け取ります。サービスの中断の影響を受けていない AWS リージョンで運用しているか、障害が発生したサービスを使用していない場合は、標準の AWS Incident Detection and Response エンゲージメントを通じて引き続きサポートを受けることができます。AWS Health の詳細については、「AWS Health とは」をご参照ください。

サービス中断時の AWS Incident Detection and Response のサポートの詳細については、次のインシデント対応ワークフロー図を参照してください。この図は、AWS チームが実行する手順の概要と、インシデント対応チームがお客様と協力してサービスの中断を特定、軽減、解決する方法を示しています。

インシデント対応をリクエストする

ワークロードで重大なインシデントが発生しても、AWS Incident Detection and Response でモニタリングしているアラームで検出されなかった場合は、サポートケースを作成してインシデント対応をリクエストできます。オンボーディング中のワークロードを含め、AWS Incident Detection and Response にサブスクライブしているワークロードに関するインシデント対応は、AWS Support Center Console、AWS サポート API、または AWS Support App in Slack を使用してリクエストできます。

次の図は、Incident Detection and Response チームにインシデント支援をリクエストした AWS のお客様のエンドツーエンドのワークフローを示しています。最初のリクエストから調査、緩和、解決までのステップが詳しく示されています。

ワークロードに影響しているアクティブなインシデントのインシデント対応をリクエストするには、サポート ケースを作成します。サポートケースを作成すると、AWS Incident Detection and Response は、ワークロードの復旧を加速するために必要な、AWS の専門家とのカンファレンスブリッジにお客様をつなぎます。

AWS Support Center Console を使用してインシデント対応をリクエストする

1. AWS Support Center Console コンソールを開いて [ケースの作成] を選択します。
2. [技術] を選択します。
3. [サービス] で、[インシデントの検出と対応] を選択します。
4. [カテゴリ] で、[アクティブインシデント] を選択します。
5. [重要度] で、[ビジネスクリティカルなシステムのダウン] を選択します。
6. このインシデントの [件名] を入力します。例:

   AWS Incident Detection and Response - アクティブなインシデント - workload_name
7. このインシデントの [問題の説明] を入力します。次の詳細情報を入力します。
   - 技術情報:

     ワークロード名

     影響を受けた AWS リソース (ARN)

- ビジネス情報:

  ビジネスへの影響の説明

  [オプション] カスタマーブリッジの詳細

8. AWS エキスパートの手配を迅速化するために、以下の詳細を提供します。
   - 影響を受けた AWS のサービス
   - その他影響を受けたサービス/サービス以外
   - 影響を受けた AWS リージョン
9. [連絡先の追加] セクションに、このインシデントに関する通信を受信する E メールアドレスを入力します。

   次の図は、[追加の連絡先] フィールドが強調表示されたコンソール画面を示しています。

10. [送信] を選択します。

    インシデント対応リクエストを送信したら、組織から E メールアドレスを追加できます。アドレスを追加するには、ケースに返信し、[連絡先の追加] セクションに E メールアドレスを追加します。

    次の図は、[返信] ボタンが強調表示された [ケースの詳細] 画面を示しています。

    次の図は、[追加の連絡先] フィールドと [送信] ボタンが強調表示されたケースを示しています。

11. AWS Incident Detection and Response は 5 分以内にケースを確認し、適切な AWS の専門家とカンファレンスブリッジに参加します。

AWS サポート API を使用してインシデント対応をリクエストする

サポートケースは、AWS サポート API を使用してプログラムで作成できます。詳細については、「AWS サポート ユーザーガイド」の「AWS サポート API について」を参照してください。

AWS Support App in Slack を使用してインシデント対応をリクエストする

AWS Support App in Slack を使用してインシデント対応をリクエストするには、次の手順を実行します。

1. AWS Support App in Slack を設定した Slack チャネルを開きます。
2. 次のコマンドを入力します。

```
/awssupport create
```

3. このインシデントの [件名] を入力します。AWS Incident Detection and Response - アクティブなインシデント - workload_name を入力します。
4. このインシデントの [問題の説明] を入力します。次の詳細情報を入力します。

   技術情報:

   影響を受けるサービス:

   影響を受けるリソース:

   影響を受けるリージョン:

   ワークロード名:

   ビジネス情報:

   ビジネスへの影響の説明:

   [オプション] カスタマーブリッジの詳細:
5. [次へ] を選択します。

6. [問題のタイプ] で、[テクニカルサポート] を選択します。
7. [サービス] で、[インシデントの検出と対応] を選択します。
8. [カテゴリ] で、[アクティブインシデント] を選択します。
9. [重要度] で、[ビジネスクリティカルなシステムのダウン] を選択します。

