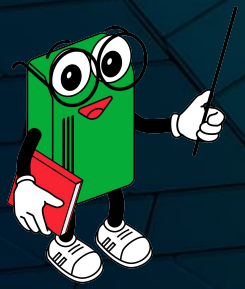
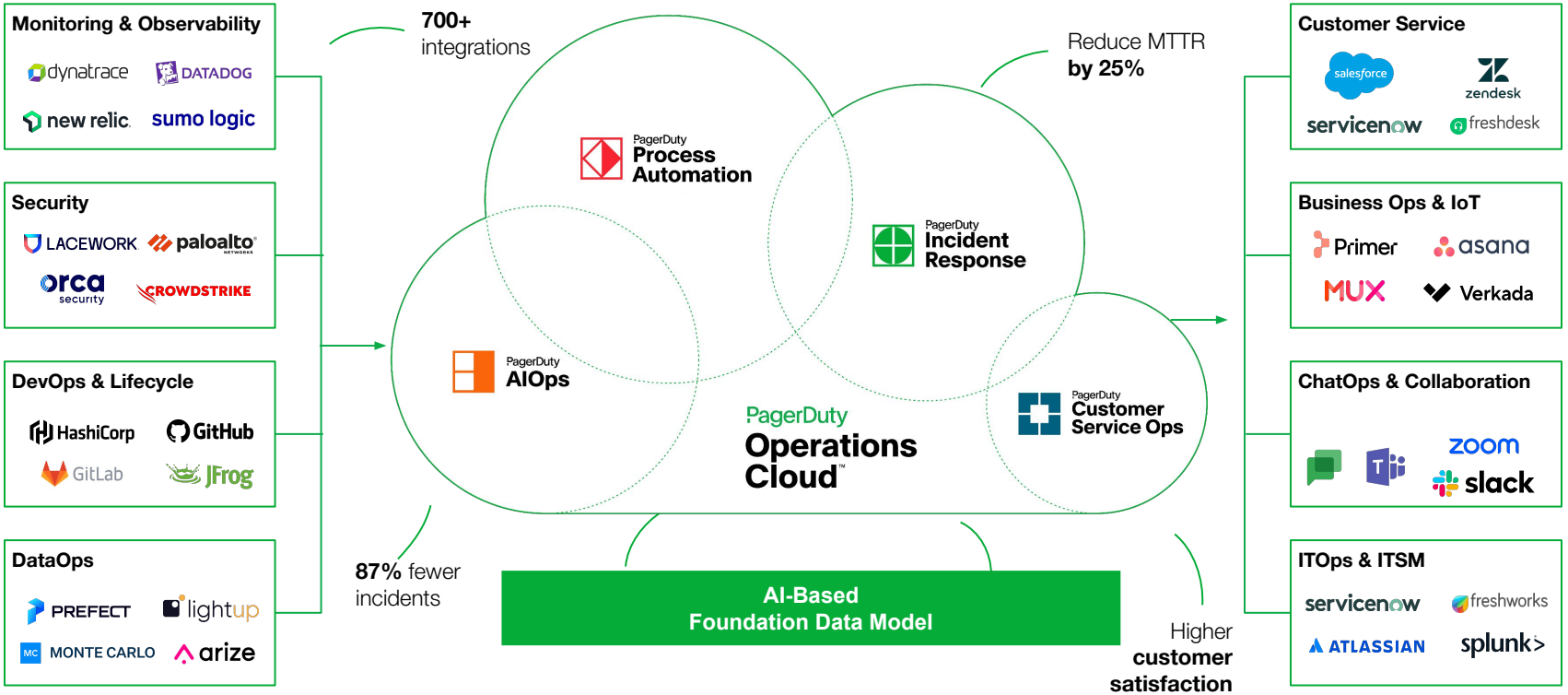
PagerDuty



AIOps and Automation Workshop

Aug 2024, Melbourne/Sydney

The PagerDuty Operations Cloud



PagerDuty

Have you ever heard?

PagerDuty is constantly going off!

I get notified for the same thing 10 times

The wrong teams are getting notified

It takes us way too long to figure out the service at fault

It takes hours to resolve a major incident

We don't have context on our incidents

We are spending too much time triggering the same action

PagerDuty

Inundated by alert noise

Fewer Incidents

Identify incidents that matter

Confused about where & what

Faster Resolution

Improve situational awareness

Wasted time on manual tasks

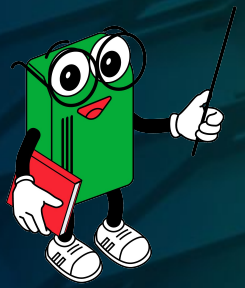
Greater Productivity

Automate manual toil

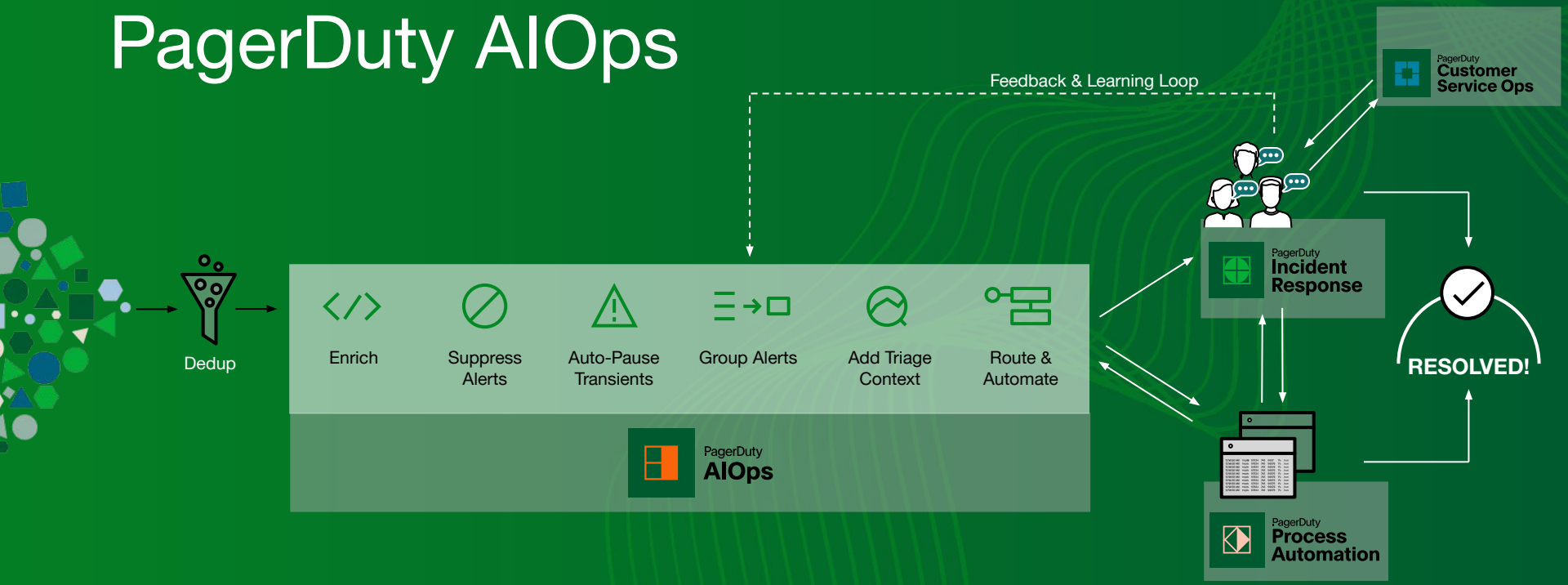
PagerDuty

Learning Objectives

Reduce Noise
Improve context
Automate manual toil



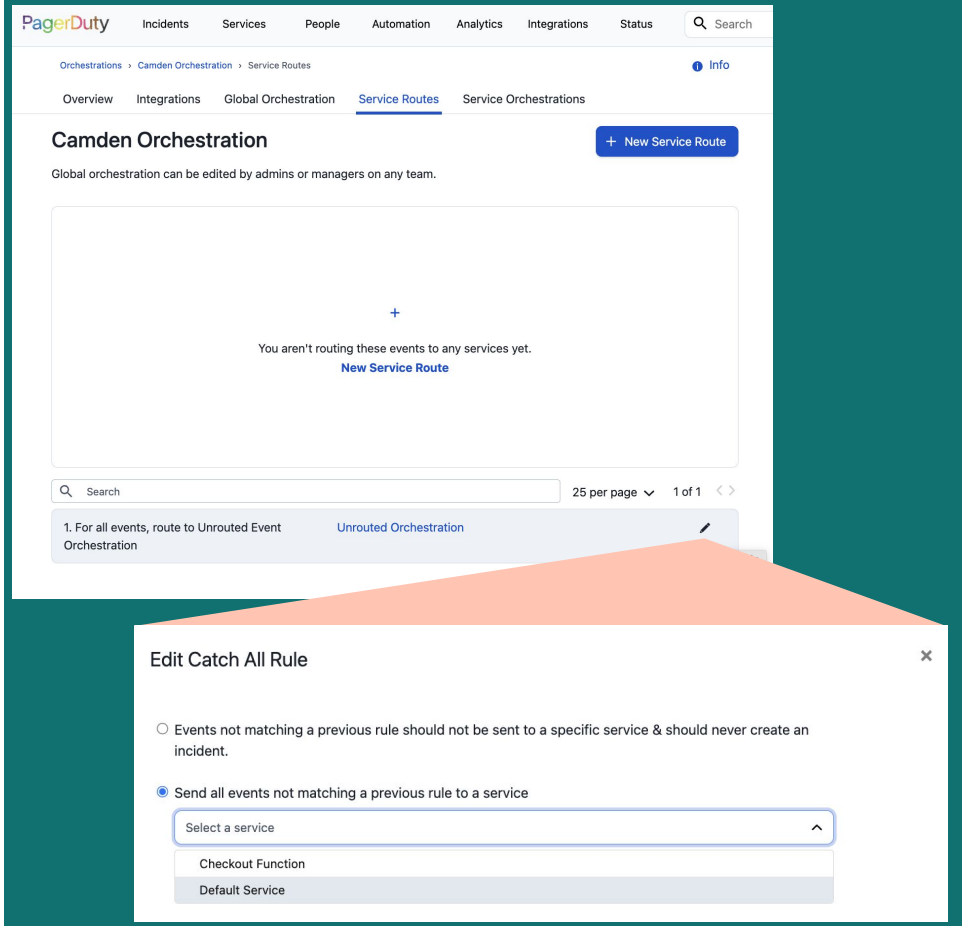
The Power of PagerDuty AIOps



Let’s Dive In!

Set Up an Event Orchestration

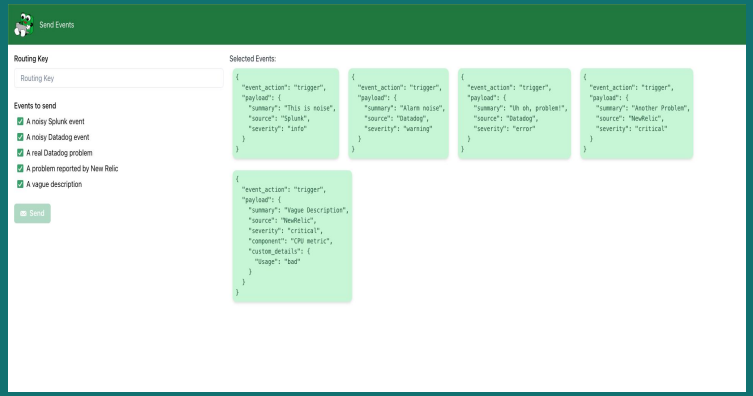
1. Menu: AIOps > Event Orchestration > +New Orchestration
2. Name your orchestration Ex: [YOURCOLOUR] Orchestration
3. Navigate to Service Routes
4. Edit Catch All Rule to direct events to **your** Technical Service



PagerDuty University

Send Some Example Incidents

1. Select your Global Integration Key
2. Go to
3. Paste your integration key
4. Check the box for each event
5. Execute!



PagerDuty University

PagerDuty

Noise Reduction

How Do We Group Alerts?

Group alerts to reduce noise and improve triage response

Time-based

Groups based on specified timeframe

Easy to configure and widely available

Content-based

Groups based on an exact match of a user defined PD-CEF field

Have complete control over alert grouping

Intelligent

Groups based on machine learning looking at alert summaries, triggered time, and past responder behavior

Works out of the box and gets smarter over time

Intelligent + Content

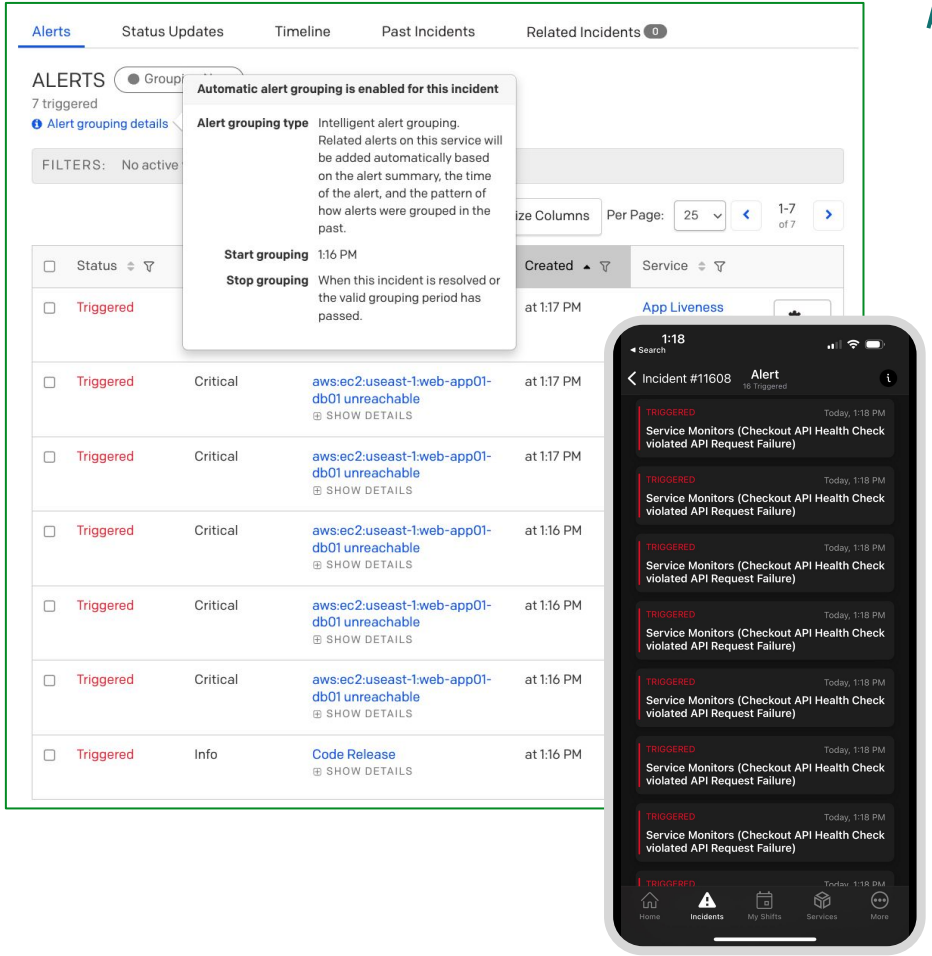
Combine both Content-Based and Intelligent Alert Grouping with a flexible time window for maximum precision and correlation control

