

JAPAN | 2024

aws SUMMIT



インシデント対応を 10倍速くする方法、教えます - PagerDuty と AWS で爆速障害対応

草間 一人

PagerDuty株式会社

プロダクトエバンジェリスト

PagerDuty

インシデント対応を 10倍速くする方法、教えます

PagerDutyとAWSで爆速障害対応

PagerDuty

Product Evangelist

Kazuto Kusama @jacopen

Kazuto Kusama

@jacopen



Product Evangelist

@PagerDuty Japan



Founder

@Cloud Native Innovators Association



Organizer

@Platform Engineering Meetup



PagerDuty Operations Cloud

インシデントをより早く・少ないリソースで解決 / 将来のインシデントを未然に防ぐ

1. 検知

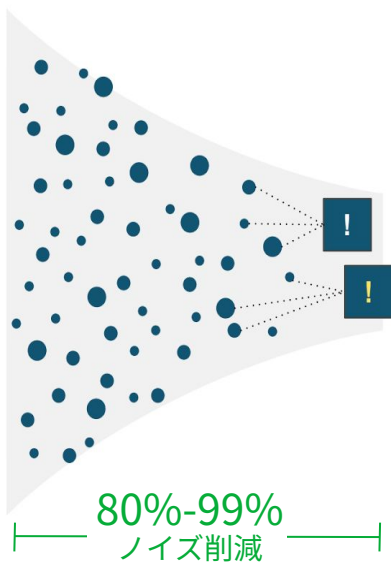
あらゆるツールから
イベントを受信



700+
Integrations

2. トリアージ

インシデントを特定
自動処理



3. 動員

最適な担当者に通知



担当者が最適な
通知方法を選択

架電、SMS、メール
Appプッシュ通知、チャット

自動エスカレーション
スケジュール管理

4. 協力/解決

迅速な解決を支援

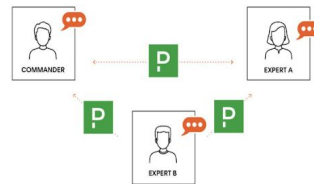
解決のヒントを提示

- 過去の類似インシデント
- 直近の構成/コード変更
...etc.

診断・修復作業の自動化



チーム内外と円滑に連携



5. 学習/予防

運用改善のための
知見を提供



対応履歴
MTTA/MTTR 分析
担当者の負荷状況
ポストモーテム

インシデントマネジメントに脚光

2月 21 インシデントマネジメント 事態収拾のための取り組みに迫る Lunch LT ★

主催：ファインディ株式会社



ハッシュタグ： #インシデントマネジメント_findy

募集内容	無料参加枠(オンライン) 無料	参加者数 382人
	公募枠 無料	先着順 (抽選終了) 1/1人

1月 16 Incident Response Meetup vol.1 【増枠】 ★
障害対応しNight

主催：Incident Response Meetup実行委員会



ハッシュタグ： #障害対応

募集内容	現地参加枠 無料	先着順 54/50人
	リモート参加枠(Zoom) 無料	先着順 454/500人

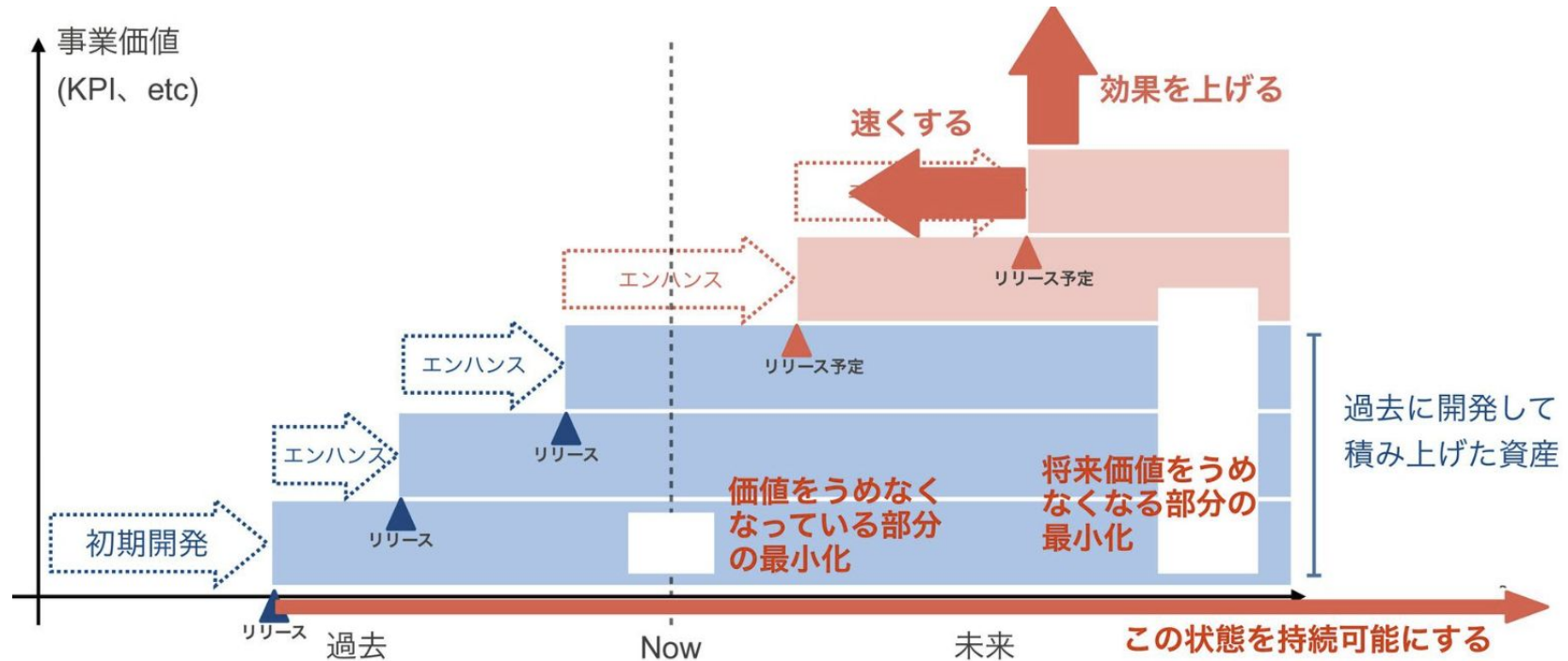
なぜインシデント対応が重要なのか

- 世の中におけるサービスの重要性が高まった
 - APIで連携し合うのはごく普通になってきた。1つのインシデントがさまざまな場所に波及する確率も高まってきた
- 構成要素の複雑化、障害対応の難化
 - クラウド、オンプレなどさまざまな選択肢
 - コンテナをはじめとしたクラウドネイティブ技術
 - マイクロサービス化の流れ
- コミュニケーション要素の増大
 - 上記の要素により組織が拡大し、コミュニケーションパスが複雑化