10. オプションで、[通知する追加の連絡先] フィールドに最大 10 件の追加の連絡先をカンマで区切って入力します。これらの追加の連絡先は、このインシデントに関する E メール連絡のコピーを受信します。

11. [Review] (レビュー) を選択します。

12. 自分にのみ表示される新しいメッセージが Slack チャネルに表示されます。ケースの詳細を確認し、[ケースを作成] を選択します。

13. ケース ID は、AWS Support App in Slack からの新しいメッセージで提供されます。
14. Incident Detection and Response は 5 分以内にケースを確認し、適切な AWS の専門家とのカンファレンスブリッジにつなげます。
15. Incident Detection and Response からの連絡が、ケースのスレッドで更新されます。

AWS Support App in Slack で Incident Detection and Response のサポートケースを管理する

AWS Support App in Slack を使用すると、Slack での サポート ケースの管理、AWS Incident Detection and Response ワークロードでの新しいアラームによって開始されたインシデントに関する通知の受信、インシデント応答リクエストの作成ができます。

AWS Support App in Slack を設定するには、「サポート ユーザーガイド」に示されている手順に従ってください。

> **Important**
>
> - ワークロードでのすべてのアラームによって開始されたインシデントの通知を Slack で受信するには、AWS Incident Detection and Response にオンボードしたワークロードのすべてのアカウントで AWS Support App in Slack を設定する必要があります。サポートケースは、ワークロードアラームが発生したアカウントで作成します。
> - インシデント時にユーザーに代わって複数の重要度の高いサポートケースを開いて、サポート リゾルバーをエンゲージできます。Slack チャネルの通知設定に一致する、インシデント中に開いたすべてのサポートケースに関する通知を Slack で受け取ります。
> - AWS Support App in Slack を通じて受信した通知は、インシデント中に AWS Incident Detection and Response で E メールまたは電話を介してエンゲージしたワークロードの初回連絡先とエスカレーション連絡先を置き換えるものではありません。

トピック

- Slack でのアラームによって開始されたインシデントの通知

- Slack でインシデント対応リクエストを作成する

Slack でのアラームによって開始されたインシデントの通知

Slack チャネルで AWS Support App in Slack を設定すると、AWS Incident Detection and Response でモニタリングしているワークロードでのアラームによって開始されたインシデントの通知を受け取ります。

次の例は、アラームによって開始されたインシデントの通知が Slack にどのように表示されるかを示しています。

通知の例

アラームによって開始されたインシデントが AWS Incident Detection and Response によって確認されると、次のような通知が Slack で生成されます。

AWS Incident Detection and Response によって追加された完全な連絡内容を表示するには、[詳細を表示] を選択します。

AWS Incident Detection and Response からの以降の更新はケースのスレッドに表示されます。

[詳細を表示] を選択して、AWS Incident Detection and Response によって追加された連絡内容の全文を表示します。

Slack でインシデント対応リクエストを作成する

AWS Support App in Slack でインシデント対応リクエストを作成する手順については、「インシデント対応をリクエストする」を参照してください。

Incident Detection and Response でのレポート作成

AWS Incident Detection and Response では、サービスの設定方法、インシデントの履歴、Incident Detection and Response サービスのパフォーマンスを、それぞれ理解するのに役立つ運用データとパフォーマンスデータを提供しています。このページでは、設定データ、インシデントデータ、パフォーマンスデータなど、使用可能なデータの種類について説明します。

設定データ

- オンボーディングされたすべてのアカウント
- すべてのアプリケーションの名前
- 各アプリケーションに関連付けられたアラーム、ランブック、サポートプロファイル

インシデントデータ

- 各アプリケーションにおけるインシデントの日付、数、期間
- 特定のアラームに関連するインシデントの日付、数、期間
- インシデント後レポート

パフォーマンスデータ

- サービスレベル目標 (SLO) のパフォーマンス

必要な運用データおよびパフォーマンスデータについては、テクニカルアカウントマネージャーにお問い合わせください。

Incident Detection and Response のセキュリティと回復性

サポート でのデータ保護には、AWS 責任共有モデルが適用されます。このモデルで説明されているように、「AWS」 は、「AWS クラウド」 のすべてを実行するグローバルインフラストラクチャを保護する責任があります。お客様は、このインフラストラクチャでホストされているコンテンツに対する管理を維持する責任があります。このコンテンツには、使用される AWS のサービス のセキュリティ設定と管理タスクが含まれます。