Specify which fields and criteria need to be met before Intelligent Alert Grouping is applied

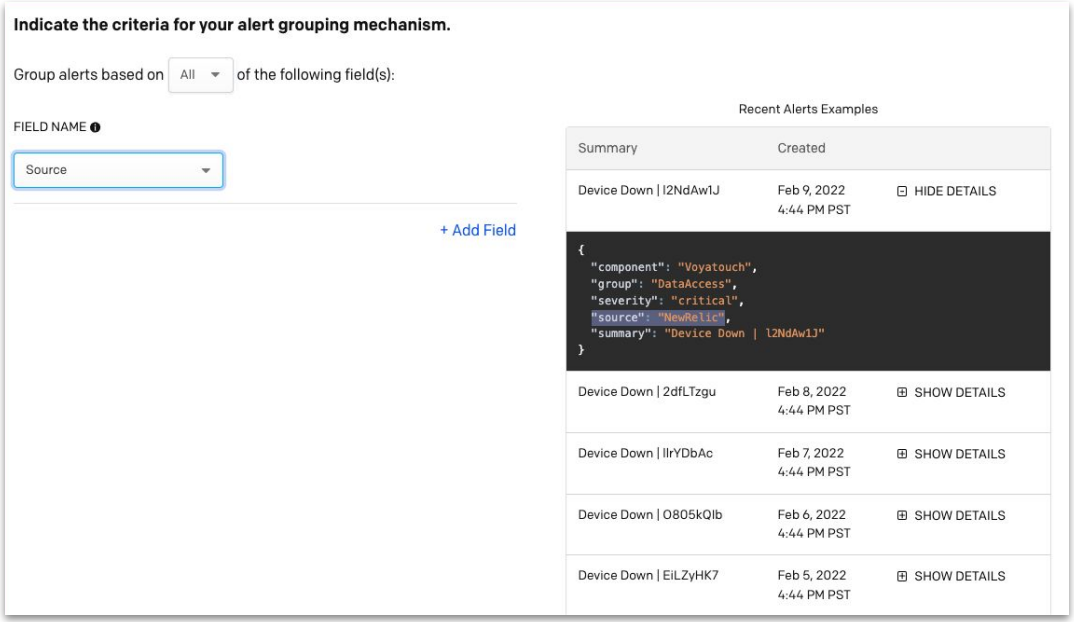
Intelligent Alert Grouping

Auto group alerts based on ML that learns over time from patterns in inbound signals *and* responder behavior.

Stop alert fatigue from alert storms



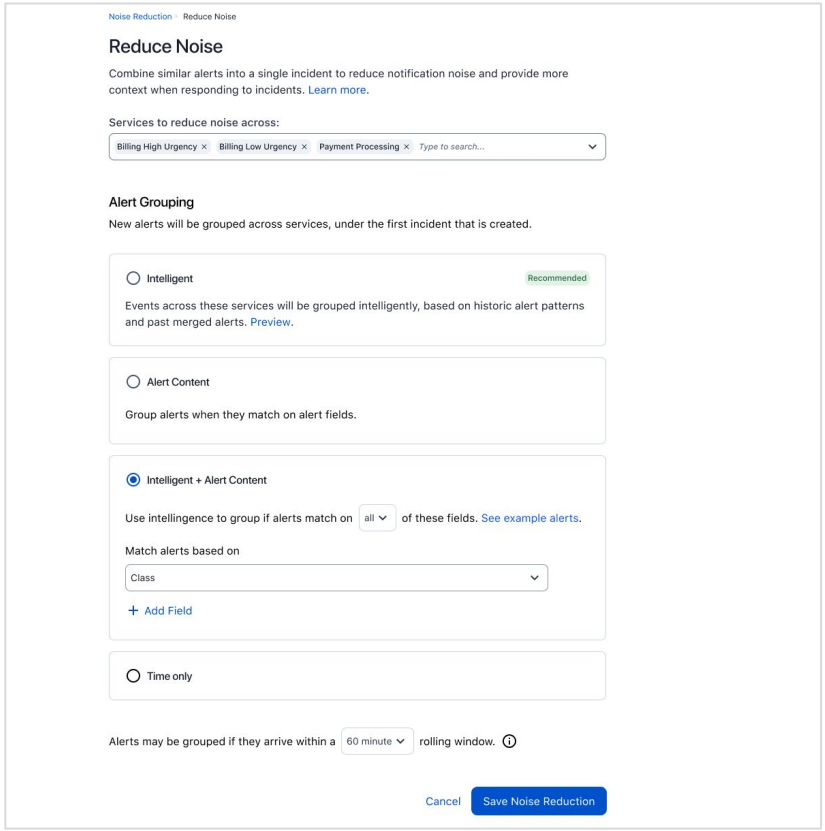
Content-Based Alert Grouping



Gain control in how alerts are grouped to reduce alert noise and fatigue

Specify service rules that group alerts by their fields and description

Intelligent + Content-Based Alert Grouping

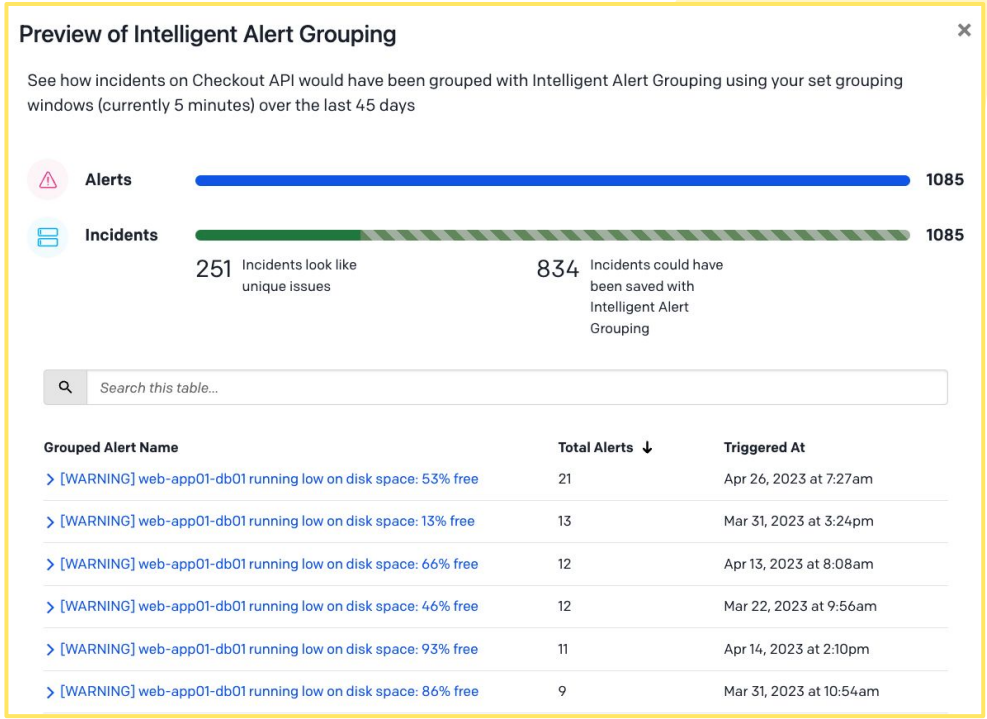


Gain control in how alerts are grouped to reduce alert noise and fatigue

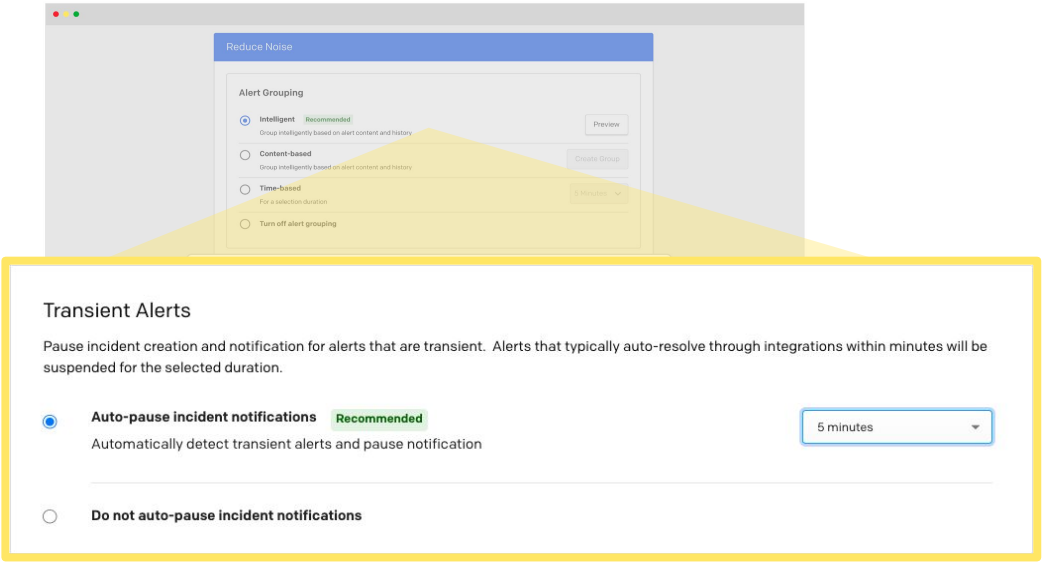
Specify service rules that group alerts by their fields and description

Preview Intelligent Alert Grouping

- View how many incidents could have been saved over 45 days
- Expand alert groups to see what would have been grouped together



Auto-Pause Incident Notifications



Automatically remove unnecessary noise from flapping alerts with the click of a button.

PagerDuty uses machine learning to detect and pause transient alerts that historically auto-resolve themselves so that responders can stay focused on work that matters.

Practice Problem: Reduce Noise

1. Resolve all open incidents
2. Navigate to Services > Service Directory
3. Select **your** Checkout Service
4. Navigate to the Settings tab
    a. Click the Edit button in the Reduce Noise section
5. Click on Time-based alert grouping and save
6. Send at least two events from the Event Sender page

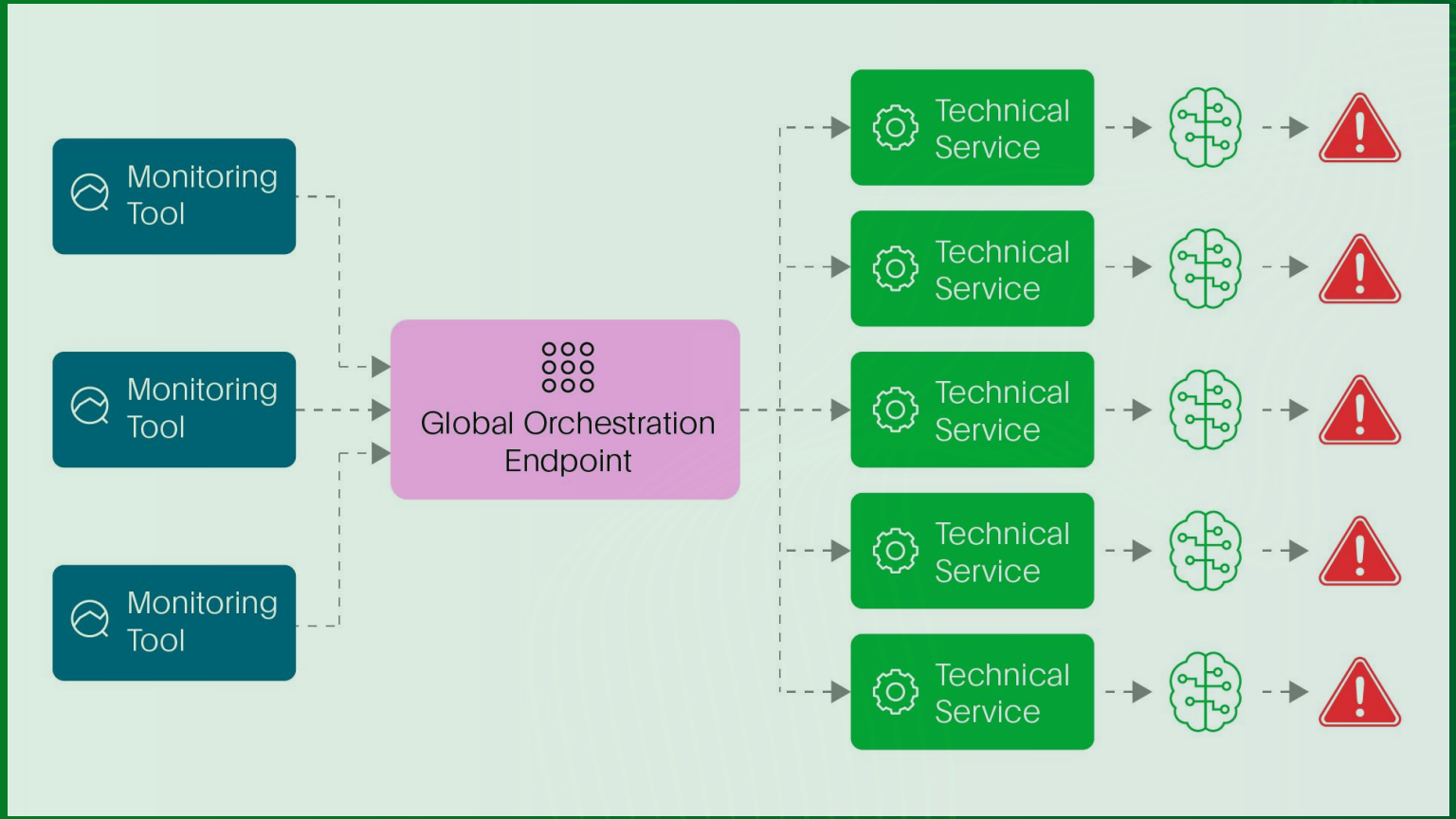
PagerDuty University

How many incidents were triggered this time?