体系的な取り組みが必要不可欠に

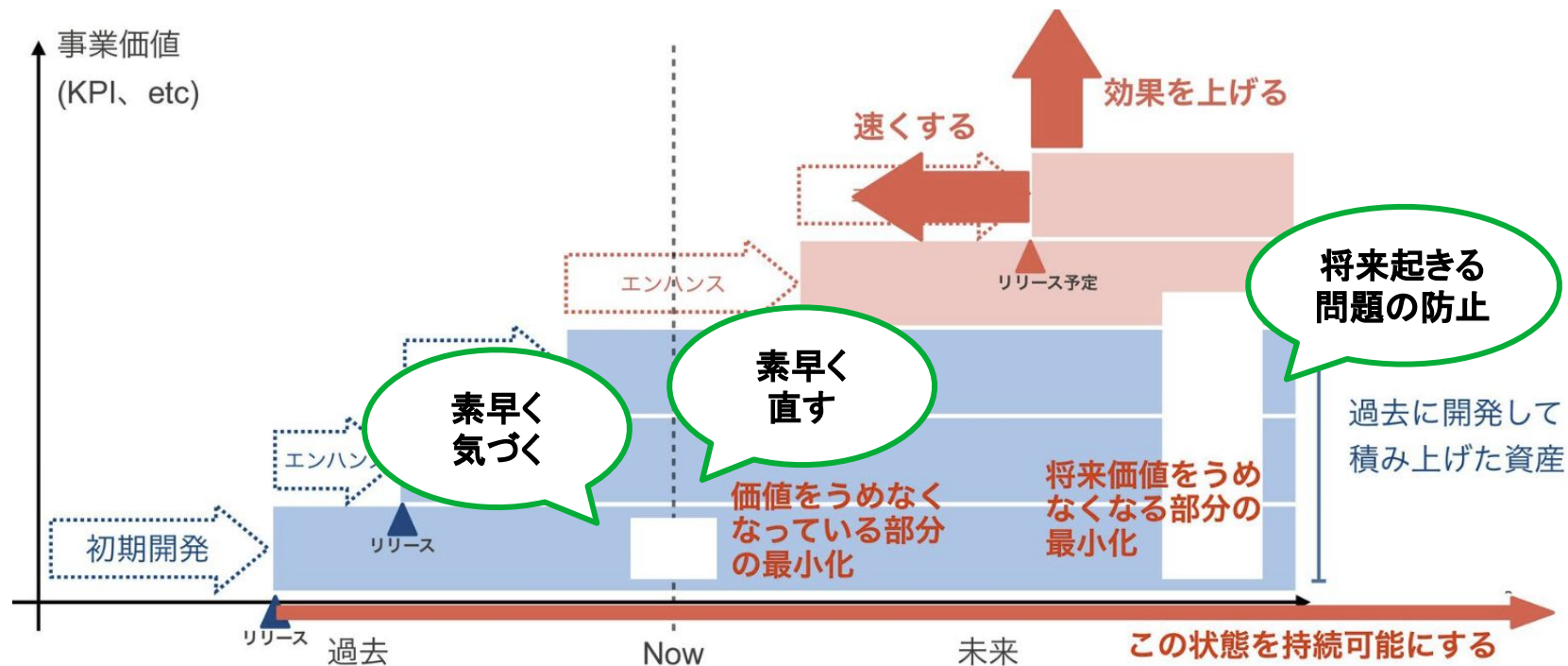
- 一人(ないしは少数)が単騎で動くことの危うさ
 - システムの複雑化にともなう対応の長期化
 - 暗黙知
 - 二次災害の危険性
 - 恒久対応や再発防止策が後回しに
- 組織として対応能力を高めていかないといけない
 - 体系だった指揮系統
 - 組織としてのノウハウの継承
 - サステナブルな組織作り

価値の総量の最大化



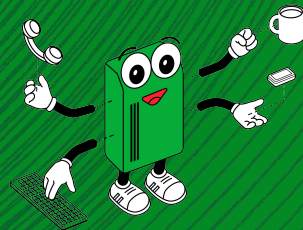
事業価値とエンジニアリング・リソース効率性とフロー効率性 / Business Value and Engineering
<https://speakerdeck.com/recruitengineers/business-value-and-engineering-2022> より引用

