# Rapid incident resolution

300-level live demo script

## Introduction

In this demo, I’ll show you how Cloud Pak for Watson AIOps helps SREs and IT Ops quickly identify, diagnose, and resolve incidents across mission-critical workloads.

You’ll see how:

- Watson AIOps intelligently correlates multiple disparate sources of information such as logs, metrics, events, and topology;
- All of this information is condensed and presented in actionable alerts instead of large quantities of unrelated alerts, and;
- You can resolve a problem within seconds to minutes of being notified using Watson AIOps’ automation capabilities

We will be using an application called Quote of the Day, which will serve as a proxy for any type of app. This is a content delivery app that serves up random quotations. The application is built on a microservices architecture and the services are running on Kubernetes.

# 1 - Simulating a failure

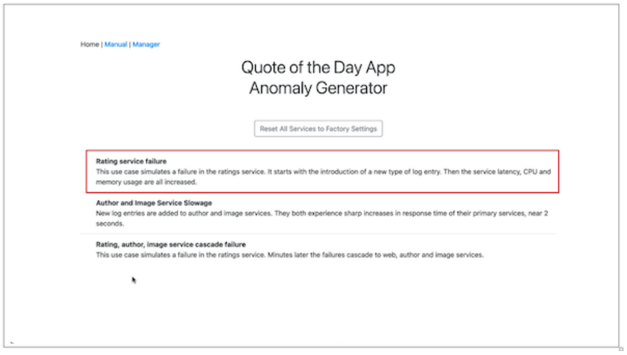
## 1.1 - Navigate to the Anomaly Generator and input sources

**Actions**

- Navigate to the **Anomaly Generator**
- Choose **Rating service failure**
- Click **Start**

**Narration**

To see how this all works, I'm going to generate an anomaly in our application.



Quote of the Day App
Anomaly Generator

**Rating service failure**
This use case simulates a failure in the ratings service. It starts with the introduction of a new type of log entry. Then the service latency, CPU and memory usage are all increased.

**Author and Image Service Slowage**
New log entries are added to author and image services. They both experience sharp increases in response time of their primary services, near 2 seconds.

**Rating, author, image service cascade failure**
This use case simulates a failure in the ratings service. Minutes later the failures cascade to web, author and image services.

**Actions**

- The anomaly generator will take a minute or so to start showing alerts in Slack
- While it is running, navigate to the various input sources: Prometheus and ELK

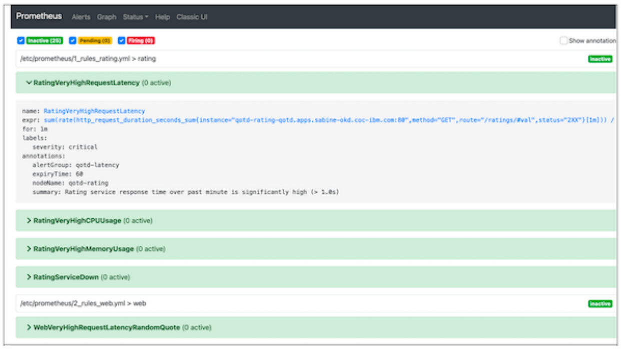
**Narration**

Without Watson AIOps, we would get all sorts of alerts and notifications from multiple sources when a problem occurs.

Prometheus would start firing alerts, but we would have to look at them manually to find out if they are related.

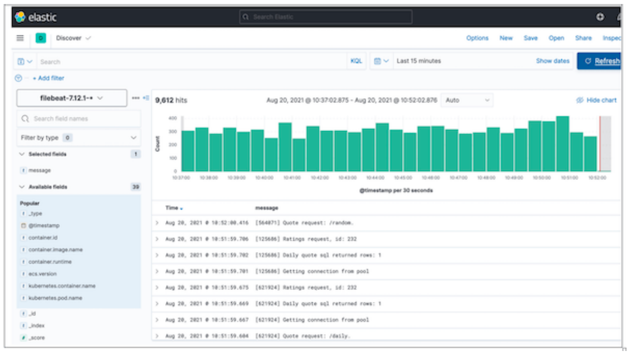
Same with ELK – we might see new types of logs or errors coming in, but again it's time consuming to determine the relationship. Hundreds of logs are streaming in every minute, making it very difficult for a human to keep up with them.