PagerDuty

Global Event Orchestration

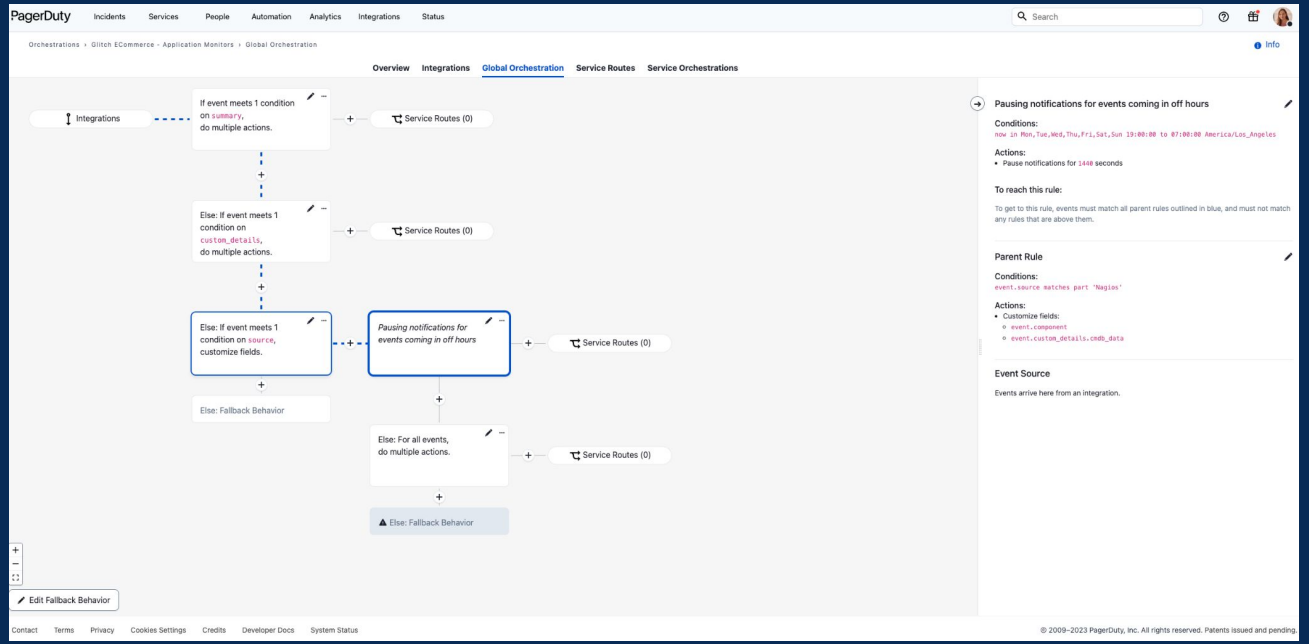
Global Event Orchestration



Global Event Orchestration

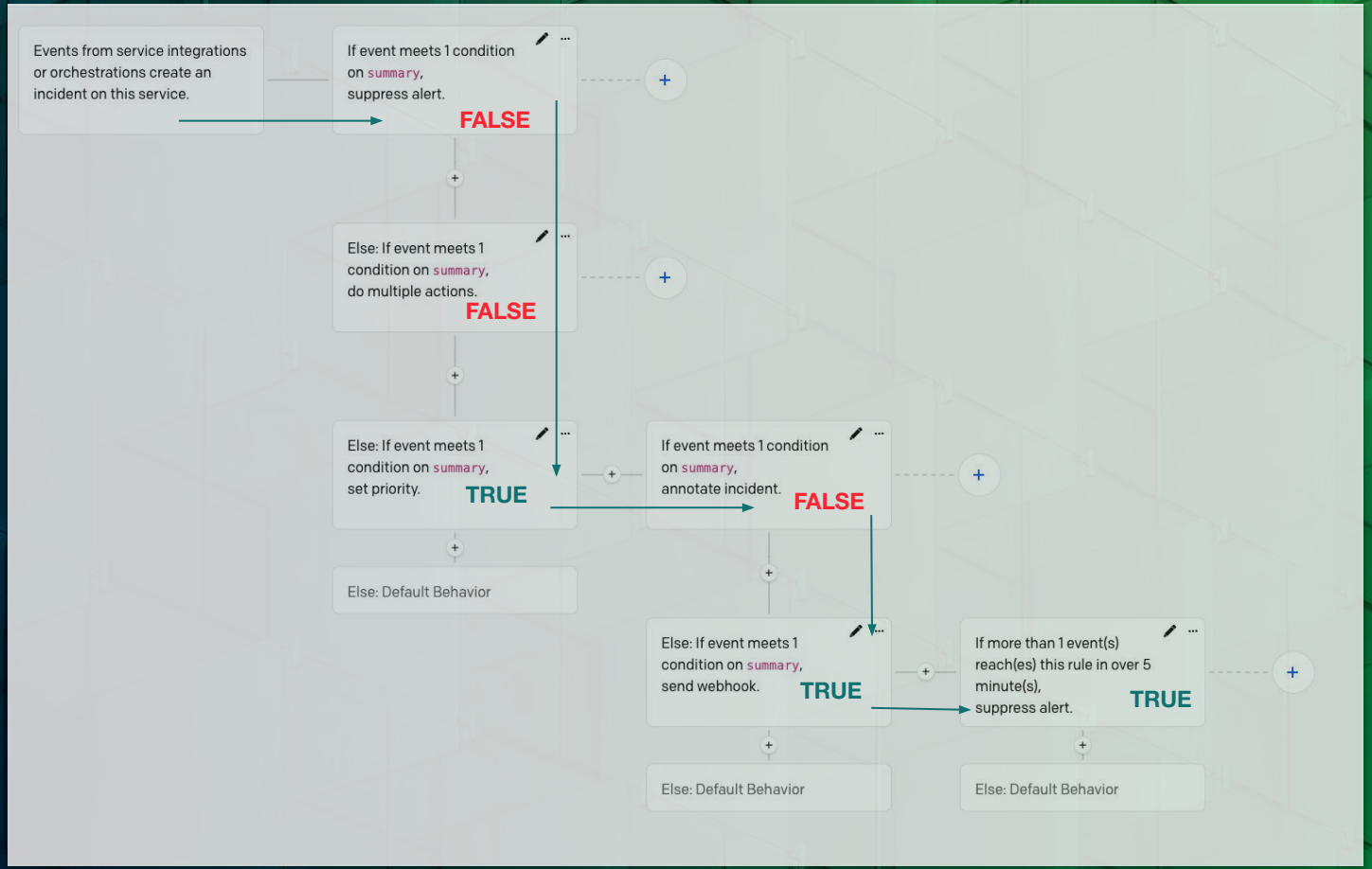
Normalize data and automate incident response for the entire organization

Route, enrich, and modify events on ingest to remove noise and automate processes across any or all services within PagerDuty



PagerDuty

Reading a Service Orchestration

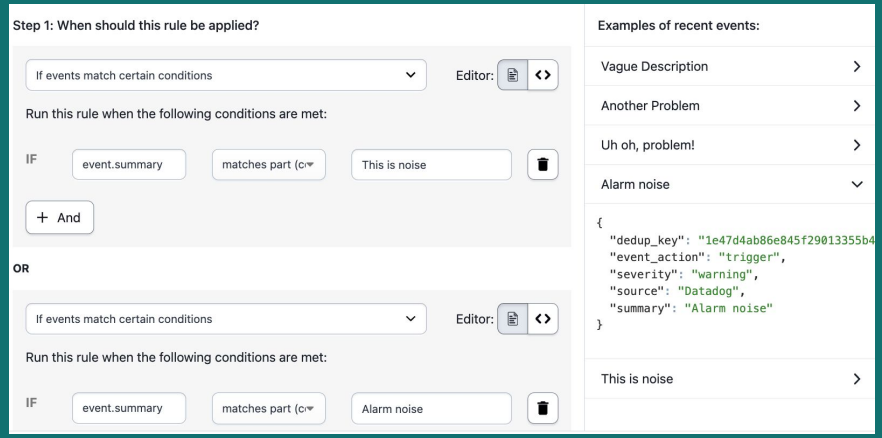


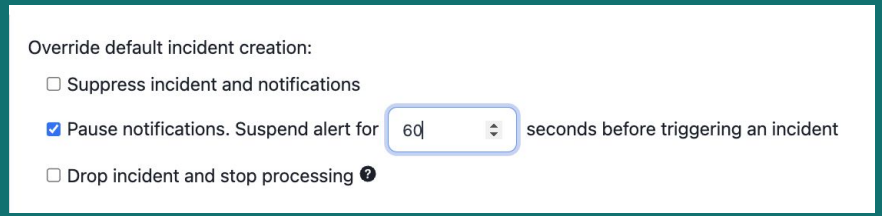
PagerDuty

Building a Global Event Orchestration

Practice Problem: **Reduce Noise**

1. AIOps > Event Orchestration
   a. Open Orchestration created earlier
2. Navigate to the Global Orchestration tab
   a. Click New Rule
3. Click on the two example events with “Noise” in the titles
   a. Click on the Summary fields
   b. Click Next
4. Select “Pause notification” for 60 seconds and Save

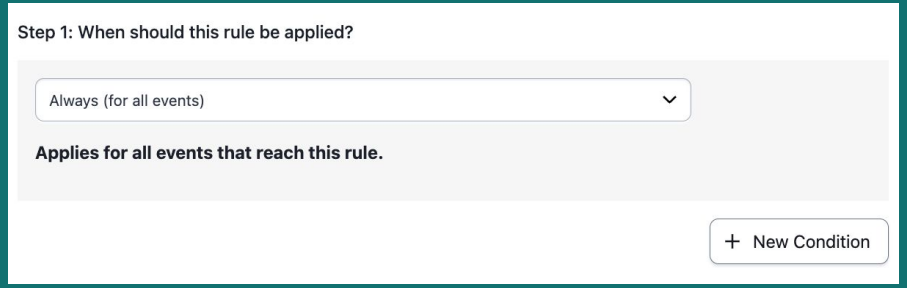


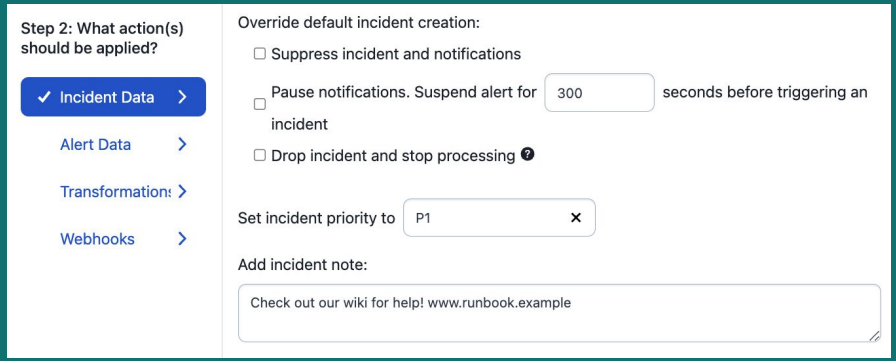


PagerDuty University

Practice Problem:
Enrich & Enhance

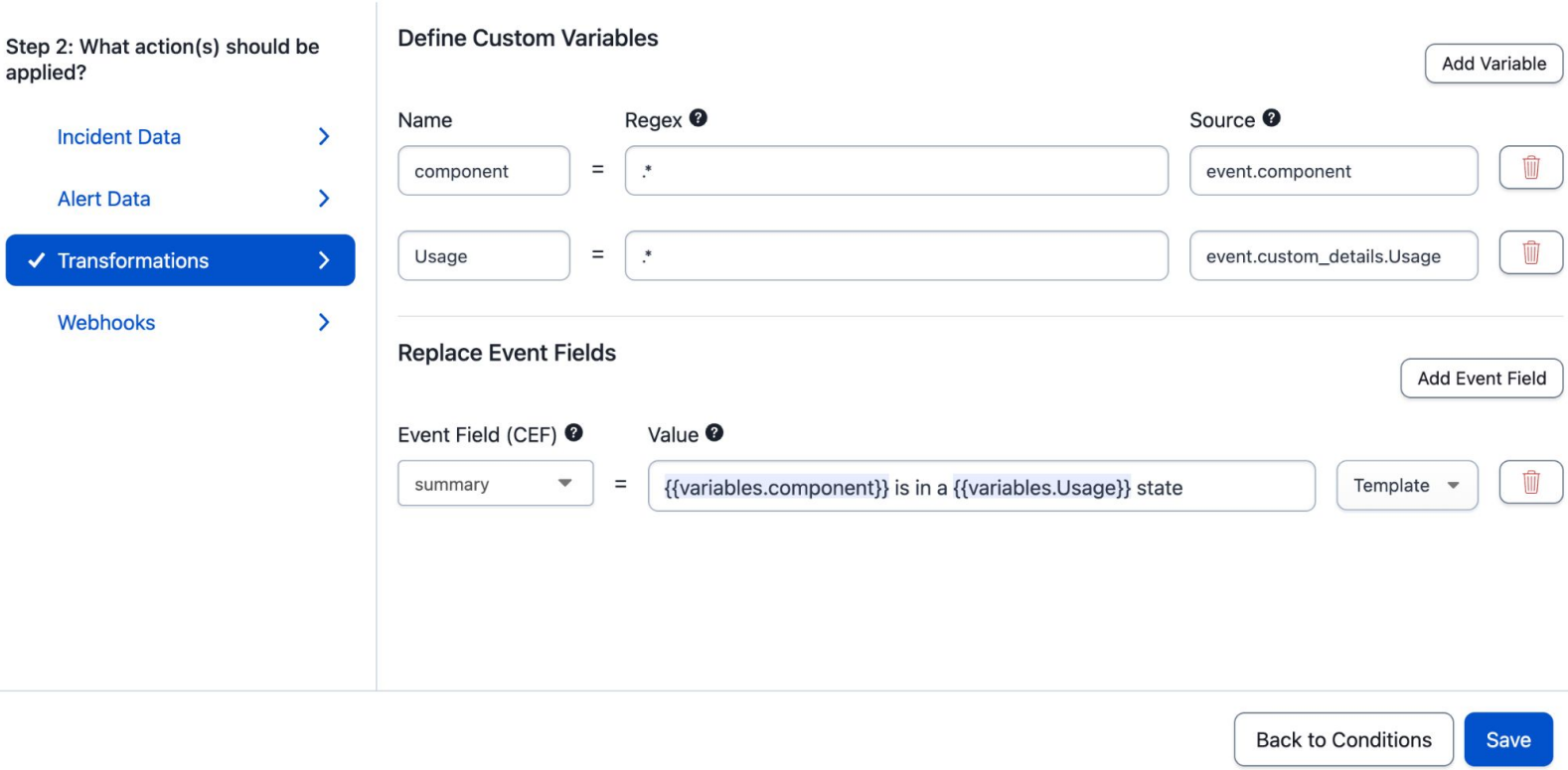
1. Create Else Rule
   a. Select Always (for all events)
   b. Click Next
2. Set Priority to P1
3. Add a note
   a. ex: Check out our wiki for help! www.runbook.example
4. Save