価値の総量の最大化



事業価値とエンジニアリング・リソース効率性とフロー効率性 / Business Value and Engineering

インシデントコマンダー



インシデントコマンドーのもと、体系的な対応をする

インシデントコマンドーは、インシデント対応の指揮者。

重大インシデントを**解決に導く**ことを目的とし、**意思決定**を行う。

日々の地位に関係なく、**重大インシデントでは最も位の高い人**



インシデントコマンドー



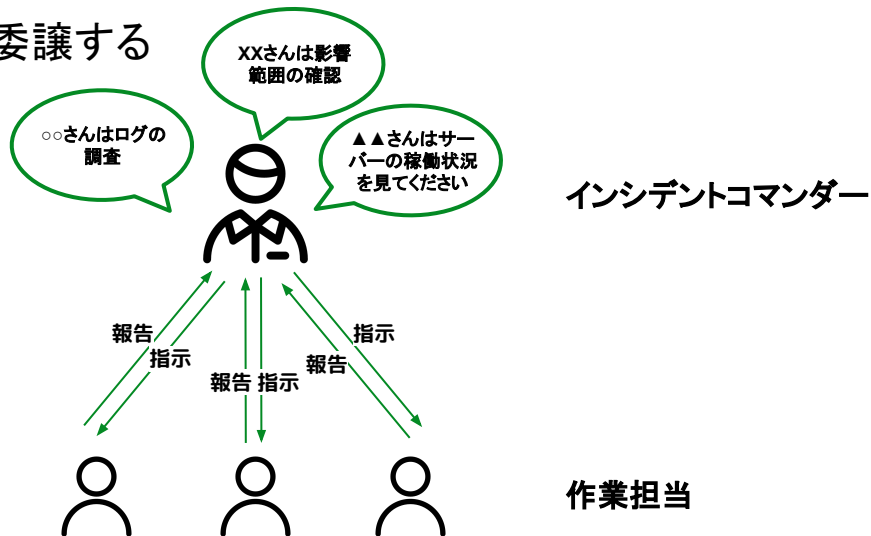
作業担当

インシデントコマンドーのもと、体系的な対応をする

インシデントコマンドーは、直接手を動かさない。

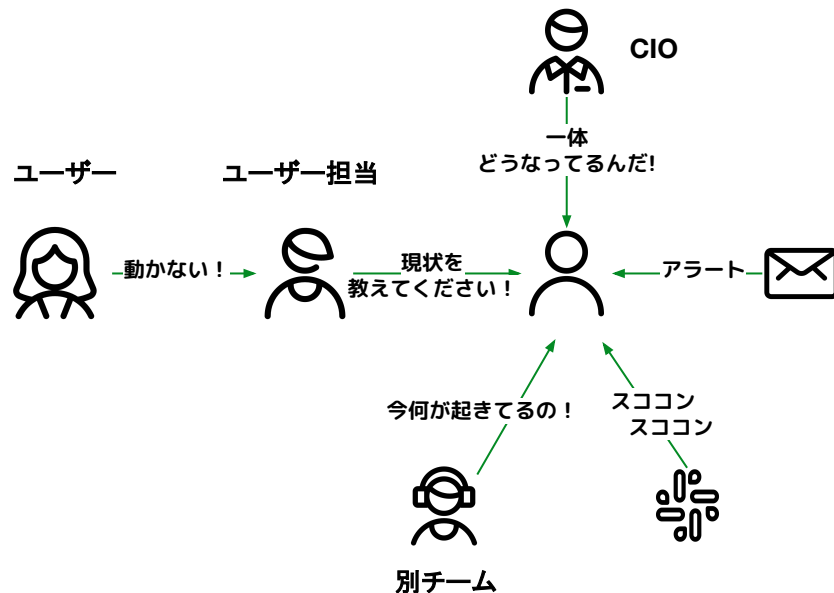
コマンドを実行したり、修正したり、メトリクスやログを調査したりしない

それらの行動は作業担当に委譲する



何故直接手を動かさないのか

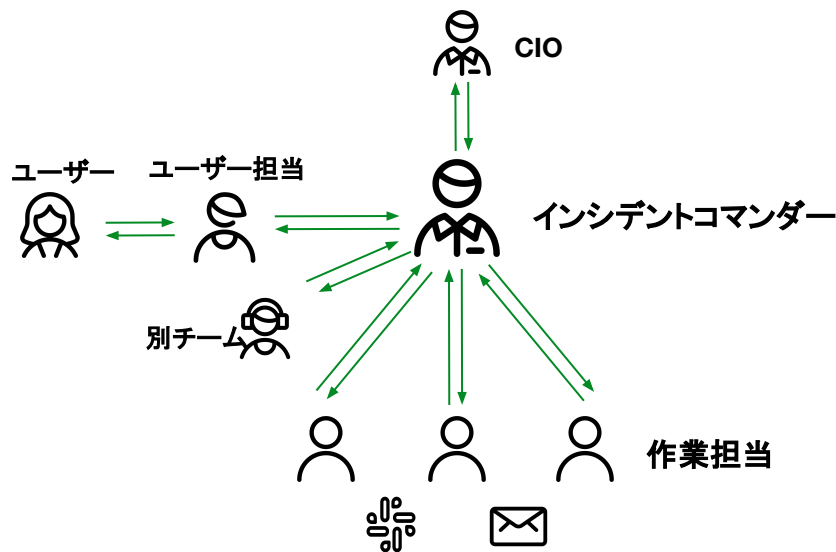
インシデントを解消していくには、たくさんの人たちと連携していく必要がある。
一人で作業をしながら、他の人の対応をするのは無謀。どちらかが犠牲になる



インシデントコマンドーのもと、体系的な対応をする

インシデントコマンドーがインシデント対応の最高責任者として、全体の交通整理を行う。作業担当には作業に専念してもらう。

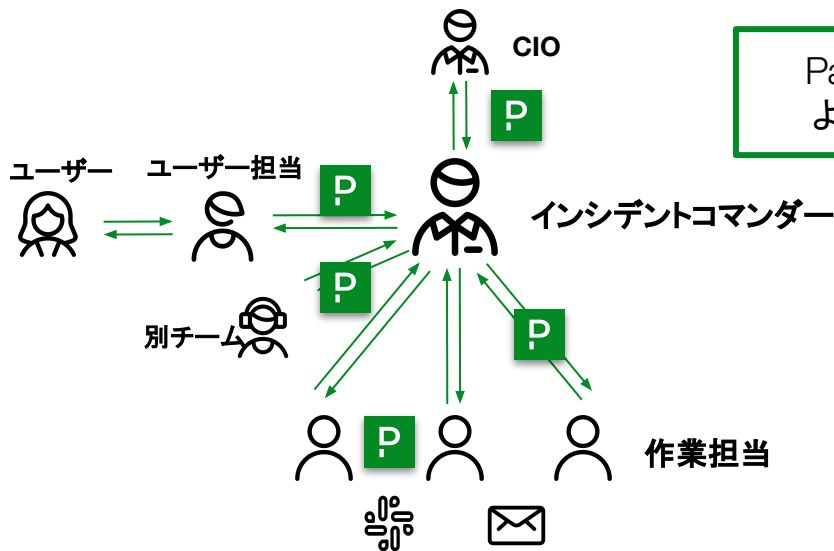
作業したくなるICも居ると思うが、そこはぐっとこらえる。それが最速への道



インシデントコマンドーのもと、体系的な対応をする

インシデントコマンドーがインシデント対応の最高責任者として、全体の交通整理を行う。作業担当には作業に専念してもらう。

作業したくなるICも居ると思うが、そこはぐっとこらえる。それが最速への道



PagerDuty Operations Cloud

インシデントをより早く・少ないリソースで解決 / 将来のインシデントを未然に防ぐ

1. 検知

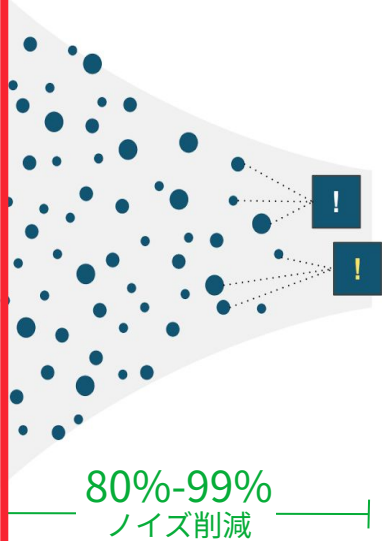
あらゆるツールから
イベントを受信



700+
Integrations

2. トリアージ

インシデントを特定
自動処理



3. 動員

最適な担当者に通知



担当者が最適な
通知方法を選択

架電、SMS、メール
Appプッシュ通知、チャット

自動エスカレーション
スケジュール管理

4. 協力/解決

迅速な解決を支援

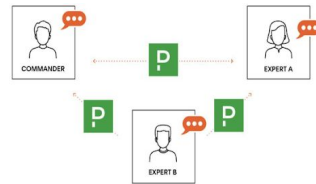
解決のヒントを提示

- 過去の類似インシデント
- 直近の構成/コード変更
...etc.

診断・修復作業の自動化



チーム内外と円滑に連携



5. 学習/予防

運用改善のための
知見を提供



対応履歴
MTTA/MTTR 分析
担当者の負荷状況
ポストモーテム



Amazon CloudWatch



PagerDuty

PagerDuty

Amazon CloudWatch Integration Guide | PagerDuty

Configure the Amazon Cloudwatch integration

Amazon CloudWatch + PagerDuty Benefits

- Amazon CloudWatch provides monitoring for AWS resources and customer-run applications. The service can collect data, gain insight, and alert users to fix problems within applications and organizations.
- Amazon CloudWatch gives system-wide visibility into resource utilization, and notifications can be set for metrics that cross specified thresholds. These notifications can be automatically sent to PagerDuty, which reliably alerts the correct on-call responder through their preferred contact methods.

Requirements

General:

- This integration expects to find in the `Message` property a nested JSON-encoded object; if this is not received, no alert will trigger. If you have any questions or need any assistance, please [contact our Support team](#).

To Configure the Integration:

- In **PagerDuty**: Managers, Admins, Global Admins and Account Owners can configure the integration.

Note

This integration is available for Amazon CloudWatch on AWS Cloud or AWS Outposts.

How it Works

- When an AWS service metric goes beyond a predefined threshold, a CloudWatch alert sends a notification to a PagerDuty endpoint, triggering an incident.
- When the AWS service metric returns to an OK state below the predefined threshold, a resolve event is sent to the same endpoint, resolving the PagerDuty incident.

PagerDuty Operations Cloud

インシデントをより早く・少ないリソースで解決 / 将来のインシデントを未然に防ぐ

1. 検知

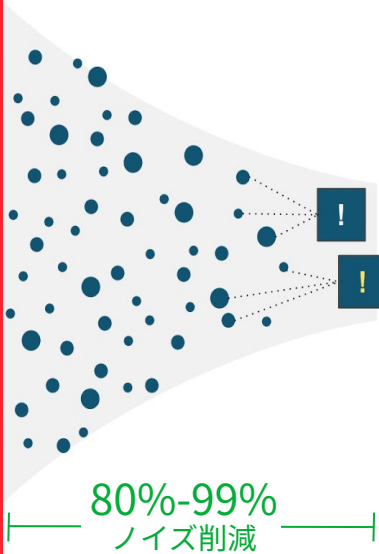
あらゆるツールから
イベントを受信



700+
Integrations

2. トリアージ

インシデントを特定
自動処理



3. 動員

最適な担当者に通知



担当者が最適な
通知方法を選択

架電、SMS、メール
Appプッシュ通知、チャット

自動エスカレーション
スケジュール管理

4. 協力/解決

迅速な解決を支援

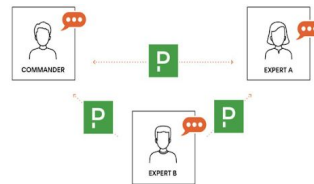
解決のヒントを提示

- 過去の類似インシデント
- 直近の構成/コード変更
...etc.