データプライバシーの詳細については、「データプライバシーのよくある質問」を参照してください。

欧州でのデータ保護の詳細については、AWS セキュリティブログに投稿されたAWS 責任共有モデルおよび GDPR のブログを参照してください。

データ保護の目的で、AWS アカウントの認証情報を保護し、個々のユーザーアカウントを AWS Identity and Access Management (IAM) で設定することをお勧めします。この方法により、それぞれの職務を遂行するために必要なアクセス許可のみを各ユーザーに付与できます。また、次の方法でデータを保護することをお勧めします。

- 各アカウントで多要素認証 (MFA) を使用します。
- Secure Sockets Layer/Transport Layer Security (SSL/TLS) 証明書を使用して AWS リソースと通信します。TLS 1.2 以降が推奨されます。詳細については、「SSL/TLS 証明書とは何ですか?」を参照してください。
- AWS CloudTrail で API とユーザーアクティビティロギングを設定します。詳細については、AWS CloudTrail を参照してください。
- AWS のサービス 内でデフォルトである、すべてのセキュリティ管理に加え、AWS の暗号化ソリューションを使用します。
- Amazon Macie などのアドバンストマネージドセキュリティサービスを使用します。これは、Amazon S3 に保存されている個人データの検出と保護を支援します。Amazon Macie に関する情報については、「Amazon Macie」を参照してください。
- コマンドラインインターフェイスまたは API を使用して AWS にアクセスするときに FIPS 140-2 検証済みの暗号化モジュールが必要な場合は、FIPS エンドポイントを使用します。利用可能な FIPS エンドポイントの情報については、「連邦情報処理規格 (FIPS) 140-2」を参照してください。

顧客の E メールアドレスなどの機密情報やセンシティブ情報は、タグや [名前] フィールドなどの自由形式のフィールドに配置しないことを強くお勧めします。これは、コンソール、API、AWS CLI、または AWS SDK で サポート または他の AWS のサービスを使用する場合も同様です。タグまたは名前に使用する自由記入欄に入力したデータは、課金や診断ログに使用される場合があります。外部サーバーへ URL を供給する場合は、そのサーバーへのリクエストを検証するために、認証情報を URL に含めないことを強くお勧めします。

AWS Incident Detection and Response によるアカウントへのアクセス

AWS Identity and Access Management (IAM) は、AWS リソースへのアクセスを安全に管理するためのウェブサービスです。IAM を使用して、誰を認証 (サインイン) し、誰にリソースの使用を認可する (アクセス許可を付与する) かを制御します。

AWS Incident Detection and Response およびアラームデータ

デフォルトでは、Incident Detection and Response は、アカウント内のすべての CloudWatch アラームの Amazon リソースネーム (ARN) と状態を受信し、オンボーディングされたアラームが ALARM 状態に変わったときにインシデント検出と対応プロセスを開始します。Incident Detection and Response がアカウントから受け取るアラームに関する情報をカスタマイズする場合は、テクニカルアカウントマネージャーにお問い合わせください。

ドキュメント履歴

以下の表に、IDR ガイドの前回のリリース以降に行われた重要な変更を示します。

変更	説明	日付
Incident Detection and Response がサービスイベントを処理する方法に関する情報を更新しました。	「サービスイベントのインシデント管理」セクションを更新しました。 更新済みのセクション: サービスイベントのインシデント管理	2025 年 5 月 15 日
新しい機能: Incident Detection and Response との連動によるアラームの抑制	[マネージドワークロード] に、アラームを一時的またはスケジュールに従って抑制する方法に関する情報を提供する新しいセクションを追加しました。 新規セクション: Incident Detection and Response との連動によるアラームの抑制	2025 年 4 月 9 日
AWS Support Center Console を使用してインシデント対応をリクエストする手順を更新	[問題の説明] フィールドに入力する情報の詳細を追加しました。 更新済みのセクション: インシデント対応をリクエストする	2025 年 2 月 6 日
他の AWS リージョンの追加	「Incident Detection and Response が利用可能なリージョン」セクションに他の AWS リージョンが追加されました。 更新済みのセクション: Incident Detection and Response が利用可能なリージョン	2024 年 11 月 1 日
「AWS Support App in Slack で Incident Detection and Response のサポートケースを管理する」ページの更新	「インシデント管理」ページを移動して、テキストを改訂し、スクリーンショットを置き換えました。	2024 年 10 月 10 日