PagerDuty University

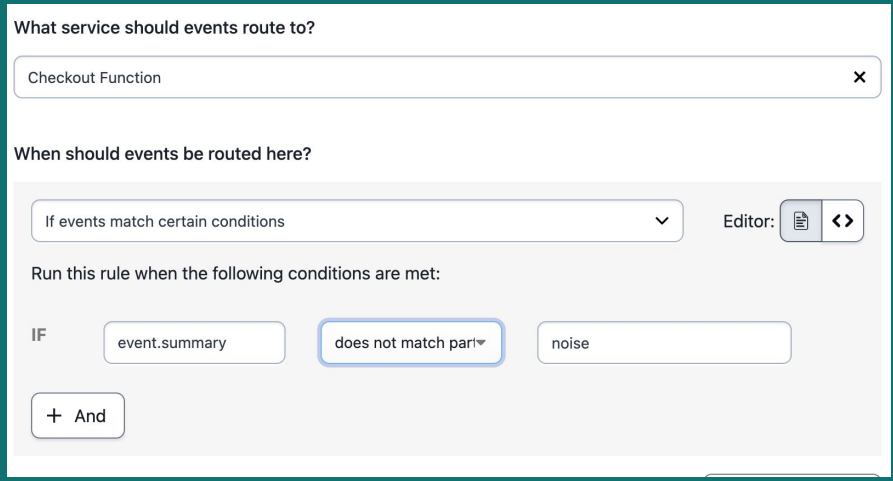
Additional Enrichment



PagerDuty University

Practice Problem: **Service Routes**

1. Navigate to the Service Routes tab
2. Click on the New Service Route button
3. Route to the Checkout Service
4. Set the condition payload.summary **does not match** the word noise
5. Click save



PagerDuty University

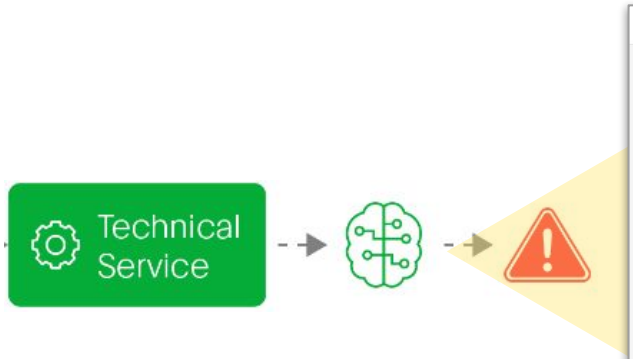
PagerDuty

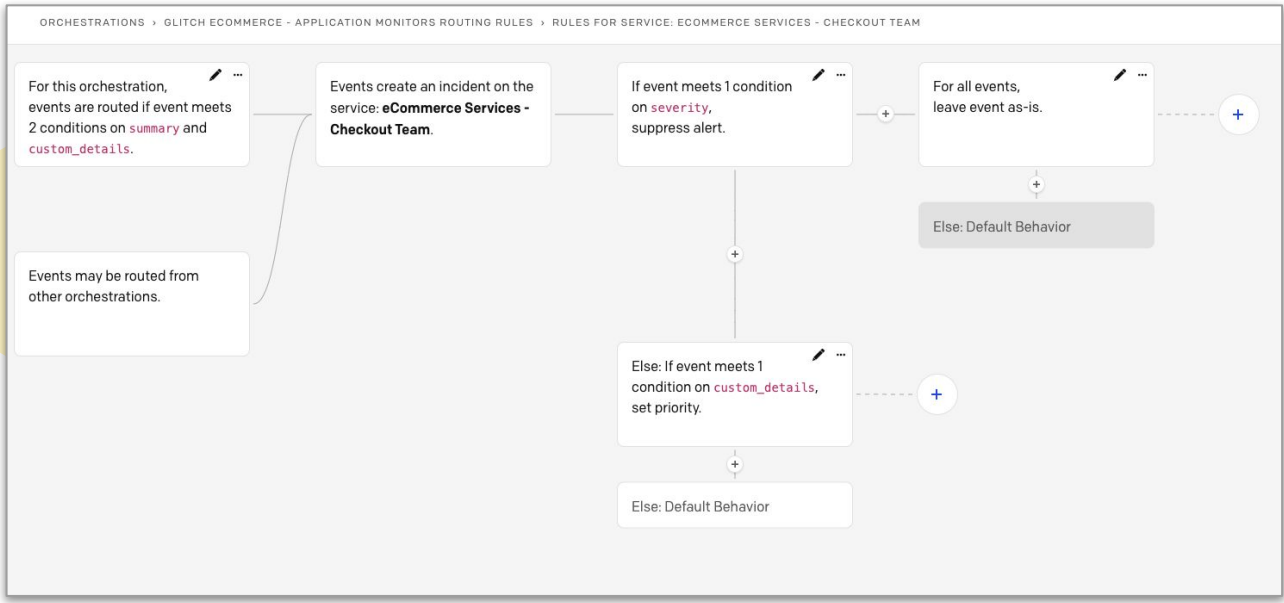
Send in events!

- View updated priority and summary
- Check out notes
- What about those noisy alerts?

💡 Think about it: How might you edit your Checkout Service further?

Service Orchestration Rules

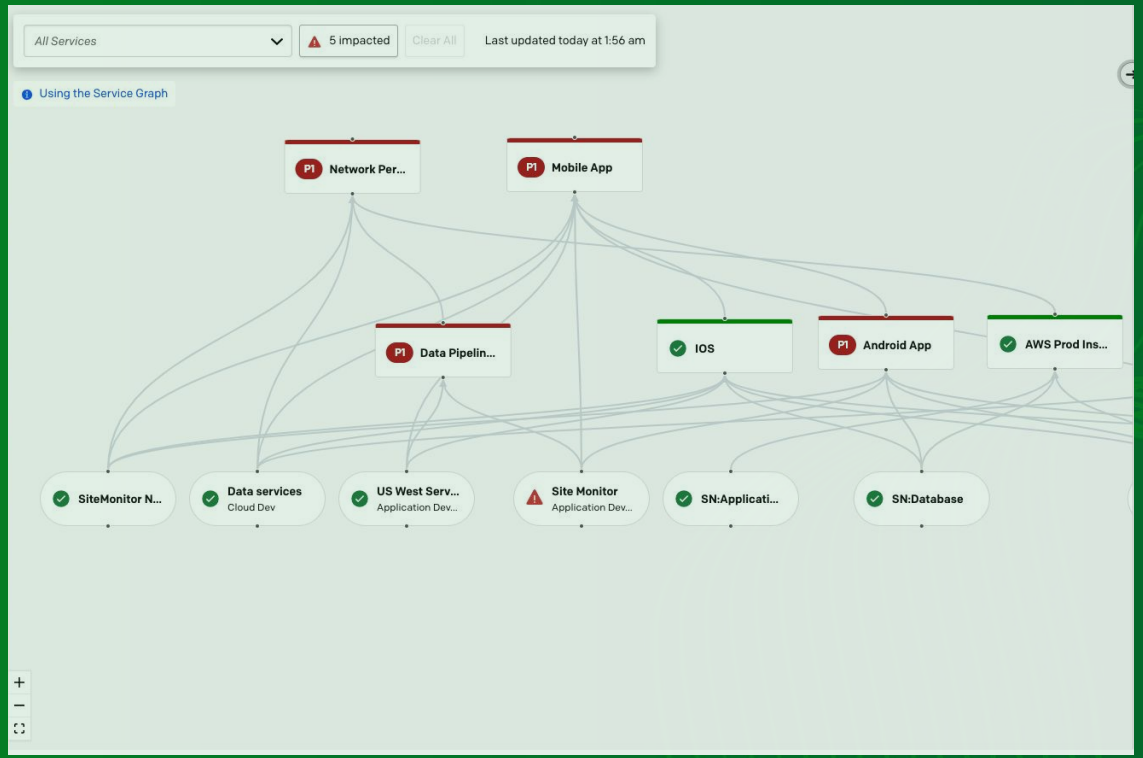




PagerDuty

Root Cause Analysis

Service Graph

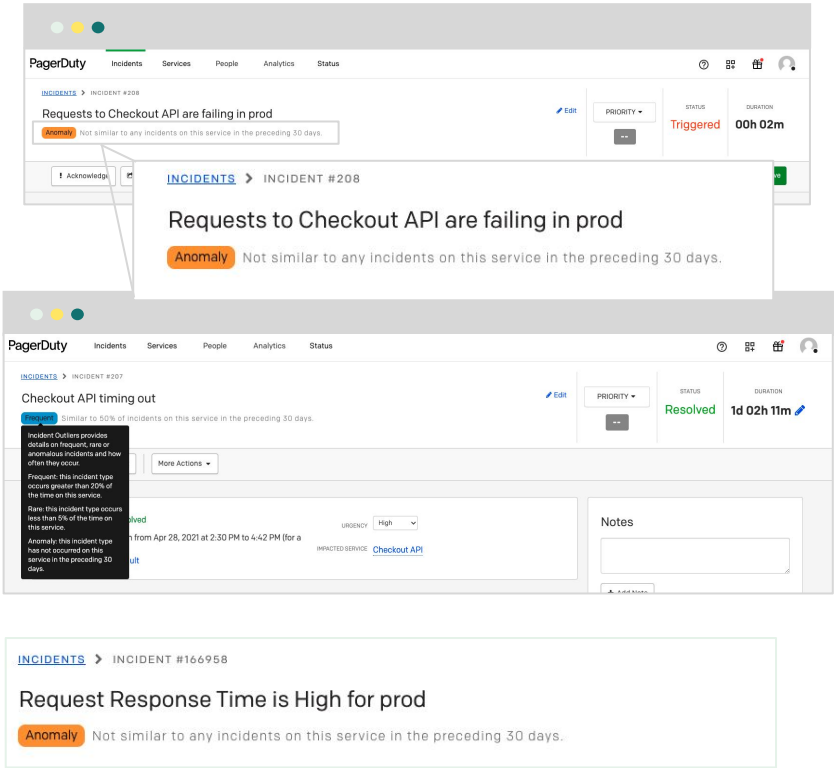


- Understand how technical and business services work together at a glance
- See the full blast radius of an issue
- Zero-in on probable cause

Outlier Incidents

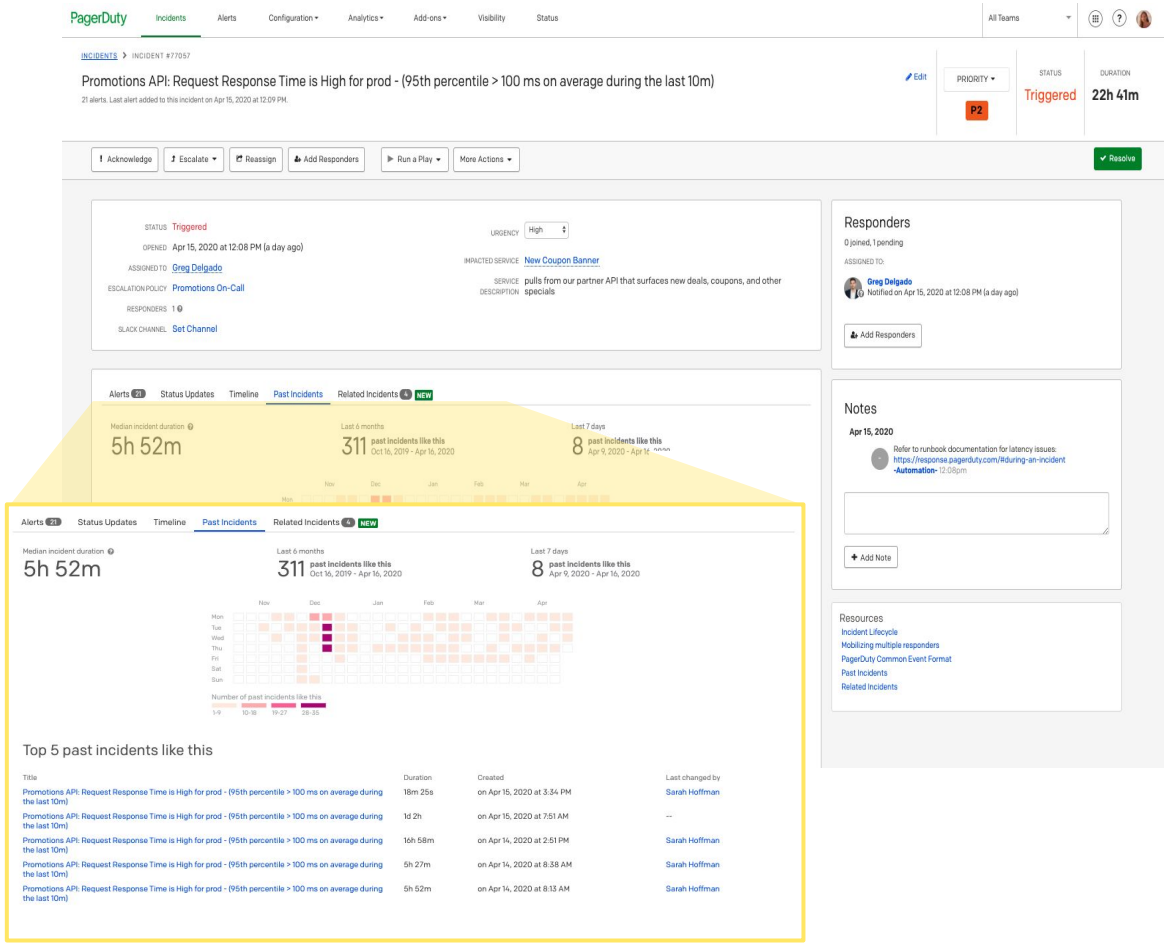
Using historical incident data identify incident types that are anomalies, rare occurrences or frequent offenders.