診断・修復作業の自動化



チーム内外と円滑に連携



5. 学習/予防

運用改善のための
知見を提供



対応履歴
MTTA/MTTR 分析
担当者の負荷状況
ポストモーテム

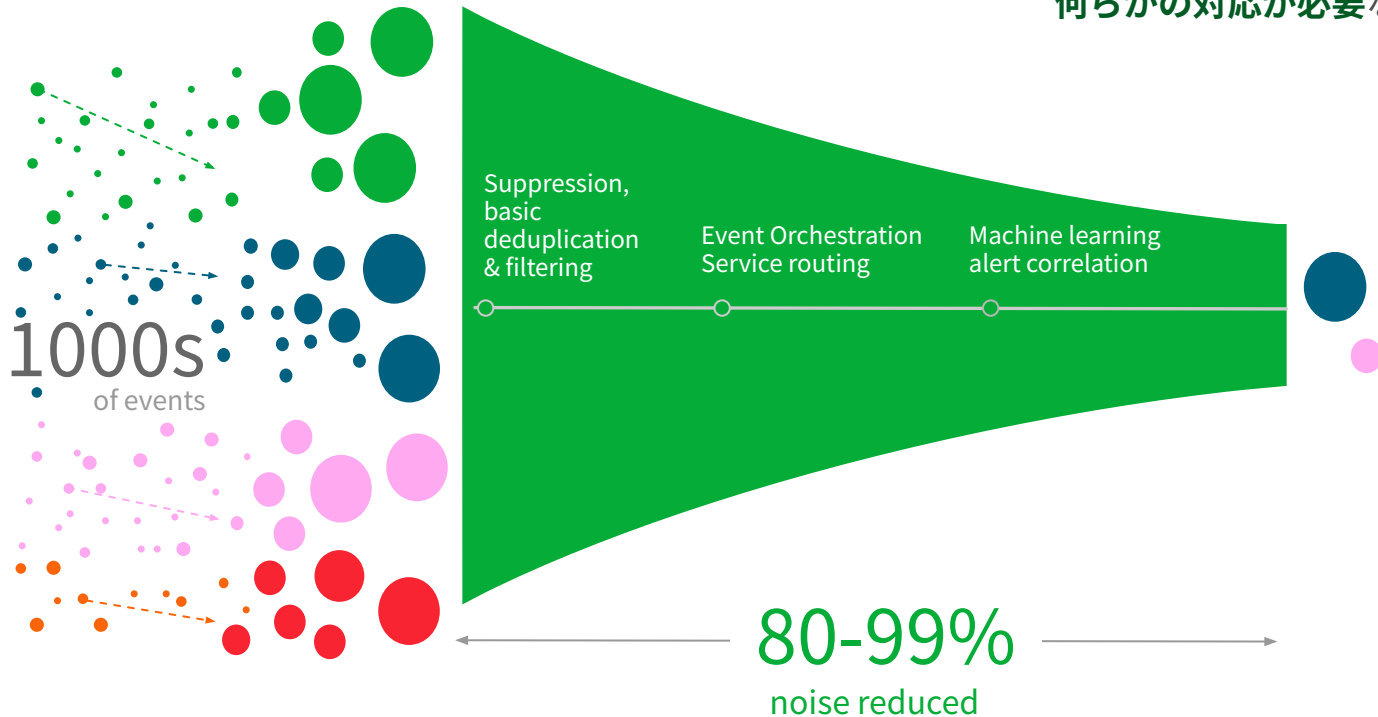
ノイズ削減: 大量のアラートから”インシデント”を特定

Event (= Alert, Signal):

監視ツール等が送られる雑多な情報

Incident:

サービスに影響を及ぼしかねない課題。
何らかの対応が必要なもの。



PagerDuty Operations Cloud

インシデントをより早く・少ないリソースで解決 / 将来のインシデントを未然に防ぐ

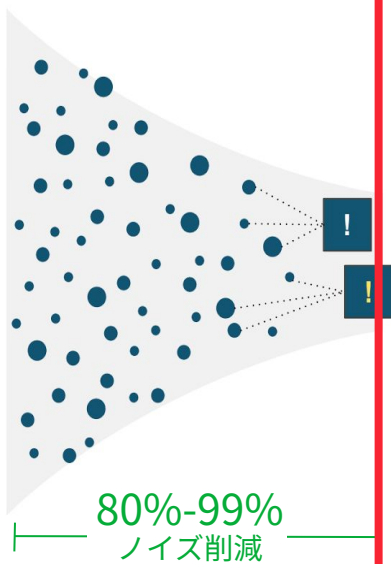
1. 検知

あらゆるツールから
イベントを受信



2. トリアージ

インシデントを特定
自動処理



3. 動員

最適な担当者に通知



担当者が最適な
通知方法を選択

架電、SMS、メール
Appプッシュ通知、チャット

自動エスカレーション
スケジュール管理

4. 協力/解決

迅速な解決を支援

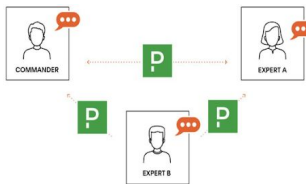
解決のヒントを提示

- 過去の類似インシデント
- 直近の構成/コード変更
...etc.