As we'll see in a moment, with Watson AIOps, all of this information gets correlated and presented all in one place. This includes recommendations for how to resolve the incident. We can take action directly from the notification and resolve the incident quickly.



```
name: RatingVeryHighRequestLatency
expr: sum(rate(http_request_duration_seconds_sum{instance="qotd-rating-qotd.apps.sabine-okd.coc-ibm.com:80",method="GET",route="/ratings/#val",status="2XX"}[1m])) /
for: 1m
labels:
  severity: critical
annotations:
  alertGroup: qotd-latency
  expiryTime: 60
  nodeName: qotd-rating
  summary: Rating service response time over past minute is significantly high (> 1.0s)
```

RatingVeryHighCPUUsage (0 active)
RatingVeryHighMemoryUsage (0 active)
RatingServiceDown (0 active)
/etc/prometheus/2_rules_web.yml > web
inactive
WebVeryHighRequestLatencyRandomQuote (0 active)



```
Time                          message
Aug 20, 2021 @ 10:52:00.416   [564871] Quote request: /random.
Aug 20, 2021 @ 10:51:59.706   [125686] Ratings request, id: 232
Aug 20, 2021 @ 10:51:59.702   [125686] Daily quote sql returned rows: 1
Aug 20, 2021 @ 10:51:59.701   [125686] Getting connection from pool
Aug 20, 2021 @ 10:51:59.675   [621924] Ratings request, id: 232
Aug 20, 2021 @ 10:51:59.669   [621924] Daily quote sql returned rows: 1
Aug 20, 2021 @ 10:51:59.667   [621924] Getting connection from pool
Aug 20, 2021 @ 10:51:59.604   [621924] Quote request: /daily.
```

# 2 - Getting notified of an emerging problem

## 2.1 - AIOps formats notification as story and adds affected services to it

**Action**

- Story starts showing up in Slack

**Narration**

Notifications are now appearing in Slack. We're using Slack in this demo, but Watson AIOps also integrates with Microsoft Teams.

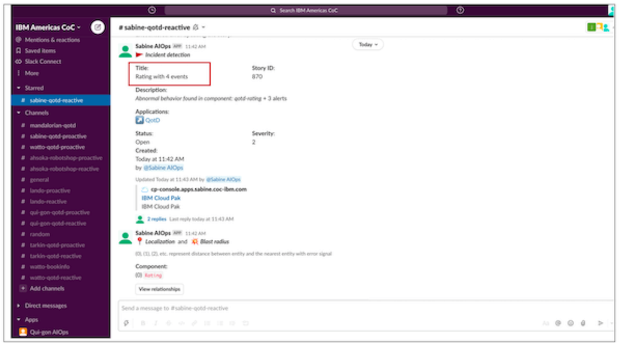
Watson AIOps formats the notifications into a "story" using AI to correlate events, metrics, alerts, and logs. Each story brings together the various notifications for all the affected services by the same underlying issue. Imagine if each piece of data presented in the story was a separate notification – we'd quickly be inundated with alerts.

The story is like a home base for action when a problem arises. Instead of manually correlating things across multiple different tools, it's all right here immediately when the notification is received.

In addition to providing a highly contextualized view of the incident, it enables us to jump in-context to other point tools to explore further details. This helps eliminate tool silos and helps us restore service faster.

This story is telling us there's a problem with the Rating service, which is one of the microservices in our Quote of the Day application.

In the background, the AI has done the work for us and presents us with a curated view of relevant information. It shows which services are affected: the relevant events and alerts that are indicative of the symptoms of this problem; anomalies that Watson AIOps has found in the log files; and similar incidents that have occurred in the past so we can see how they were successfully resolved. We'll explore each of these components in more detail.



Sabine AIOps APP 11:42 AM

Incident detection

Title:
Rating with 4 events

Story ID:
870

Description:
*Abnormal behavior found in component: qotd-rating + 3 alerts*

Applications:
QotD

Status:
Open

Severity:
2

Created:
Today at 11:42 AM
by @Sabine AIOps

Updated Today at 11:43 AM by @Sabine AIOps

cp-console.apps.sabine.coc-ibm.com
IBM Cloud Pak
IBM Cloud Pak

2 replies Last reply today at 11:43 AM

Sabine AIOps APP 11:42 AM

*Localization* and *Blast radius*

(0), (1), (2), etc. represent distance between entity and the nearest entity with error signal