- Provide context on frequency of certain types of incidents
- Classification of incidents as anomalies, rare or frequent events
- Analytics on frequency of occurrence



Past Incidents

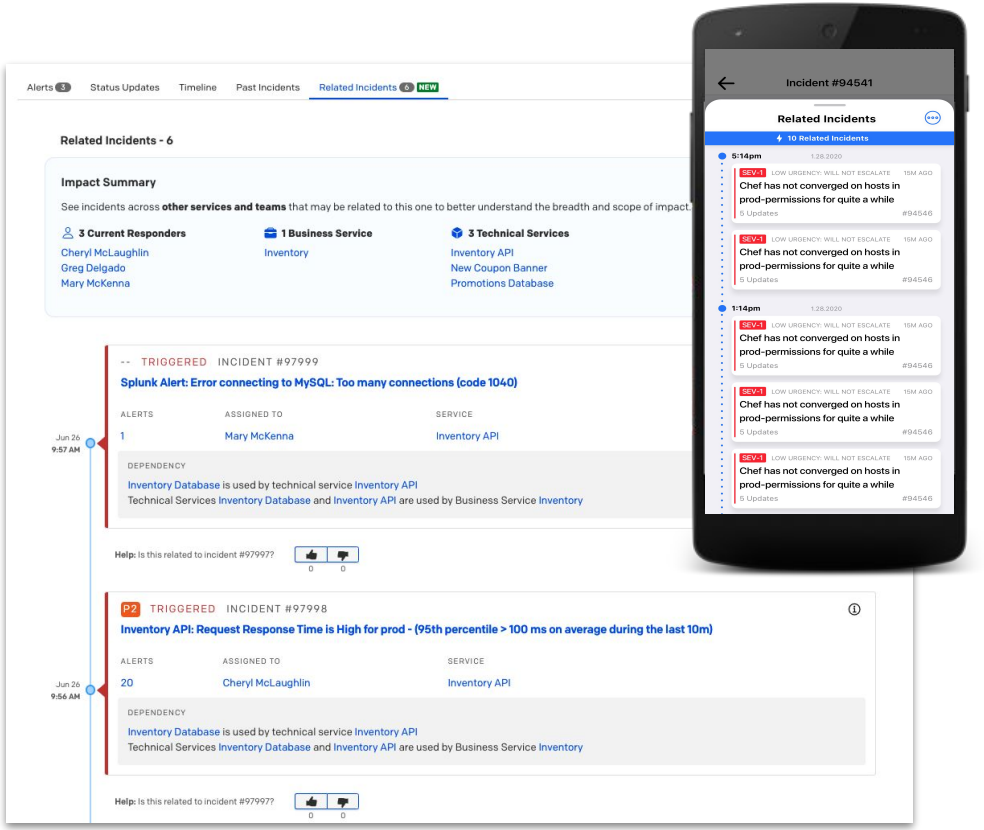
- Accelerate triage through past response data from similar incidents
- Find out who resolved it and how it was resolved



Related Incidents

View real time contextual insights across multiple services and teams

- See what is happening right now across the business
- Understand if issue is local or impacts others
- Find the right people and work together to fix the problem

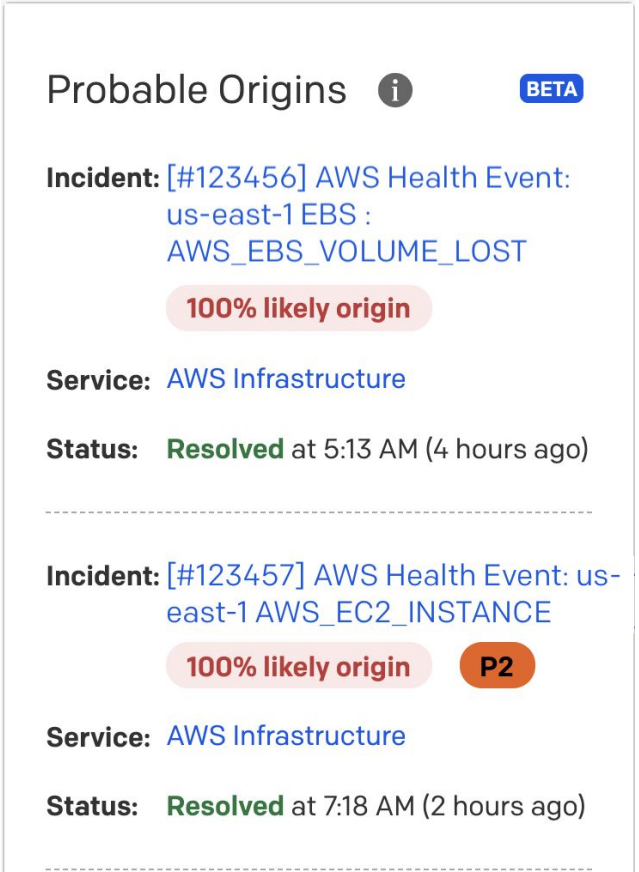


PagerDuty

Probable Origin

- Jumpstart triage efforts with an auto-generated list of likely origin points for faster resolution.

- Use historical data of correlated incident patterns to surface where (and where not) to look first when troubleshooting major incidents.



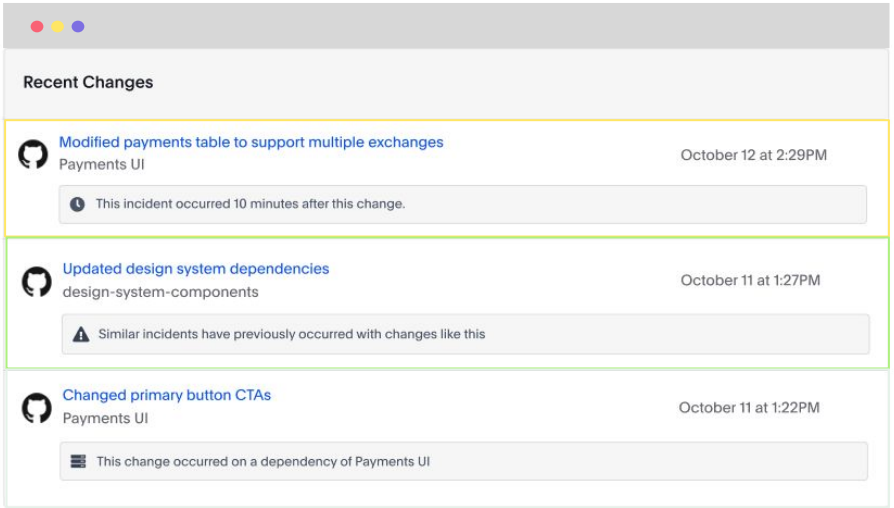
PagerDuty

How often do changes lead to incidents?

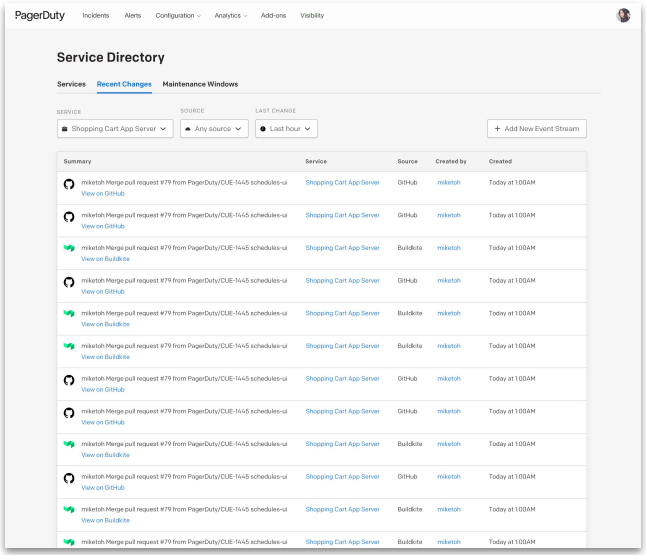
Change Correlation

Surface the changes that most likely caused the incident based on time, related service, or machine learning analysis of similar incidents

Immediately eliminate irrelevant changes and pinpoint the potential contributing factor



Change Events



Know exactly what changes occurred and whether it could be the cause of an incident

Prevent a problem from getting worse and coordinate an effective response with contextual information about recent changes

PagerDuty

Automation - Now we know there's a problem. What can we do?

PagerDuty

Event Driven Automation

Virtual Responder or Responder assisted

Automated diagnostics & resolution

+

Runbook Automation

Access to knowledge and tools, agnostic for scripting or Infrastructure as Code.

Allow developers to run code and democratize automation for larger groups.

+

Workflow Automation

Simplicity in mind

Built for Low Code/No Code use cases with human in the loop.

Enterprise citizen developers

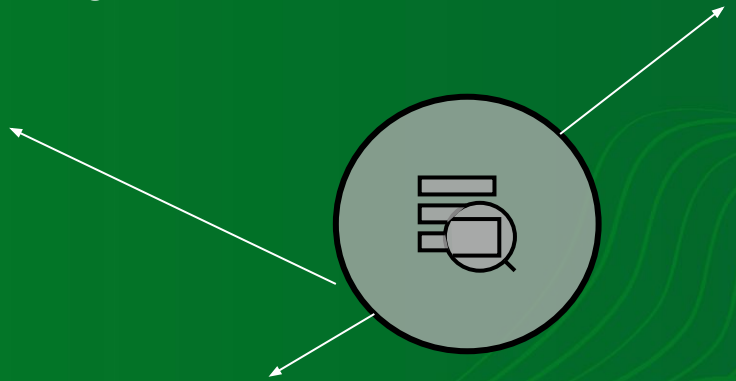
Generative AI/Machine Learning

Noise Reduction, with human reinforcement
PagerDuty CoPilot
Enhance & Simplify Context
Describe and build Runbooks

PagerDuty

Event Driven Response & Diagnostics

Enrich existing events with relevant data
Time, date, status & logs
Platform status
Service status
3rd party status
Kubernetes status
Restart Servers
Restart Services
DB Unlocks
Flush Storages
Clearing Files/Memory
Adding more Disk or Memory Space
Open/Update/Close Tickets
Initiate DR Failover
Healing
Escalation



Data Distribution

Task / Job Scheduling
ETL (Extract-Transform-Load)
File Transfers
Mass data Removal
GDPR data removal
Complex Workflow / Rules
Big Data Replication
Data Remodeling
Dbase jobs, restart
Export-Import mass data

Service Request Automation

Infrastructure Provisioning
Auto shut down unused multi cloud resources
Onboarding/ Deleting Users
Decommission Hardware
Adding Servers
Adding Storage
Software Deployment
Software Updates
De/Provisioning AWS Services
Opening Ports, Switching/Routing
Production Patching
Vulnerability Patching
Increasing Capacity
Security Settings
Validate Security
Change Configs
Adding VLANs
Creating Slack channels
Adding DNS hosts
Firewall port settings
SSL Certificates validation checks
Get next available IP from DDI
White list/ blacklisting IP/ domains
Dbase creation inside a SQL instance
Benchmark testing
…
….(creativity)

General Use Case Examples

Source: https://docs.The Platform.com/docs/manual/plugins/kubernetes-plugins-overview.html

PagerDuty

Event Driven Response & Diagnostics

Getting Pod Status & description
Getting Cluster Status
Running Processes & Limits
Get & Parse events
Tail container logs
Execute in-container commands
Run Kubernetes Jobs
Run Ephemeral containers, with additional tooling to capture debug state
Check other status (eg cron jobs)
Rollout history

Service Request Automation

Self Service for Kubernetes application deployments
Job Trigger delegation with Guardrails
Interaction with ITSM Tools
Manage cloud services, AWS, Azure, GCP
Wrap existing developer tools
Trigger existing pipelines
Trigger 3rd party APIs

Increasing Resource limits for deployments
Triggering deployments
Triggering external components (eg Jenkins/Git/Azure Pipelines)
Pod Deletion
Scale deployment to zero, then back up to original
Undo deployments
Execute script inside container
Execute K8s jobs

Kubernetes Examples

PagerDuty

Event driven Response

Revoke access (eg AD, Okta etc)
Rate Limit API (eg Network Gateway)
Disable Network Interface
Block specific IP Address (IP Tables eg)
Shutdown a service
Disconnect a network share
Stop Running Processes
Disable a user account
Change network configuration via terraform
Update CDN Settings
Update DNS Settings
Run additional direct SSH commands to appliance
Suspend an account
Collect logs
Snapshot Environments
Invoke devops pipeline to rebuild
Apply Patches
Reset sessions

Service Request Automation

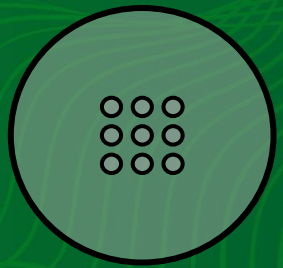
User account creation/deletion.
Password reset requests.
Deploying software updates.
Security patch application.
Log file parsing/analysis.
Password complexity checks.
User account lockouts.
Disk encryption tasks.
Firewall rule additions.
VPN connection testing.
Security alert notifications.