診断・修復作業の自動化



チーム内外と円滑に連携



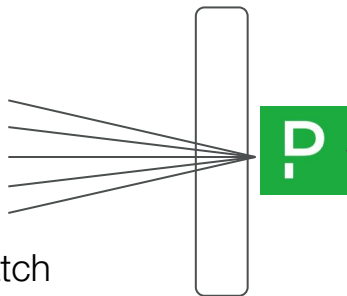
5. 学習/予防

運用改善のための
知見を提供



対応履歴
MTTA/MTTR 分析
担当者の負荷状況
ポストモーテム

オンコール



必要なアラートだけに絞り込み



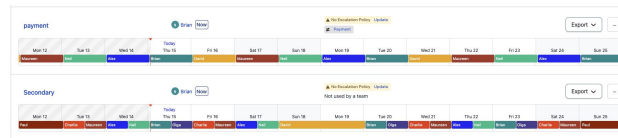
電話やSMS、プッシュ通知、Slack
など、人それぞれ適した通知

一次対応者

(応答がなければ)
二次対応者

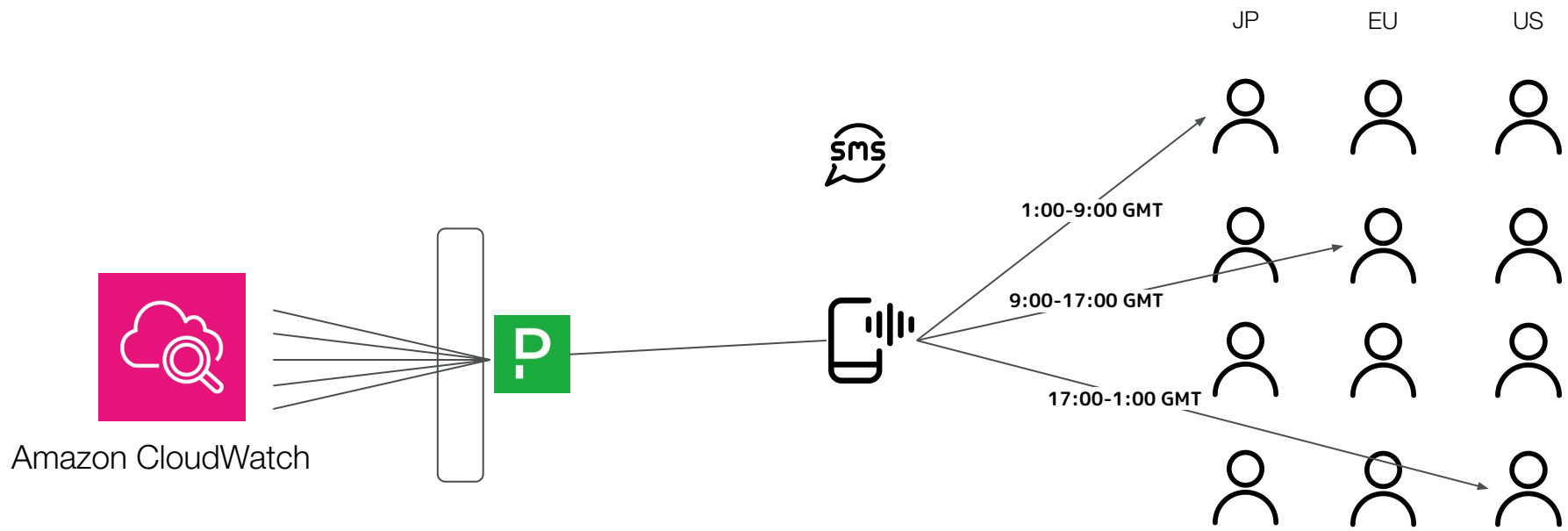


オンコールの
ローテーション



Alert Name	Time	Priority	Status	Assigned To
payment	Wed 12	High	Resolved	John
payment	Thu 13	High	Resolved	Alice
payment	Fri 14	High	Resolved	Bob
payment	Sat 15	High	Resolved	Charlie
payment	Sun 16	High	Resolved	Diana
payment	Mon 17	High	Resolved	Eve
payment	Tue 18	High	Resolved	Frank
payment	Wed 19	High	Resolved	Grace
payment	Thu 20	High	Resolved	Heidi
payment	Fri 21	High	Resolved	Ivan
payment	Sat 22	High	Resolved	Judy
payment	Sun 23	High	Resolved	Kyle
payment	Mon 24	High	Resolved	Liam
payment	Tue 25	High	Resolved	Mia
Secondary	Wed 12	Medium	Resolved	John
Secondary	Thu 13	Medium	Resolved	Alice
Secondary	Fri 14	Medium	Resolved	Bob
Secondary	Sat 15	Medium	Resolved	Charlie
Secondary	Sun 16	Medium	Resolved	Diana
Secondary	Mon 17	Medium	Resolved	Eve
Secondary	Tue 18	Medium	Resolved	Frank
Secondary	Wed 19	Medium	Resolved	Grace
Secondary	Thu 20	Medium	Resolved	Heidi
Secondary	Fri 21	Medium	Resolved	Ivan
Secondary	Sat 22	Medium	Resolved	Judy
Secondary	Sun 23	Medium	Resolved	Kyle
Secondary	Mon 24	Medium	Resolved	Liam
Secondary	Tue 25	Medium	Resolved	Mia

かしこくスケジュール



必要なアラートだけに絞り込み

電話やSMS、プッシュ通知、Slack
など、人それぞれ適した通知

グローバルな連携

PagerDuty Operations Cloud

インシデントをより早く・少ないリソースで解決 / 将来のインシデントを未然に防ぐ

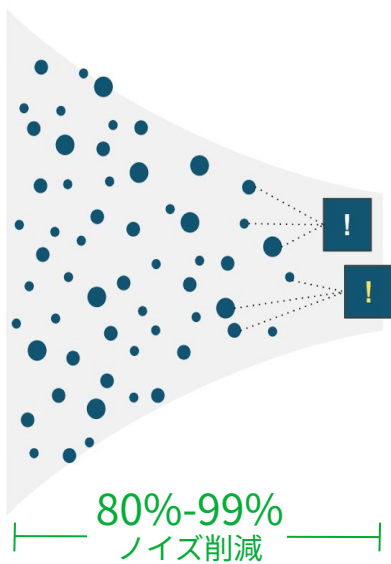
1. 検知

あらゆるツールから
イベントを受信



2. トリアージ

インシデントを特定
自動処理



3. 動員

最適な担当者に通知



担当者が最適な
通知方法を選択

架電、SMS、メール
Appプッシュ通知、チャット

自動エスカレーション
スケジュール管理

4. 協力/解決

迅速な解決を支援

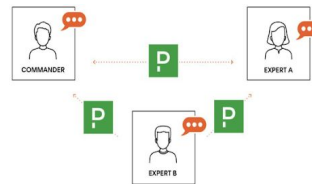
解決のヒントを提示

- 過去の類似インシデント
- 直近の構成/コード変更
...etc.