Component:
(0) Rating

**Action**

- Additional affected services are added to the story

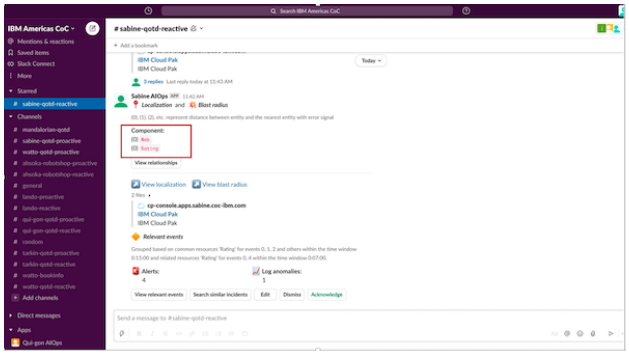
**Narration**

When this story first appeared, the only affected service was the Rating service.

Watson AIOps updates the stories in real time as more information comes in and gets correlated to this story.

So now we can see that in addition to Rating, the Web service is also affected. This is now even more critical since the Web service is the customer-facing front-end of the application, and we need to ensure that users can still access the app.

We need to find the root cause and fix this problem as quickly as possible.



IBM Cloud Pak
IBM Cloud Pak

3 replies Last reply today at 11:43 AM

Sabine AIOps APP 11:42 AM
📍 *Localization* and 💥 *Blast radius*

(0), (1), (2), etc. represent distance between entity and the nearest entity with error signal

Component:
(0) Web
(0) Rating

2 files

cp-console.apps.sabine.coc-ibm.com
IBM Cloud Pak
IBM Cloud Pak

🔶 *Relevant events*

Grouped based on common resources 'Rating' for events 0, 1, 2 and others within the time window 0:15:00 and related resources 'Rating' for events 0, 4 within the time window 0:07:00.

Alerts:
4

Log anomalies:
1

# 3 - Determining which service caused the problem

## 3.1 - Review the notification to learn about issue

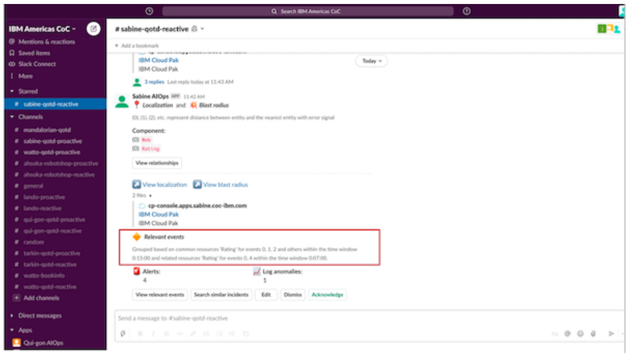
**Action**

- Scroll to the **Relevant events** section of the story

**Narration**

We need to find out where the issue began so we can prevent it from causing cascading failures across the components of the application.

Instead of having to go to Prometheus or another tool to look at the alerts, we can see them right here from the notification. Watson AIOps has determined that these events are related, and provides an explanation for how it determined the relationships. We can see that there are two groups of events based on related resources and the timing of the events.



IBM Cloud Pak
IBM Cloud Pak

3 replies Last reply today at 11:43 AM

Sabine AIOps APP 11:42 AM
Localization and Blast radius
(0), (1), (2), etc. represent distance between entity and the nearest entity with error signal

Component:
(0) Web
(0) Rating

2 files
cp-console.apps.sabine.coc-ibm.com
IBM Cloud Pak
IBM Cloud Pak

Relevant events
Grouped based on common resources 'Rating' for events 0, 1, 2 and others within the time window 0:15:00 and related resources 'Rating' for events 0, 4 within the time window 0:07:00.

Alerts: 4
Log anomalies: 1

**Actions**

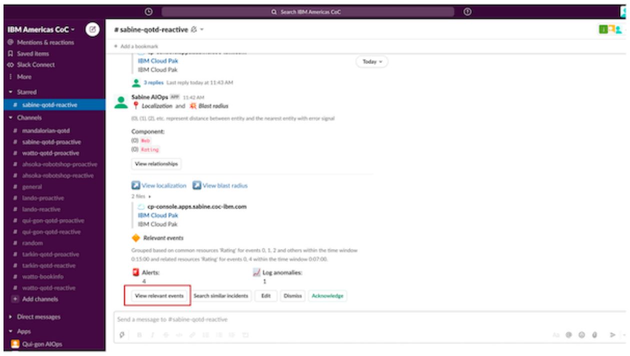
- Click on the **View relevant events** button at the bottom of the notification
- A pop-up will appear with the grouped events

**Narration**

We can inspect the grouped events right here, without searching for them in another tool.