Network Examples

PagerDuty

Self Service/Scheduled DBA Orchestration

Backup/Verify
Apply Patches
Check/Extend Disk space
Create/Modify/Delete User Accounts
Analyze,Optimise Queries
Execute Data Integrity Scripts
Manage DB Log Rotation
DB Switch over for DR
Removal or correction of outdated, duplicated, or incorrect records.
Moving Transaction Logs
Automate the application of schema changes across multiple environments.
Automate the process of moving data from one database or format to another.
Distribute Reports
Package SQL Queries for less technical teams



Database Admin Examples

Exercise 1
First Job

Objective

Utilise Runbook automation to run a basic command
Save that command as a reusable job that could be triggered by:

- Colleagues
- An event
- A schedule
- A webhook
- An API call from a website

Scenario

- You are on call
- Its 3am. Your phone pings.
- You are notified by PagerDuty of a service issue.
  “Cannot Run ServiceAPP - Insufficient space on volume”,
  And its having a customer impact…

You open your Runbook to see what happens with this kind of failure…….

Scenario

Responder Runbook

Operator Guidance for error…

1. This error can occur when the directory fills with small temporary files.
Instructions
Log onto the machine and run the following command
to assess filesystem.

```
find /home/ec2-user/temp -type f -size -100k
```

Conclusion

Your very first job was created by running a single command

- You used a command in your runbook
- You ran it interactively
- You saved it as a reusable job that could be triggered by

    - An event such as an Incident
    - Manually from within PagerDuty
    - From an ITSM tool like ServiceNow or Jira
    - From the self-serve UI
    - From a Webhook from another application
    - From a schedule

Multiple end point commands,

Exercise 2
Automation Actions

Scenario

- You are on call (again)
- Its 3am. Your phone pings.
- You are notified by PagerDuty of a service issue.
  “Machine Resource issue on environment. Customer Experience Impacted”

You open your Runbook to see what happens with this kind of failure…….

Scenario

Responder Runbook

Operator Guidance for error…

1. You will need to run 3 different sets of diagnostics to figure out what's Happening.

   Log onto the machine with your credentials

- **CPU : ps -eo pcpu,pid,cmd | sort -k 1 -nr | head -5**
- **Memory: ps -eo pmem,pid,cmd | sort -k 1 -nr | head -5**
- **Disk: sudo du -Shx / | sort -rh | head -10**

Objective

Utilise a pre-built Automation action to trigger both manually and automatically from a PagerDuty Event

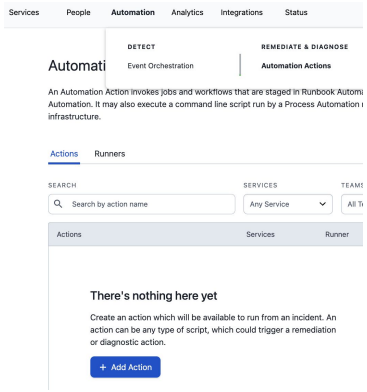
- Set up your first Automation Action in PagerDuty
- Gather multiple diagnostics information in Incident notes
- Trigger incident and manually run diagnostics process

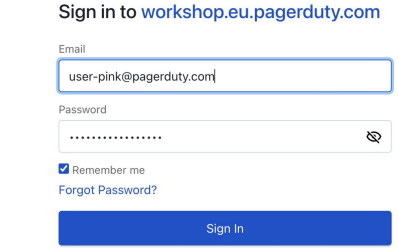
Actions

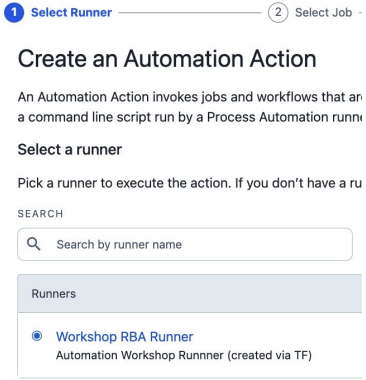
Do this:

1. Log into PagerDuty Incident Management using your credentials
2. Select the Automation Tab, Automation Actions
3. Click on Add Action
4. Select the Workshop RBA connection to pick up the jobs for the server

Screenshot/Notes







Actions

4. Select Job to populate Job list

5. Use Command + F (Shift F for windows) in your web browser find your project name

6. Browse through until you find this Job for your Project. Eg user-red

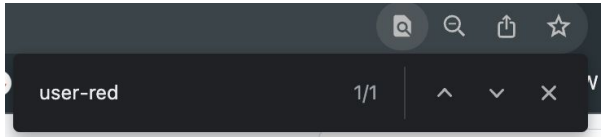
The job is called

2. Diagnostics - Top CPU & Memory Processes and Top Disk Consuming Files

Select this job by checking the option box.

Screenshot/Notes

Select Job



2. Diagnostics - Top CPU & Memory Processes and Top Disk Consuming Files
Diagnostics of a Linux Node's CPU, Memory & Disk to a PagerDuty incident.

user-red_pagerduty.com

Actions

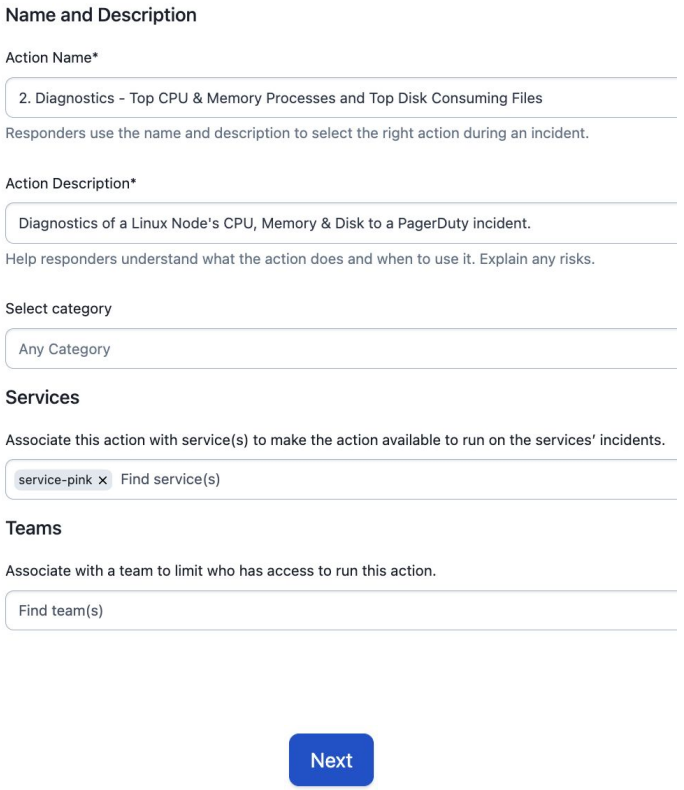
7. Keep the defaults but select the Service to match your service eg service-red

(This is where you can scope an action to multiple services and teams)

8. Add your Team eg team-red

8. Select Next

Screenshot/Notes



Actions

We can add any data from the event we need to pass over in the argument
(eg Server name, Namespace, Container Name). For this we will just use the Incident ID.

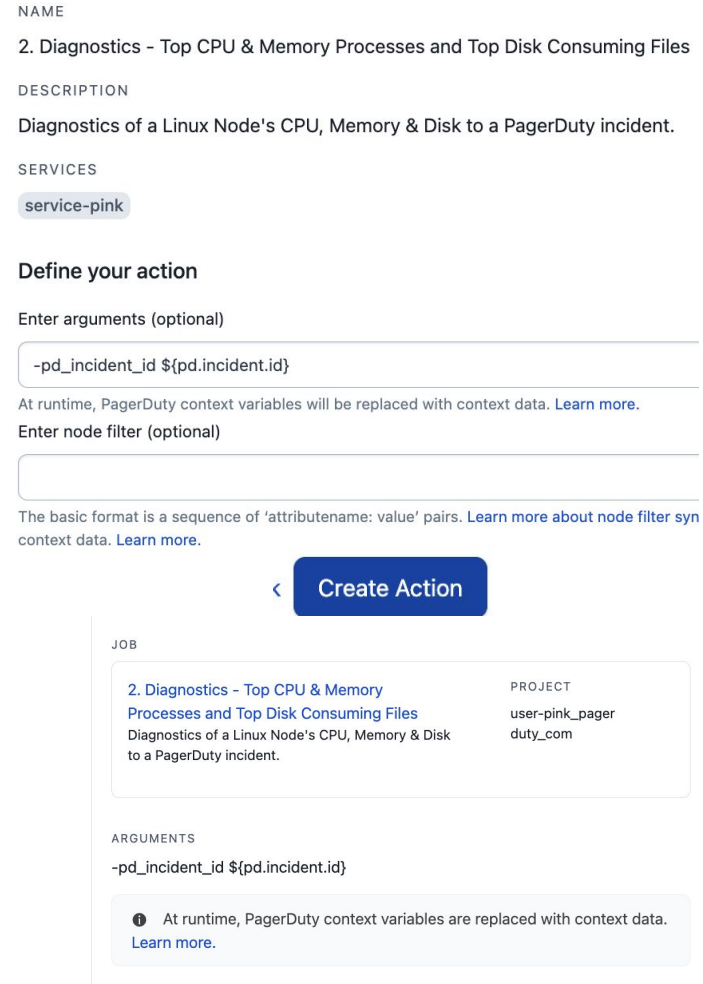
9. Copy Paste this into the argument field

-pd_incident_id ${pd.incident.id}

10. Hit Create Action to finish

Congratulations you have created your first Automation Action

Screenshot/Notes

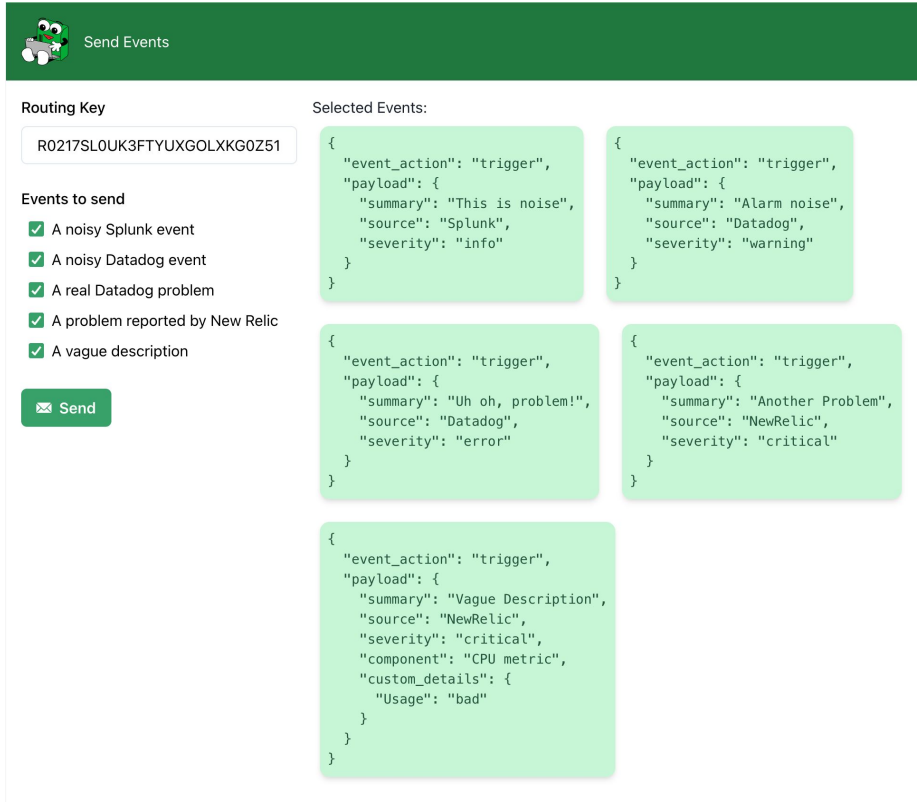


Actions

Let’s trigger an incident to test the Automation Action:

1. Select your Global Integration Key
2. Go to
3. Paste your integration key
4. Check the box for an event
5. Send!

Screenshot/Notes



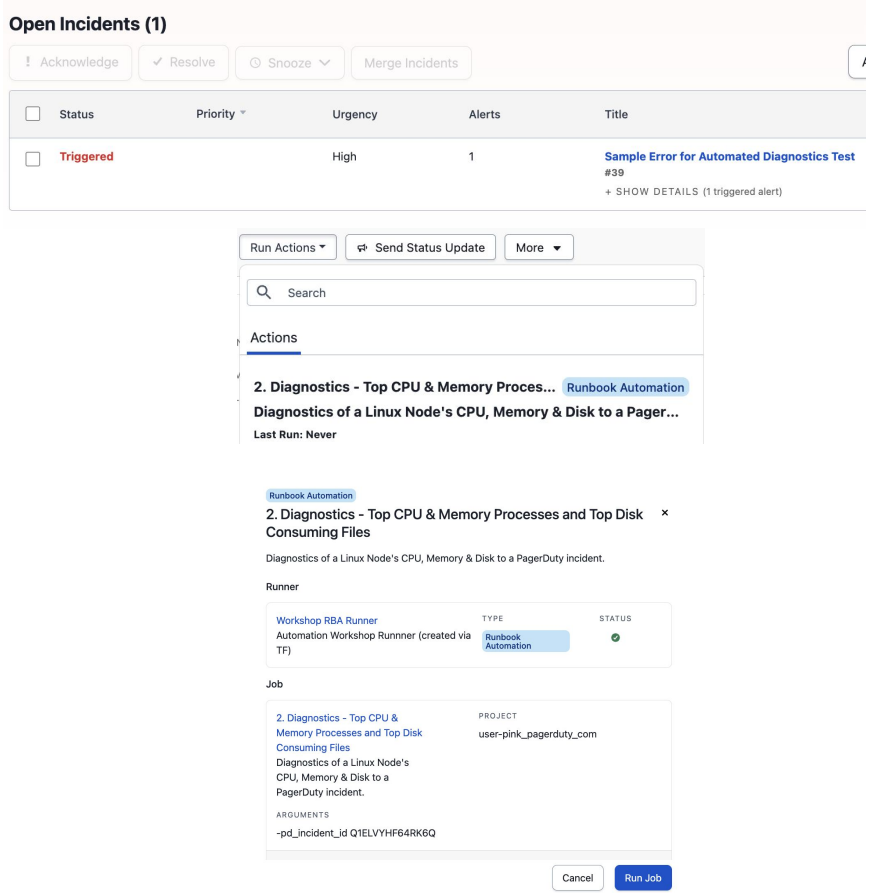
Actions

5. Back to Pagerduty Incident Management

6. Open the incident on your service

7. Select Run Actions, and run the Diagnostics Job

Screenshot/Notes



Actions

Shortly you should be able to see the three diagnostics jobs in the notes and the timeline.