診断・修復作業の自動化



チーム内外と円滑に連携



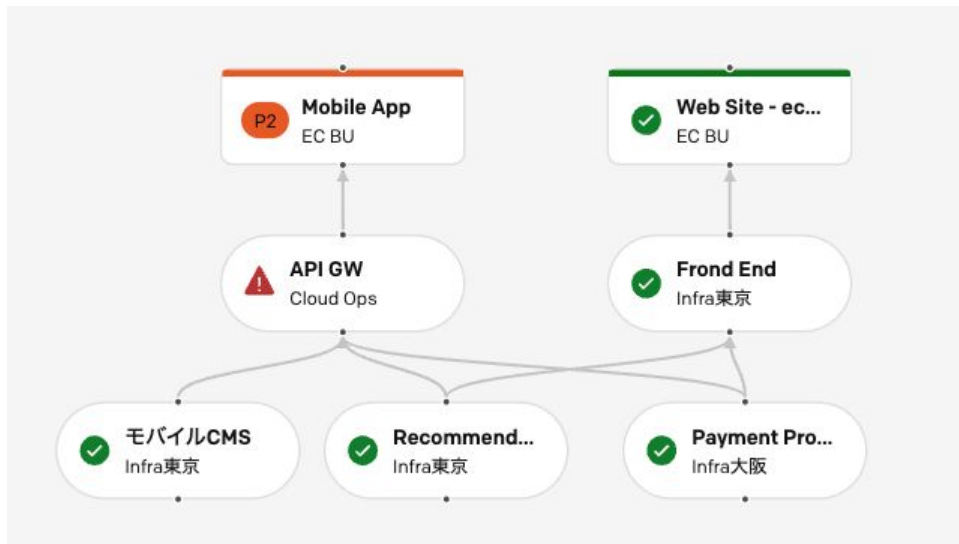
5. 学習/予防

運用改善のための
知見を提供



対応履歴
MTTA/MTTR 分析
担当者の負荷状況
ポストモーテム

+ PagerDuty だと



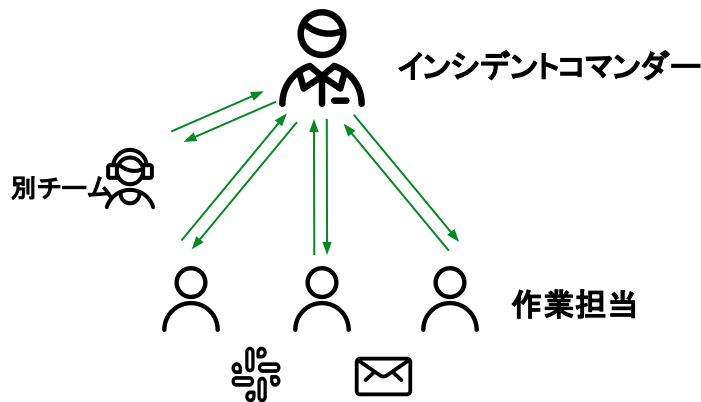
Service Graph機能で影響範囲の可視化

影響範囲の把握

インシデントが他のサービスに影響を及ぼしている可能性もある。

その場合、影響が起きているサービスとも連携しながら対応を行う必要がある。

インシデントコマンダーが状況を取りまとめて、必要に応じて外部と連携する



War room

インシデント発生時に迅速な意思決定を行っていくために関係者が招集される部屋を作る。物理的な部屋がある場合はホワイトボードとマーカー、スクリーン。加えて会議ブリッジやチャットツールのWar roomが作られることもある



+ PagerDuty だと

The screenshot shows a PagerDuty incident card with the following details:

- STATUS:** Resolved
- Open from:** Feb 9, 2024 at 3:17 PM to Feb 9, 2024 at 3:36 PM (for 19 minutes)
- URGENCY:** High
- IMPACTED SERVICE:** vCenter
- IMPACTED BUSINESS SERVICES:** Initech Cloud, Container Platform, Private IaaS
- SYNCHED WITH:** JIRA
- JIRA ISSUE:** KAN-25
- RESPONDERS:** 1
- SLACK CHANNEL:** #warroom-vcenter-90
- CONFERENCE:** https://teams.microsoft.com/l/meetup-join/19%3ameeting_MDZkYTBjMDktZDc4ZC00MDFjLTK1NGQtMGZhZjZhMDFhM...
- PRIORITY:** Default

Teams 通話 (ZoomもOK)

Slack チャンネル (TeamsもOK)

JIRAや ServiceNow と連携

**必要な環境を自動生成
手作業は少なければ少ないほど良い！**

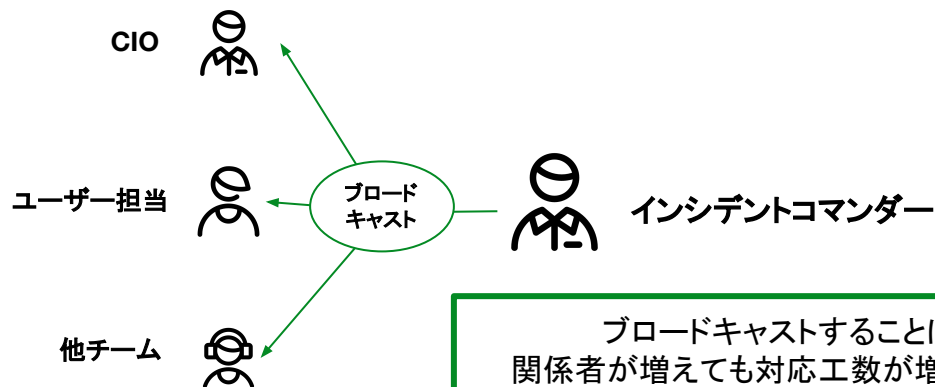
ステークホルダーとのコミュニケーション

インシデントコマンダーは、ステークホルダーに対して**適切な**コミュニケーションを取る

適切な粒度 = 詳細ではなく、

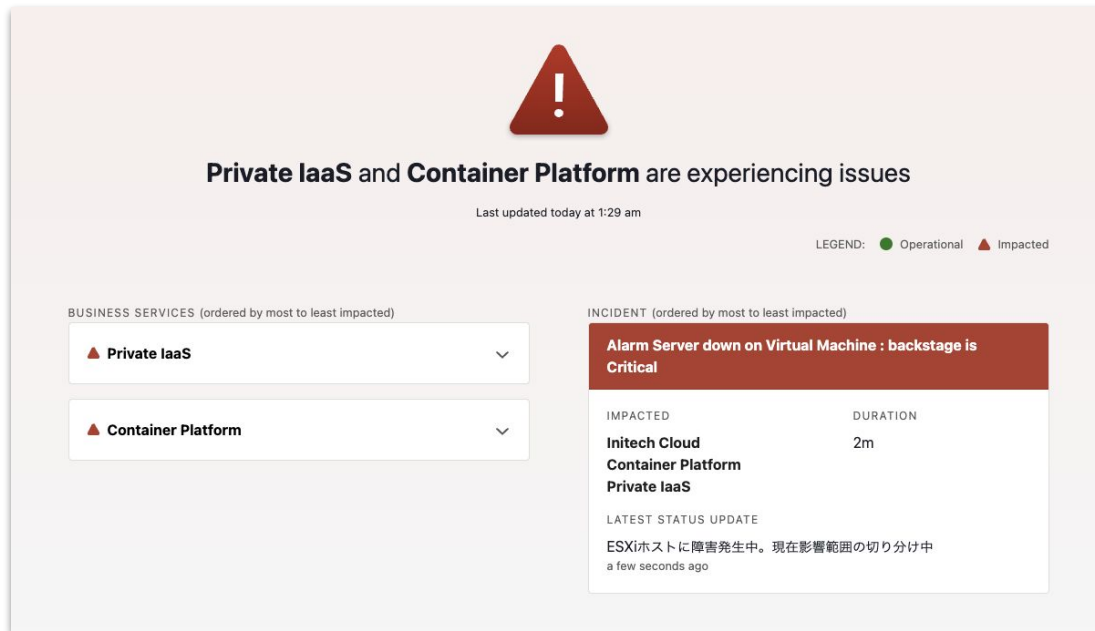
適切なタイミング = ステータス変化時 + 定期的

適切な方法 = ブロードキャスト型



ブロードキャストすることにより、
関係者が増えても対応工数が増えずに済む。
連絡漏れを防げる

+ PagerDuty だと



The screenshot displays a PagerDuty alert interface. At the top, a red warning triangle with an exclamation mark is centered. Below it, the main title reads "Private IaaS and Container Platform are experiencing issues". A timestamp indicates the alert was last updated today at 1:29 am. A legend shows a green circle for "Operational" and a red triangle for "Impacted".

Under the heading "BUSINESS SERVICES (ordered by most to least impacted)", there are two expandable cards: "Private IaaS" and "Container Platform", both marked with a red triangle icon.

Under the heading "INCIDENT (ordered by most to least impacted)", a critical incident is shown with a red header: "Alarm Server down on Virtual Machine : backstage is Critical". Below this, a table lists the impacted services and their duration:

IMPACTED	DURATION
Initech Cloud	2m
Container Platform	
Private IaaS	

Below the table, the "LATEST STATUS UPDATE" section contains the text: "ESXiホストに障害発生中。現在影響範囲の切り分け中" followed by "a few seconds ago".