It looks like the memory and CPU on the Rating service increased significantly. This is causing a significant slow-down in response time on both the Rating and Web services.

Based on this information, it seems that the Rating service is the source of the issue. But let’s get a bit more detail – this time from the log files.



IBM Cloud Pak
IBM Cloud Pak

3 replies Last reply today at 11:43 AM

Sabine AIOps APP 11:42 AM
Localization and Blast radius
(0), (1), (2), etc. represent distance between entity and the nearest entity with error signal

Component:
(0) Web
(0) Rating

View relationships

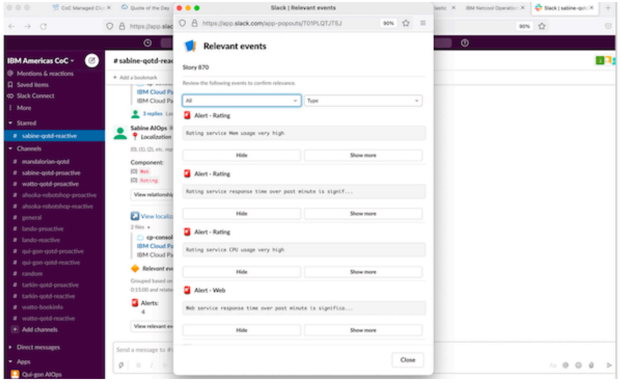
View localization View blast radius
2 files
cp-console.apps.sabine.coc-ibm.com
IBM Cloud Pak
IBM Cloud Pak

Relevant events
Grouped based on common resources 'Rating' for events 0, 1, 2 and others within the time window 0:15:00 and related resources 'Rating' for events 0, 4 within the time window 0:07:00.

Alerts: 4
Log anomalies: 1

View relevant events | Search similar incidents | Edit | Dismiss | Acknowledge

Send a message to #sabine-qotd-reactive


Slack | Relevant events
https://app.slack.com/app-popouts/T01PLQTJT5J

Relevant events

Story 870

Review the following events to confirm relevance.

All | Type

Alert - Rating
```
Rating service Mem usage very high
```
Hide | Show more

Alert - Rating
```
Rating service response time over past minute is signif...
```
Hide | Show more

Alert - Rating
```
Rating service CPU usage very high
```
Hide | Show more

Alert - Web
```
Web service response time over past minute is significa...
```
Hide | Show more

Close

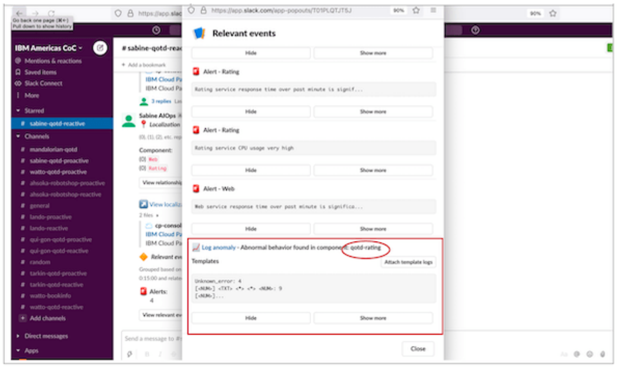
## Action

- Scroll down past the alerts to show **Log anomaly**

## Narration

Instead of needing to go to Kibana and manually sort through the hundreds or thousands of log entries that come in every minute, Watson AIOps has found several anomalies in the log files and presented them here. It trains on the log files of the application when it's operating normally, and continually monitors for deviations from that baseline.