変更	説明	日付
	更新済みのセクション: AWS Support App in Slack で Incident Detection and Response のサポートケースを管理する	
AWS Support App in Slack の新しいページを追加 AWS Incident Detection and Response によるインシデント管理を更新	AWS Support App in Slack の新しいページを追加 AWS Incident Detection and Response によるインシデント管理を更新し、新しいセクション「AWS Support App in Slack を使用してインシデント対応をリクエストする」を追加しました。	2024 年 9 月 10 日
アカウントのサブスクリプションの更新	「アカウントのサブスクリプション」セクションを更新し、アカウントのサブスクライブをリクエストしたときにサポートケースを開く場所の詳細を追加しました。 更新済みのセクション: ワークロードを Incident Detection and Response にサブスクライブする	2024 年 6 月 12 日
サービスイベントのインシデント後レポートが利用可能	「サービスイベントのインシデント管理」セクションを更新し、サービスイベントのインシデント後レポートに関する情報を追加しました。 更新済みのセクション: サービスイベントのインシデント管理	2024 年 5 月 8 日
新しいセクションを追加: ワークロードのオフボード	「ワークロードのオフボード」セクションを「開始方法」に追加し、ワークロードのオフボードに関する情報を追加しました。 詳細については、「Incident Detection and Response からのワークロードのオフボード」を参照してください。	2024 年 3 月 28 日

変更	説明	日付
アカウントのサブスクリプションの更新	「アカウントのサブスクリプション」セクションを更新し、ワークロードのオフボードに関する情報を追加しました。 詳細については、「アカウントのサブスクリプション」を参照してください。	2024 年 3 月 28 日
テストの更新	「テスト」セクションを更新し、オンボーディングプロセスの最後の手順として障害対応テストに関する情報を追加しました。 更新済みのセクション: Incident Detection and Response でオンボードしたワークロードをテストする	2024 年 2 月 29 日
AWS Incident Detection and Response とはの更新	「AWS Incident Detection and Response とは」セクションを更新しました。 更新済みのセクション: AWS Incident Detection and Response とは	2024 年 2 月 19 日
アンケートセクションの更新	ワークロードオンボーディングのアンケートを更新し、アラームの取り込みのアンケートを追加しました。セクションの名前を「オンボーディングのアンケート」から「ワークロードオンボーディングとアラームの取り込みのアンケート」に変更しました。 更新済みのセクション: Incident Detection and Response のワークロードのオンボーディングとアラームの取り込みに関するアンケート	2024 年 2 月 2 日

変更	説明	日付
AWS のサービスイベントとオンボーディング情報の更新	いくつかのセクションを更新し、オンボーディングに関する新しい情報を追加しました。 更新済みのセクション: • サービスイベントのインシデント管理 • Incident Detection and Response でのワークロードの検出 • Incident Detection and Response へのオンボーディング • ワークロードを Incident Detection and Response にサブスクライブする 新規セクション • アプリケーションチームの AWS Support Center Consoleへのアクセス権をプロビジョニングする	2024 年 1 月 31 日
関連情報セクションの追加	「アクセスプロビジョニング」に「関連情報」セクションを追加しました。 更新済みのセクション: Incident Detection and Response にアラートを取り込むためのアクセスをプロビジョニングする	2024 年 1 月 17 日
更新された手順の例	「例: Datadog と Splunk からの通知の統合」の手順 2、3、4 のプロセスを更新しました。 更新済みのセクション: 例: Datadog と Splunk からの通知を統合する	2023 年 12 月 21 日

変更	説明	日付
紹介グラフィックとテキストの更新	「Amazon EventBridge と直接統合されている APM からアラームを取り込む」のグラフィックを更新しました。 更新済みのセクション: Incident Detection and Response でインシデントに対応するためのランブックと対応計画を作成する	2023 年 12 月 21 日
ランブックテンプレートの更新	「AWS Incident Detection and Response のランブックの開発」のランブックテンプレートを更新しました。 更新済みのセクション: Incident Detection and Response でインシデントに対応するためのランブックと対応計画を作成する	2023 年 12 月 4 日
アラーム設定の更新	アラーム設定を更新し、CloudWatch のアラーム設定に関する詳細情報を含めました。 新規セクション: Incident Detection and Response でビジネスニーズに合った CloudWatch アラームを作成する 新規セクション: CloudFormation テンプレートを使用して Incident Detection and Response で CloudWatch アラームを作成する 新規セクション: Incident Detection and Response における CloudWatch アラームのユースケースの例	2023 年 9 月 28 日

変更	説明	日付
開始方法の更新	ワークロード変更リクエストの開始方法に関する情報を更新しました。 新規セクション: Incident Detection and Response でオンボードしたワークロードへの変更をリクエストする 更新済みのセクション: ワークロードを Incident Detection and Response にサブスクライブする	2023 年 9 月 5 日
「開始方法」の新しいセクション	AWS Incident Detection and Response に AWS Incident Detection and Response にアラートを取り込む アラートの取り込みを追加しました。	2023 年 6 月 30 日
元のドキュメント	AWS Incident Detection and Response の初回発行	2023 年 3 月 15 日