ステータスアップデート機能と
ステータスページ機能でブロードキャスト

要員の管理

インシデント対応は長時間にわたることもある。インシデントコマンダーは、要員の体調面に気を配り、適切に休ませる。申告が無くても休ませる。

食事や宿泊などの補給面にも気を配ること(実際の手配は委譲したほうが良い)

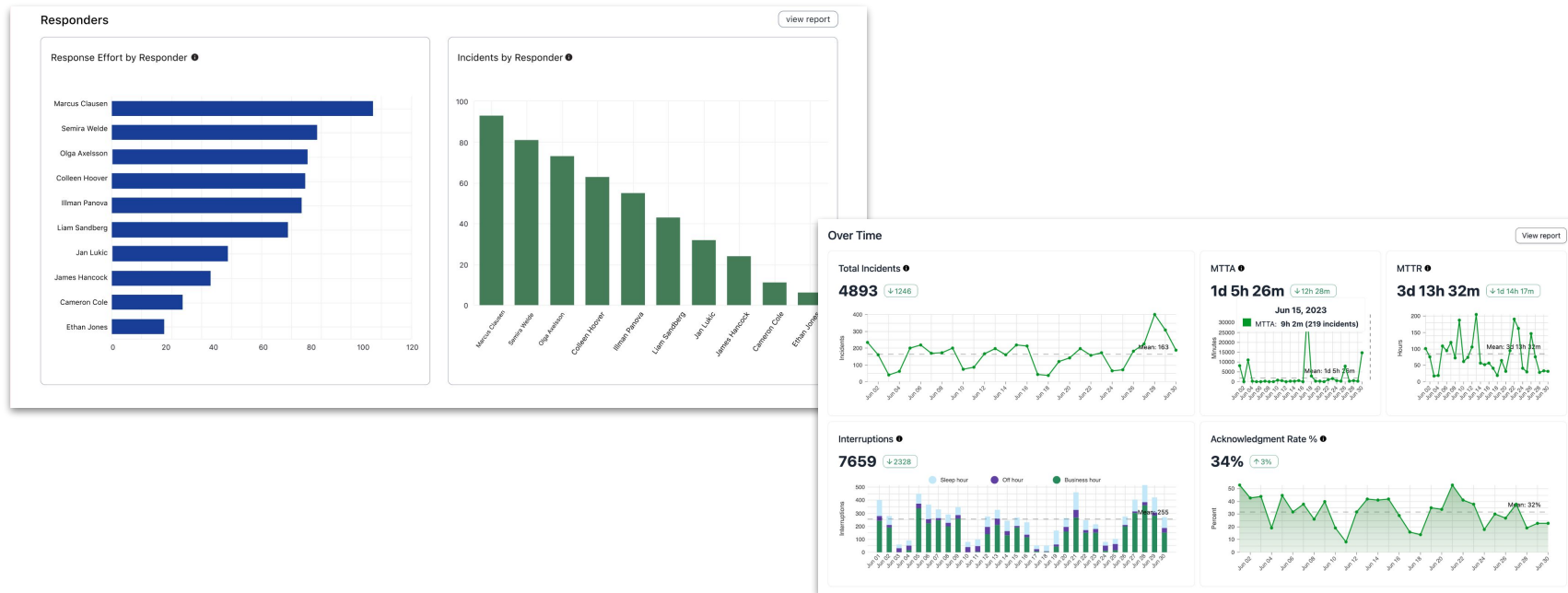


インシデントコマンダー



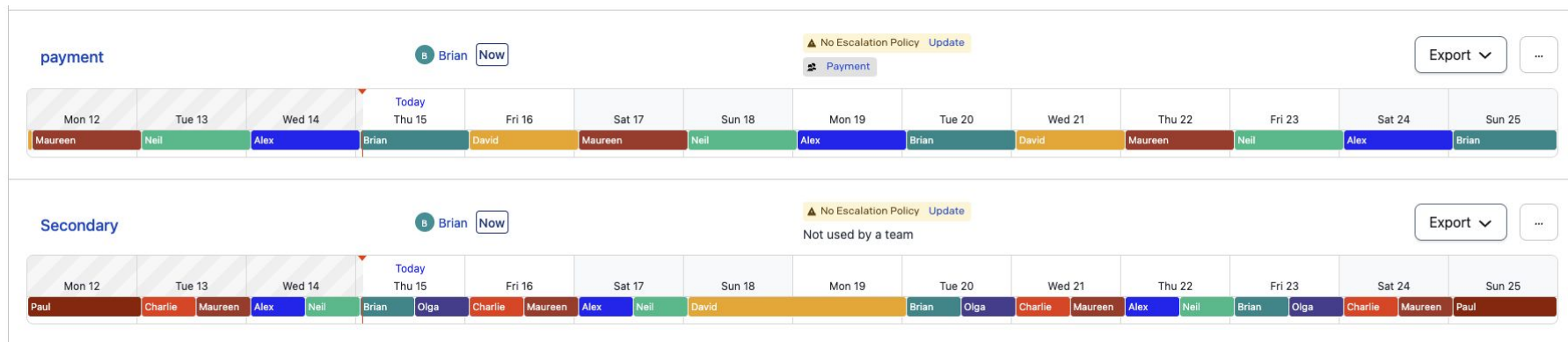
作業担当

+ PagerDuty だと



Analytics Dashboard
で状況の分析。特定の人に偏っていないかも分かる

+ PagerDuty だと



オンコールのスケジュールを管理

PagerDuty Operations Cloud

インシデントをより早く・少ないリソースで解決 / 将来のインシデントを未然に防ぐ

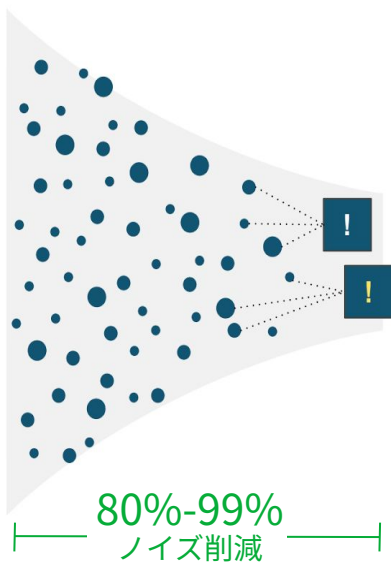
1. 検知

あらゆるツールから
イベントを受信



2. トリアージ

インシデントを特定
自動処理



3. 動員

最適な担当者に通知



担当者が最適な
通知方法を選択

架電、SMS、メール
Appプッシュ通知、チャット

自動エスカレーション
スケジュール管理

4. 協力/解決

迅速な解決を支援

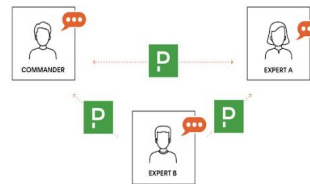
解決のヒントを提示

- 過去の類似インシデント
- 直近の構成/コード変更
...etc.

診断・修復作業の自動化



チーム内外と円滑に連携



5. 学習/予防

運用改善のための
知見を提供



対応履歴
MTTA/MTTR 分析
担当者の負荷状況
ポストモーテム

ポストモーテム

SREのプラクティスでおなじみ

- インシデントのインパクト
- 緩和や解消のために行われたアクション
- 根本原因
- インシデントの再発を避けるためのフォローアップ

きちんと纏めておくことで、**組織としての成長**に繋がる。スタンドプレーだとこのあたりの取り組みが行われないことが多い

+ PagerDuty だと

The screenshot displays the PagerDuty Timeline interface for an incident on February 9, 2022. The interface is divided into two main sections: 'Available Data' and 'Included in timeline'.