We can see that the anomalies are occurring on the Rating service, which fits with what we saw in the alerts.


https://app.slack.com/app-popouts/T01PLQTJT5J

Relevant events

Hide | Show more

Alert - Rating
```
Rating service response time over past minute is signif...
```
Hide | Show more

Alert - Rating
```
Rating service CPU usage very high
```
Hide | Show more

Alert - Web
```
Web service response time over past minute is significa...
```
Hide | Show more

Log anomaly - Abnormal behavior found in component: qotd-rating

Templates | Attach template logs
```
Unknown_error: 4
[<NUM>] <TXT> <*> <*> <NUM>: 9
[<NUM>]...
```
Hide | Show more

Close

### Action

- Click on **Show more**

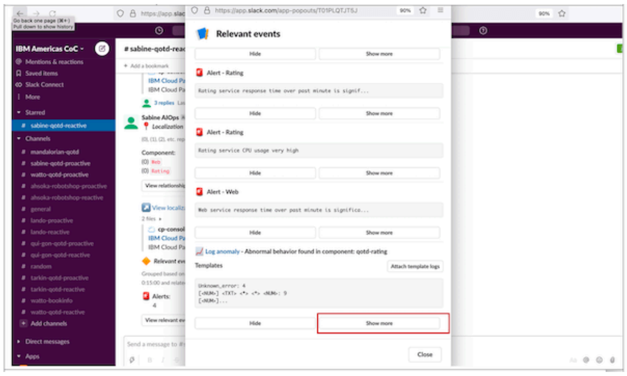
### Narration

Watson AIOps gives us additional context on the anomaly. In this case, the “unknown_error” anomaly is telling us that Watson AIOps has never seen this type of log before (hence the “unknown”) and that the log message indicates there is some type of error. Watson AIOps is not only looking at the statistical frequency of the type of log, but it is also using Natural Language Processing to analyze the content of the log message to give additional context (in this case that there’s likely an error).

Watson AIOps also explains why the anomaly was flagged. It expected to see zero (0) of this type of log, but it actually saw four (4).

Now we know that there is an unfamiliar log coming from the Rating service, and it’s indicating an error.

This further reinforces what we saw with the alerts - it looks like the Rating service is likely the root cause of this problem.



Relevant events

Alert - Rating

```
Rating service response time over past minute is signif...
```

Alert - Rating

```
Rating service CPU usage very high
```

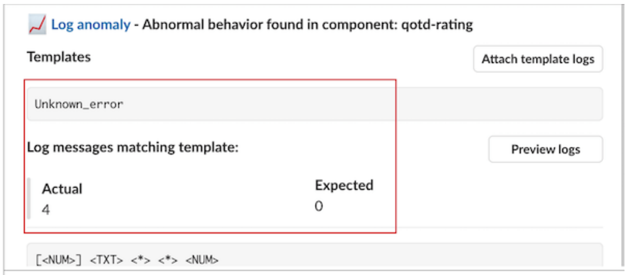
Alert - Web

```
Web service response time over past minute is significa...
```

Log anomaly - Abnormal behavior found in component: qotd-rating

Templates

```
Unknown_error: 4
[<NUM>] <TXT> <*> <*> <NUM>: 9
[<NUM>]...
```



Log anomaly - Abnormal behavior found in component: qotd-rating

Templates

```
Unknown_error
```

Log messages matching template:

Actual
4

Expected
0

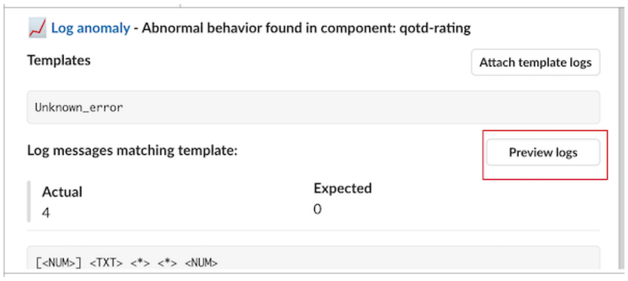
```
[<NUM>] <TXT> <*> <*> <NUM>
```

### Action

- For the ‘Unknown_error’ anomaly, click on **Preview logs**

### Narration

We can preview the log messages that caused Watson AIOps to find the anomaly.



Log anomaly - Abnormal behavior found in component: qotd-rating

Templates

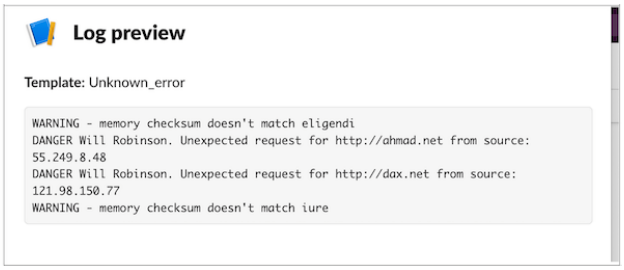
```
Unknown_error
```

Log messages matching template:

Actual
4

Expected
0

```
[<NUM>] <TXT> <*> <*> <NUM>
```



Log preview

Template: Unknown_error

```
WARNING - memory checksum doesn't match eligendi
DANGER Will Robinson. Unexpected request for http://ahmad.net from source:
55.249.8.48
DANGER Will Robinson. Unexpected request for http://dax.net from source:
121.98.150.77
WARNING - memory checksum doesn't match iure
```

# 4 - Getting resolution recommendations

## 4.1 - Watson AIOps searches similar incidents for recommendations

**Actions**

- Close the log anomaly pop-up
- Click on the **Search similar incidents** button
- Type ‘rating’ into the search box

**Narration**

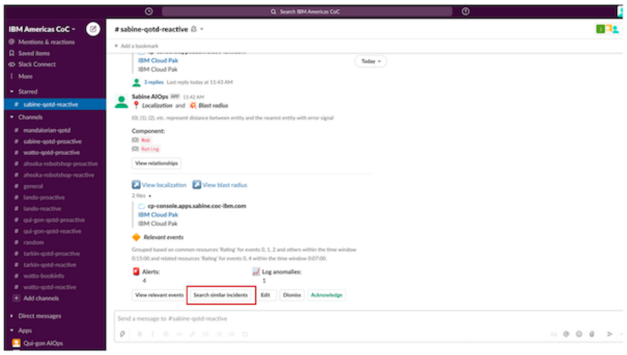
Now that we understand a bit more about what’s going on, we need to focus on our main goal: resolving the incident as quickly as possible.

Watson AIOps also brings in similar incidents and information regarding how they were resolved. This enhances operational efficiency by leveraging institutional knowledge that may be time consuming to find otherwise.

Since the alerts seem to point to the Rating service as having the memory and CPU increases, we’ll search for incidents that happened with that service.

Watson AIOps found two similar incidents. The first one seems like what we’re experiencing: Rating service is overheating and slowing. Using Natural Language Processing, Watson AIOps went through the comments on the incident and highlighted key relevant information for us, in this case that the incident was resolved by running a runbook.

If we can use the same runbook to resolve the current incident it will make our job that much easier and restore service even faster!



IBM Cloud Pak
IBM Cloud Pak

3 replies Last reply today at 11:43 AM

Sabine AIOps APP 11:42 AM

Localization and Blast radius

[0], [1], [2], etc. represent distance between entity and the nearest entity with error signal

Component:
[0] Web
[0] Rating

View relationships

View localization View blast radius

2 files

cp-console.apps.sabine.coc-ibm.com
IBM Cloud Pak
IBM Cloud Pak

Relevant events

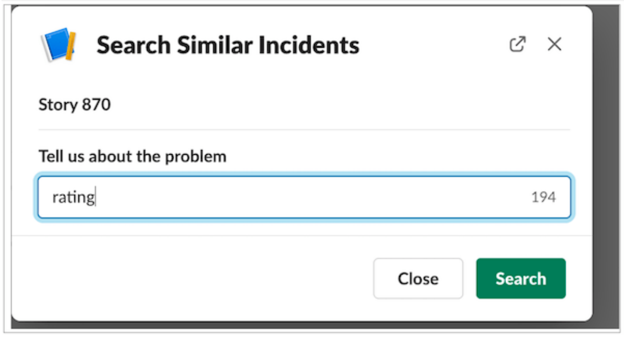
Grouped based on common resources 'Rating' for events 0, 1, 2 and others within the time window 0:15:00 and related resources 'Rating' for events 0, 4 within the time window 0:07:00.