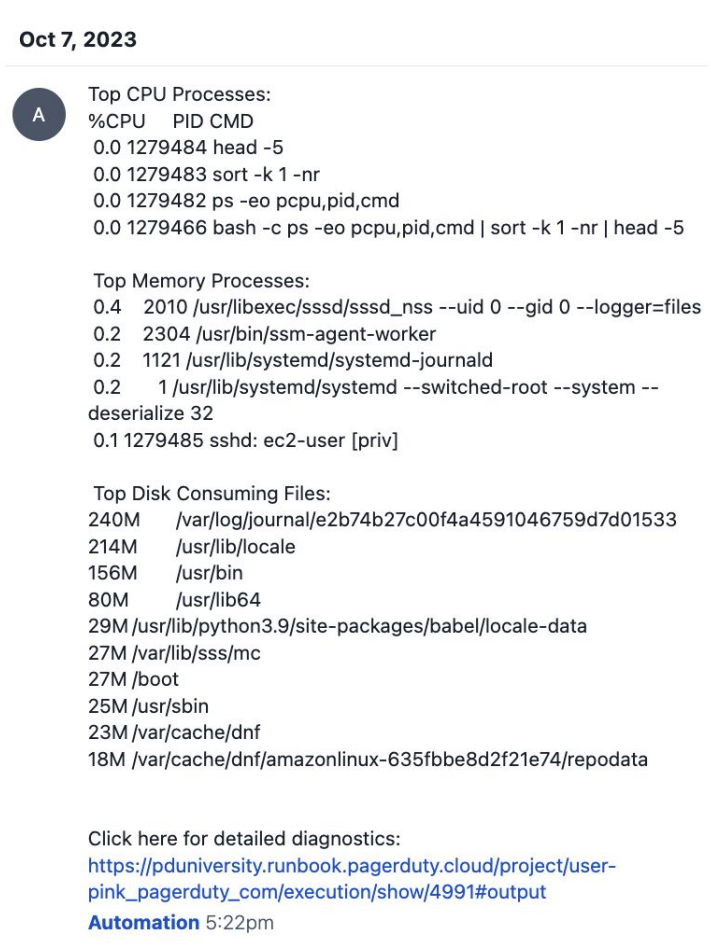
(These are also available in Slack/Teams or the mobile app)

9. Click the link, for the detailed diagnostics in the Runbook Automation UI.

10. **Resolve** your incident for our next test

Time permitting, edit the job to investigate the 3 different steps to gather machine level diagnostics.

Screenshot/Notes



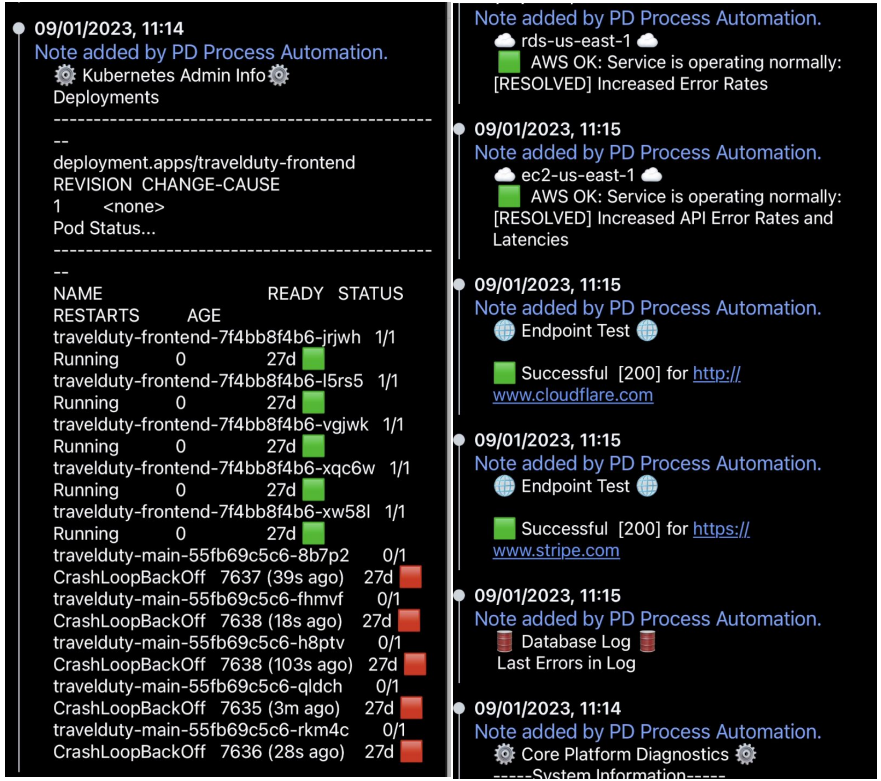
Actions

Clearly these can be more complex, and split across different environments.

Here's an example of Kubernetes, Database and Availability diagnostics.

We have a solution pack containing many modern diagnostics jobs here

Screenshot/Notes



Conclusion

Utilise a pre-built Automation action to trigger both manually and automatically from a PagerDuty Event

- Set up your first Automation Action in PagerDuty
- Gather multiple diagnostics information in Incident notes
- Trigger incident and manually run diagnostics process

Exercise 3 Automate The Trigger

Objective

Manual actions are great for real time,
user-driven response but for diagnostics, automatic is a great solution

- Orchestrate our service to execute the Action each time an incident triggers

Actions

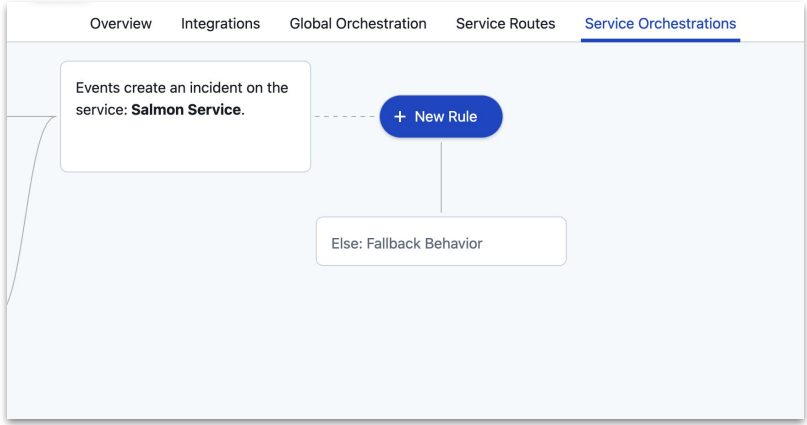
1. Log back into PagerDuty Incident Management

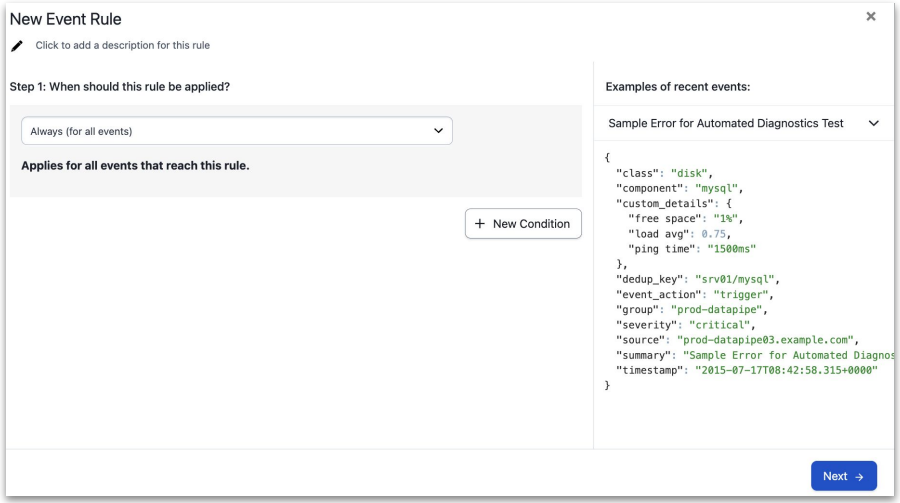
2. Go to your Event Orchestration and navigate to Service Orchestrations

3. Select your service and create a New Rule

4. Leave the condition as *Always (for all events)*

Screenshot/Notes





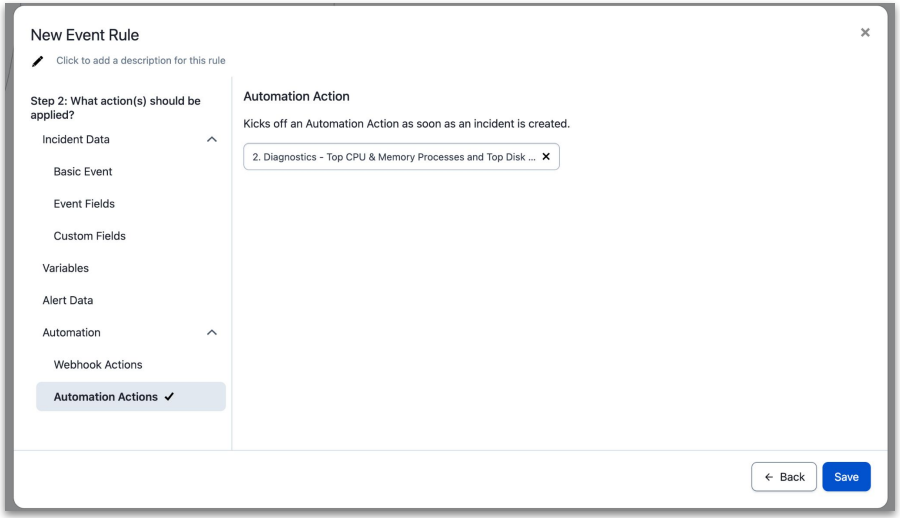
Actions

5. Select Automation > Automation Action

And select our previously created Action

6. Select Save to complete.

Screenshot/Notes

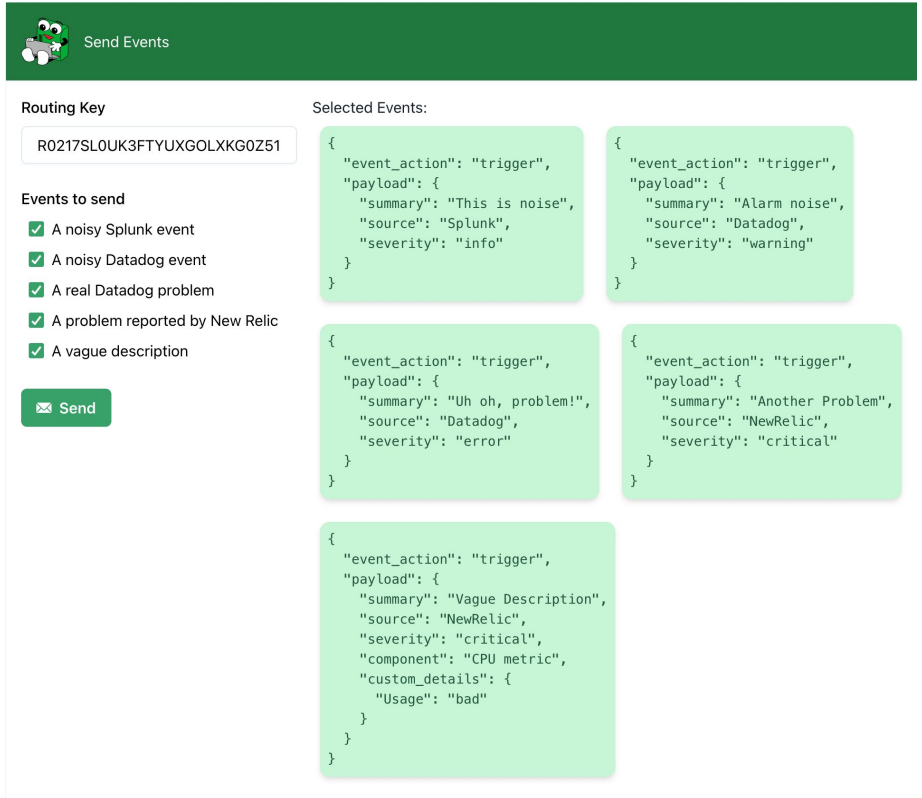


Actions

Let’s trigger another incident:

5. Select your Global Integration Key
6. Go to
7. Paste your integration key
8. Check the box for an event
9. Send!

Screenshot/Notes

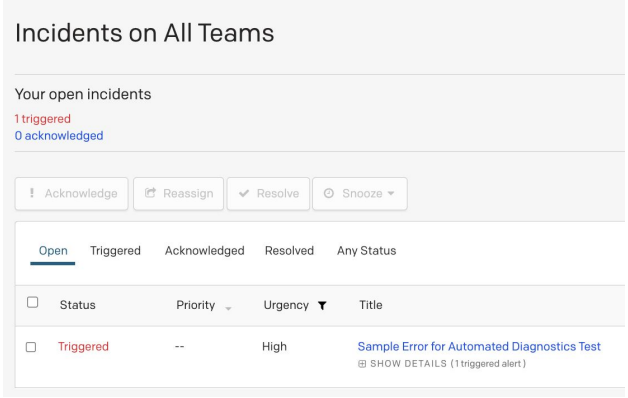


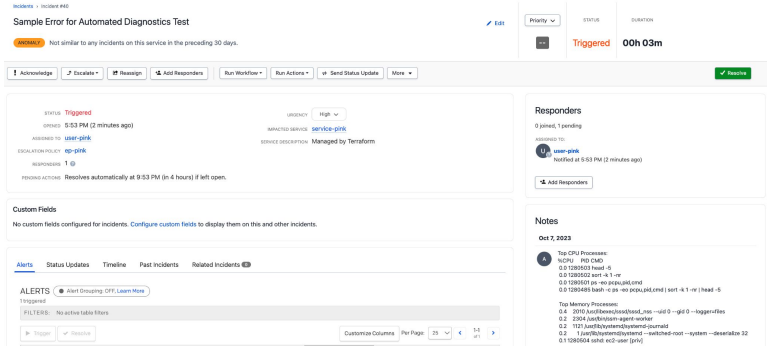
Actions

10. Browse to incidents, Select Incident

11. Diagnostics should be triggered every time the incident fires

Screenshot/Notes





Conclusion

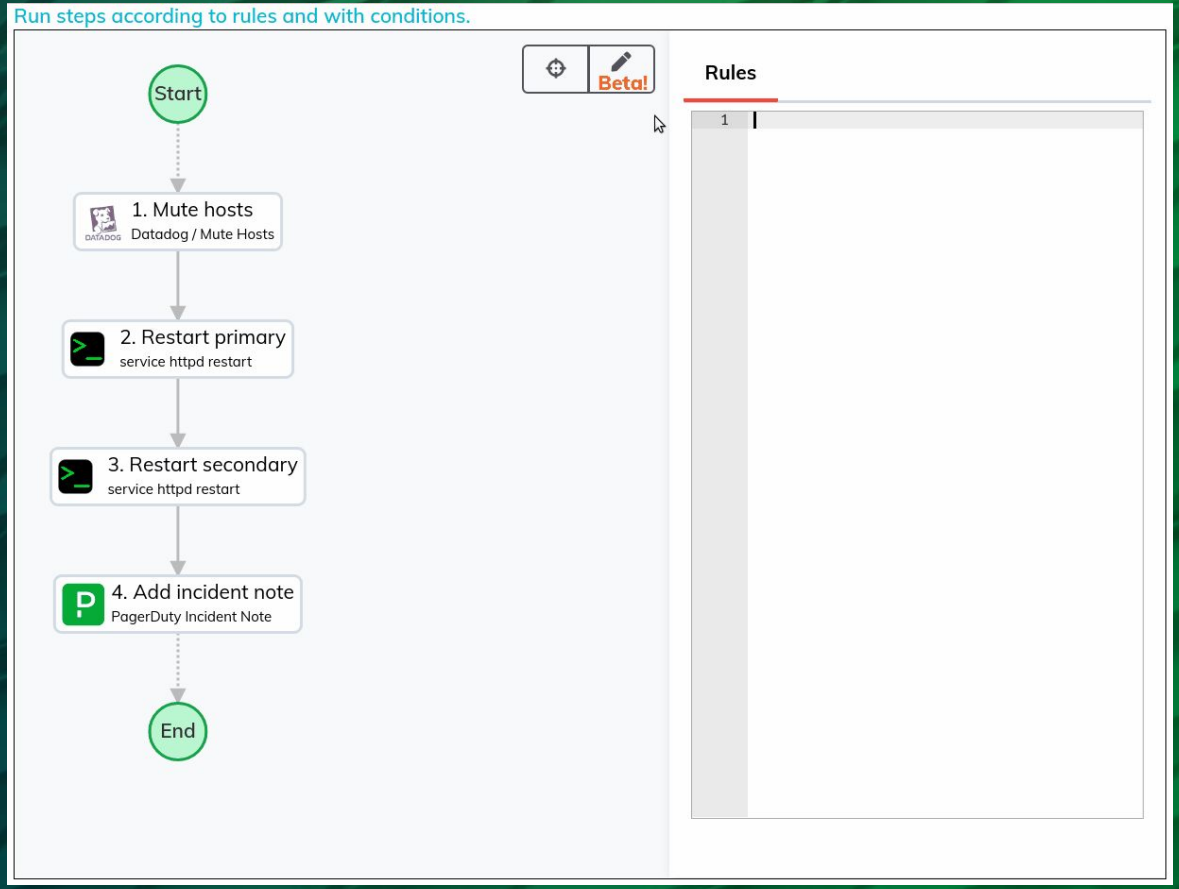
Automation Actions can be triggered as part of an incident flow

- We can use any part of the incoming event to trigger upon, and any part of the event data can be passed on to use in automations.

PagerDuty

Advanced Automation Concepts

Rulesets

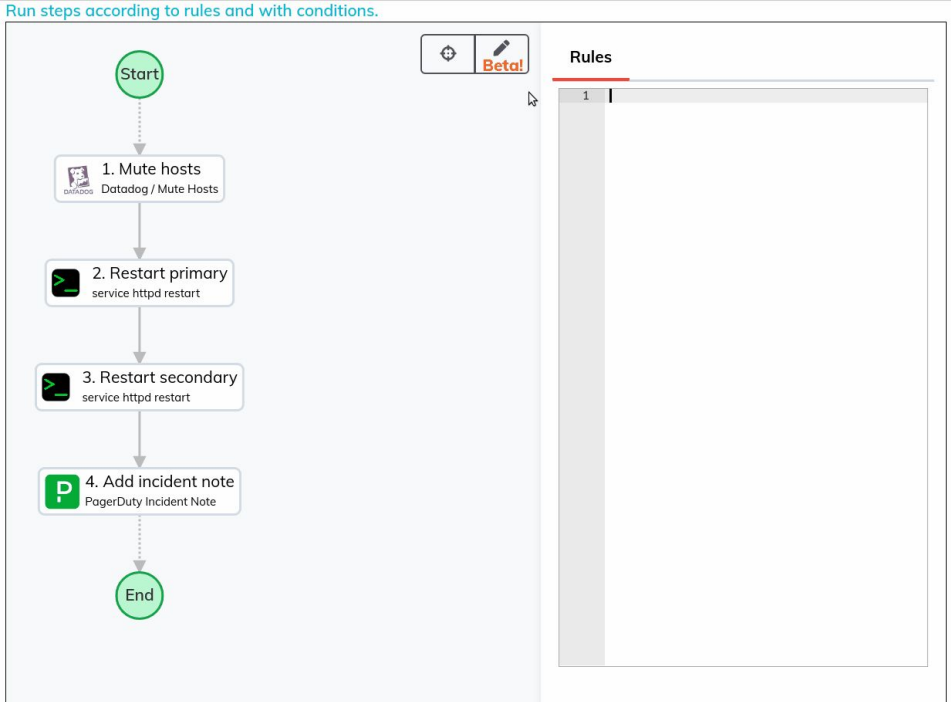


Rulesets

When you need logic you dont have in your script

IF THEN AND OR Logic

Graphic Flow Chart



Conditions

Conditions can define additional checks that must pass before a step can run, or determine when a step can be skipped. For example:

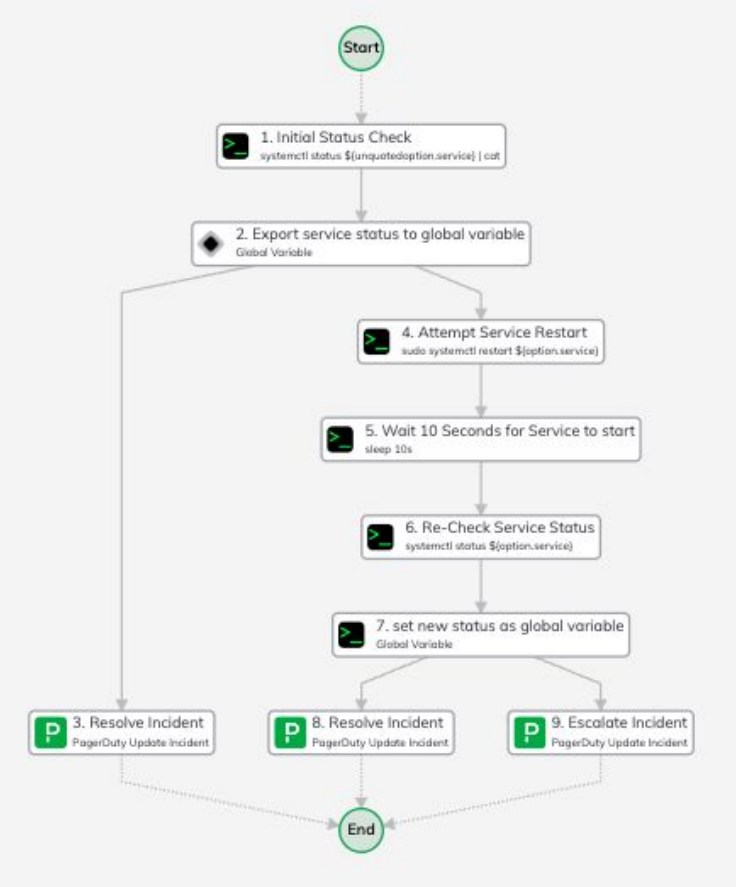
```
    if:expression
    unless:expression
```

expression defines a comparison or match that will be checked.

Valid expressions are:

```
    key.name==string: a context variable such as option.myoption has a certain value
    key.name!=string: not-equal check
    key.name=~pattern: regular expression match
    key.name!~pattern: negative regular expression match
    !key.name context: variable is unset
    key.name>=number: greater than or equal to check
    key.name<=number: less than or equal to check
    key.name>number: greater than check
    key.name<number: less than check
```

Example : Check, remediate,check,Escalate



TIP: Global Variables are essential -
Dont forget to export if you use the Ruleset

Exercise 4
Advanced Automation

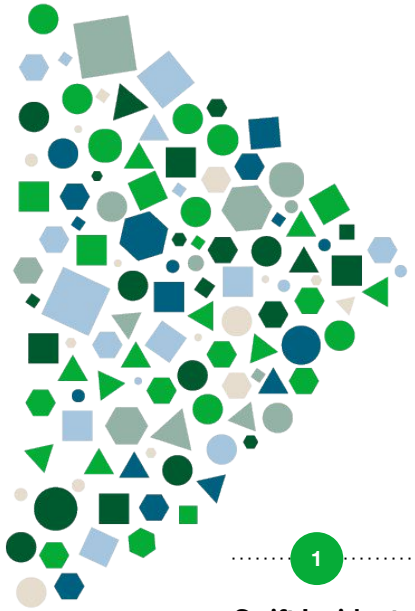
Conclusion

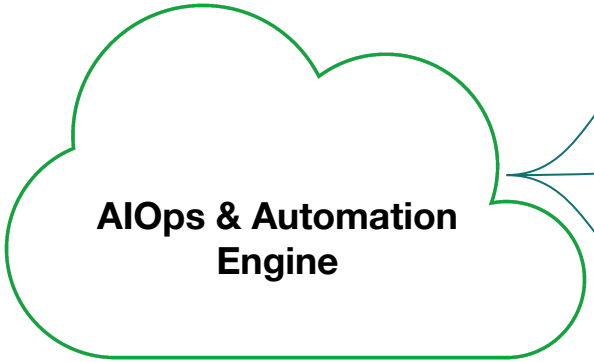
Automation Actions can be triggered as part of an incident flow

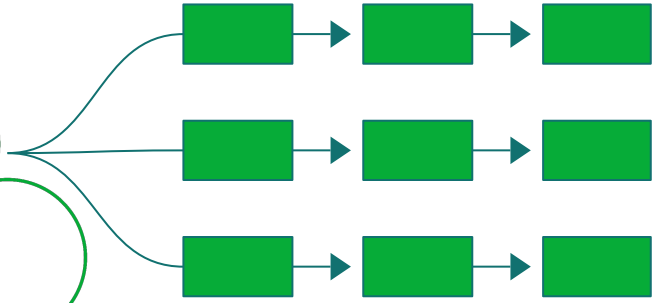
- We can use any part of the incoming event to trigger upon, and any part of the event data can be passed on to use in automations.

Event-Driven Automation

Event-driven automation is a dynamic approach where actions are triggered by specific events or conditions. It harnesses real-time event detection and predefined workflows to respond instantly to critical occurrences.



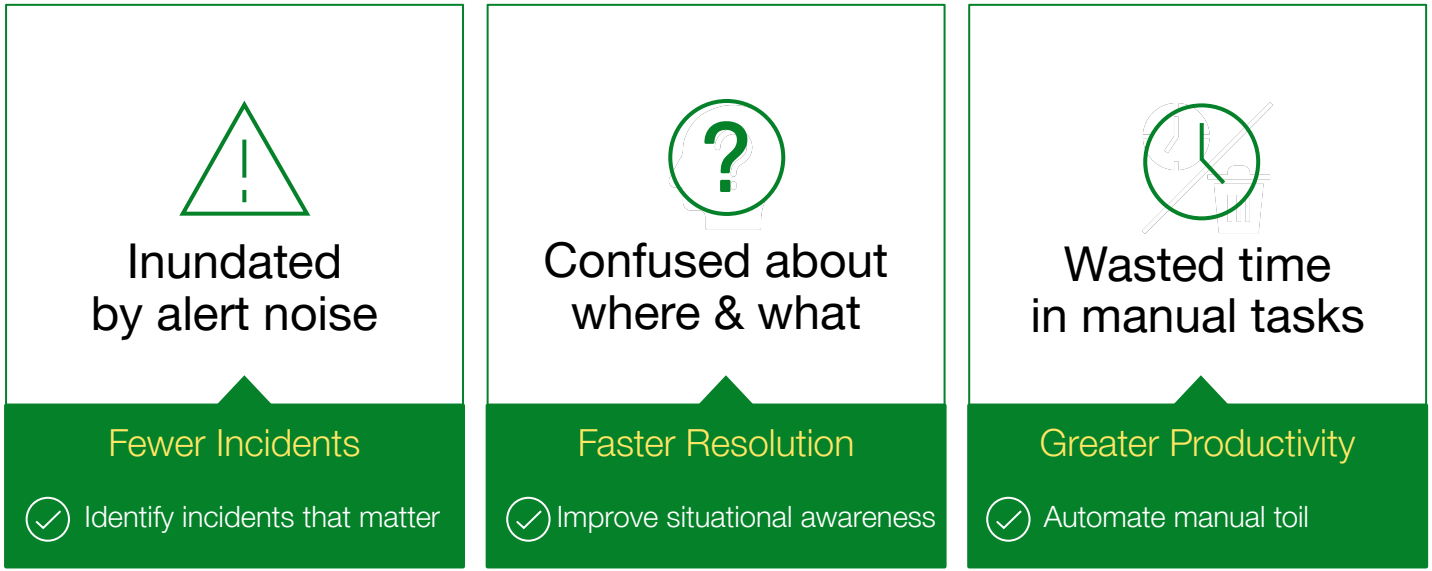




Custom Workflows

1. **Swift Incident Response**
2. **Operational Efficiency**
3. **Proactive Issue Resolution**
4. **Improved Customer Experience**

Solving Challenges with AIOps



Next steps

1. Continue your learning journey with AIOps and Automation
2. See how many manual tasks you can automate with Global Event Orchestration
3. Think of use cases that could be automated using the automation platform
4. Fill out the feedback form

PagerDuty

Thank You!