Available Data: This section lists several events related to the incident #229349, all occurring at 10:45 PM. The events include:

- Note added:** A note referring to runbook documentation for latency issues.
- Incident Urgency set to High based on Alert Severity of Critical.**
- Custom rule action Revert to Last Deploy has been attached to this incident by a service event rule.**
- Alert Product Search API: Search Response Time is High for prod - (95th percentile > 100 ms on average during the last 10m) was automatically added to this incident.**
- Priority set to "P2".**
- Assigned to Ken Barber.**

Included in timeline: This section shows the events that will be included in the final report, starting from 10:45 PM and ending at 11:50 PM. The events include:

- 10:45 PM:** Add Summary triggered through the API. Description: Product Search API: Search Response Time is High for prod - (95th percentile > 100 ms on average during the last 10m).
- 11:22 PM:** Add Summary. Restart Service initiated by Ken Barber through the website - output report.
- 11:50 PM:** Add Summary. Resolved by Ken Barber through the website.

The interface also includes a 'New Timeline Entry' button, a 'Save Timeline' button, and a timestamp: 'Last Saved Wed Feb 09 2022 23:52:30 GMT-0800'.

Postmortems

ポストモーテムの作成を支援。受信したイベント、ステータスアップデート、インシデントノート、Slackの会話などからタイムラインを作成

インシデントコマンダーになれる人はどんな人か

システムの深い技術知識は必要なし。

インシデントコマンダーの役割はインシデント対応を調整することであって、技術的な変更を行うことではない

- コミュニケーションスキル
- サービスがどのように連携しているかの理解
- 状況を判断して、行動方針に対する迅速な意思決定ができる
- フィードバックに耳を傾け、必要に応じてその場で計画を変更できる柔軟性がある
- 直近の2つの重大インシデントに、見学または対応者として関わっている
- 指揮を執り、CEOであっても通話の妨げとなる人を通話から追い出すことのできる厳格さがある

教育・育成

PagerDutyが出している、 インシデントコマンドーのガイド

(スクリーンショットは有志による翻訳)

https://ueokande.github.io/incident-response-docs-ja/training/incident_commander/

PagerDuty

REPO 6 30

Home

はじめに

オンコール

オンコールを始める

誰がオンコールになるか

アラートの原則

インシデントの前に

インシデントとはなにか？

深刻度レベル

異なる役割

通話中のエチケット

複雑なインシデント

インシデント発生中

インシデント発生中

セキュリティインシデント

インシデント収束後

インシデント収束後

ポストモーテムプロセス

ポストモーテムテンプレート

効果的なポストモーテム

トレーニング

概要

インシデントコマンドー

目的

前提条件

責務

トレーニングプロセス

卒業

started



インシデントコマンドーになりたいですか。あなたは正しい場所にたどり着きました！インシデントコマンドーはシニアメンバーである必要はなく、必要な知識があれば誰でもなることができます（もちろんインターンも含まれます）。

目的

インシデントコマンドーの目的を1文でまとめると

“ インシデントを解決に導く ”

インシデントコマンドーは重大インシデント発生中に意思決定をします。インシデントを解決するために、タスクを委譲し内容領域専門家からの意見を聞きます。日々の地位に関係なく、重大インシデントでは最も位の高い人です。コマンドーとしての意思決定は確定的なものです。

インシデントコマンドーとしての仕事は、他の背景情報や詳細情報を集約して明確な調整をするために、通話を聞きインシデントのSlackルームを見ます。インシデントコマンドーは、任意のアクションの実行や修正をしたり、グラフやログの調査をすべきではありません。それらのタスクは委譲すべきです。

インシデントコマンドーはいつでも、次のステップやバックアッププランも考慮すべきです。実行する選択肢がなく手詰まりになるのを避けて、解決に向けて前進し続けるように心がけます。

前提条件

インシデントコマンドーになる前に、次の基準を満たしている必要があります。全てを満たしていなくても、トレーニングを続けることができるので心配しないでください。

- > 優れた口頭および書面でのコミュニケーションスキルがある。
- > PagerDutyの様々なサービスがどのように連携しているかの高レベルな知識を持っている。
- > 状況を判断して、様々な戦術、戦略の効果の評価ができて、行動方針に対する迅速な意思決定ができる。
- > 専門家のフィードバックに耳を傾け、必要に応じてその場で計画を変更できる柔軟性がある。
- > 直近の2つの重大インシデントに、見学または対応者として関わっている。
- > 指揮を執り、CEOであっても通話の妨げとなる人を通話から追い出すことのできる厳格さがある。

🚩 深い技術知識は必要ありません

インシデントコマンドーはシステムの深い技術知識は必要ありません。インシデントコマンドーはインシデント対応を調整することで、技術的な変更を行うものではありません。もしあなたが開発部にいなくても、インシデントコマンドーになれると思わないでく

教育・育成

PagerDuty自身の経験に基づいた運用ガイド
PagerDuty社内で使われている
ドキュメントの編集版

- Full Service Ownership
- Incident Response
- Customer Service Operations
- DevSecOps
- Best Practices for On Call Teams
- Autoremediation
- Postmortems
- Operational Reviews
- Retrospectives
- Security Training
- Internal Stakeholder Communications
- Business Incident Response

Ops Guides

Expert-level, in-depth, and practical guides practitioners can use to improve their real-time operational chops.

Dive Deep

Real-time operations is the practice of quickly and effectively responding to digital events that require a coordinated human response. These resources are compiled from practical use, research, and experience. We update them frequently and invite your input.



Full Service Ownership Guide

Ensure the reliability of systems & services through a deeper understanding of how code functions in production.

→ [View guide](#)



Incident Response

A detailed outline of response processes for technical incidents - practiced by PagerDuty and our leading customers.

→ [View guide](#)



Customer Service Operations

Customer Service teams are a key piece of the customer experience, especially during incidents.

→ [View guide](#)



教育・育成

インシデントレスポンスについては
有志による翻訳版がある

<https://ueokande.github.io/incident-response-docs-ja/>



このドキュメントは、PagerDutyにおけるインシデント対応プロセスが載っています。重大インシデントや、新しくオンコールを始める社員の準備に利用している、PagerDutyの内部ドキュメントの一部を切り出したものです。このドキュメントではインシデントに備えることだけでなく、インシデント発生中、また収束後の対応についても説明します。オンコールを担当する人や、インシデント対応プロセスに関与する人（またきちんとしたインシデント対応プロセスを制定したい人）が読むことを想定しています。このドキュメントが何なのか、なぜ存在するかは、「このドキュメントについて」を参照してください。

どこから手を付けるべきか？

もしあなたがインシデント対応が初めてで、組織的な手順がない場合は、まず「はじめに」で何ができるかを確認してください。そして詳細な手順を、[Training Course](#)から確認することをお勧めします。

オンコールを始める

まだあなたがオンコールを体験していないのなら、それが何なのか疑問に思っても構いません。このページでは、オンコールに期待されていることと、いくつかの実例を紹介します。

- オンコールを始める - オンコールを始めるには、あなたの責務とそうでないもの
- アラートの原則 - エンジニアを呼び出すための手段やタイミングなどの原則

インシデント発生前

インシデントが発生する前に読むべきものです。実際にインシデントが発生してから読むものではないでしょう。

- インシデントとはなにか？ - インシデント対応の議論の前に、インシデントとは何なのか？
- 深刻度レベル - 深刻度レベルの分類。SEV-3とSEV-1の違いや、それぞれの対処方法
- インシデント発生時の役割 - インシデントコマンダー (Incident Commander)、記録係 (Scribe) などの、インシデント対応中の役割
- インシデント通話のエチケット - インシデント通話に参加するまでに知っておくべきエチケット

Thank you



PagerDuty



Thank you!