Alerts:
4

Log anomalies:
1

View relevant events | Search similar incidents | Edit | Dismiss | Acknowledge

Send a message to #sabine-qotd-reactive

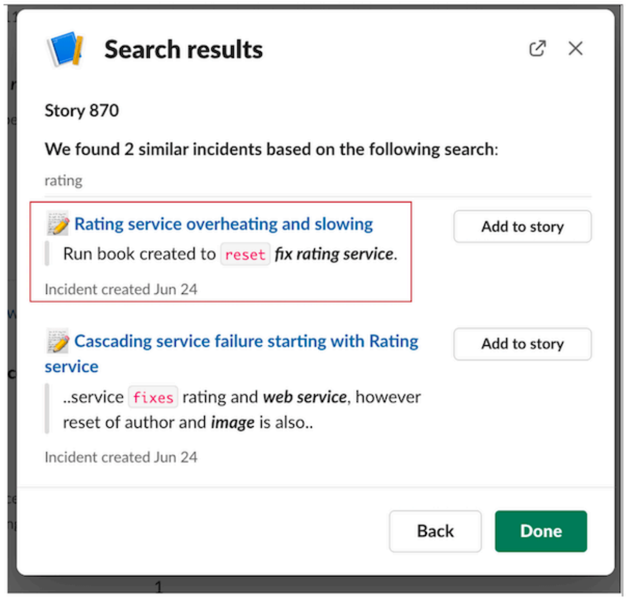


Search Similar Incidents

Story 870

Tell us about the problem

rating 194



Story 870

We found 2 similar incidents based on the following search:

rating

**Rating service overheating and slowing**

> Run book created to `reset` *fix rating service*.

Incident created Jun 24

**Cascading service failure starting with Rating service**

> ..service `fixes` rating and ***web service***, however reset of author and ***image*** is also..

Incident created Jun 24

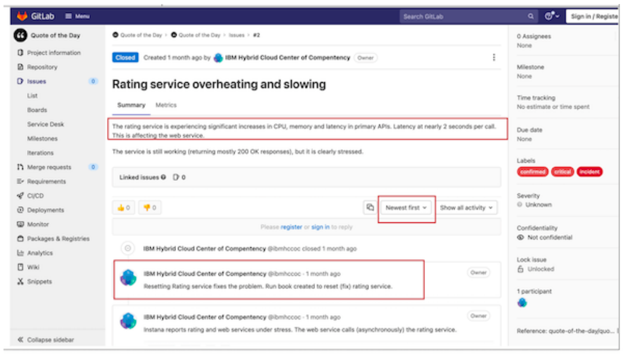
## Actions

- Click on the link for the first similar incident **Rating service overheating and slowing**
- It will open a window in GitLab
- If needed, sort comments **newest first**

## Narration

The description for this incident sounds a lot like what we are observing right now. The Rating service is stressed, with increases in CPU, memory, and latency. It's also affecting the Web service.

The comment says there is a runbook that resolved this issue. This is very useful information that may have taken a long time to find out just by searching or asking colleagues. Watson AIOps helped us find this info very quickly.



Created 1 month ago by IBM Hybrid Cloud Center of Compentency

Rating service overheating and slowing

The rating service is experiencing significant increases in CPU, memory and latency in primary APIs. Latency at nearly 2 seconds per call. This is affecting the web service.

The service is still working (returning mostly 200 OK responses), but it is clearly stressed.

Linked issues 0

Please register or sign in to reply

IBM Hybrid Cloud Center of Compentency @ibmhccoc closed 1 month ago

IBM Hybrid Cloud Center of Compentency @ibmhccoc · 1 month ago
Resetting Rating service fixes the problem. Run book created to reset (fix) rating service.

IBM Hybrid Cloud Center of Compentency @ibmhccoc · 1 month ago
Instana reports rating and web services under stress. The web service calls (asynchronously) the rating service.

0 Assignees: None
Milestone: None
Time tracking: No estimate or time spent
Due date: None
Labels: confirmed, critical, incident
Severity: Unknown
Confidentiality: Not confidential
Lock issue: Unlocked
1 participant
Reference: quote-of-the-day/quo...

# 5 - Fixing the problem and restoring service

## 5.1 - View Event Manager to determine the root cause and fix the problem

**Actions**

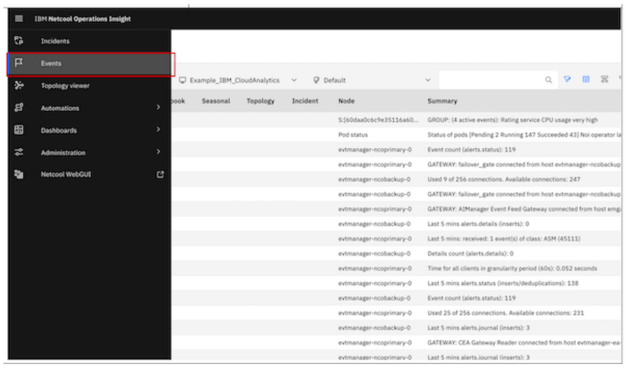
- Navigate to **Event Manager** (formally known as NOI)
- Click on **Events** in the sidebar menu
- The top event should show a red circle (**critical**)
- Click on the **Down** arrow to expand it

**Narration**

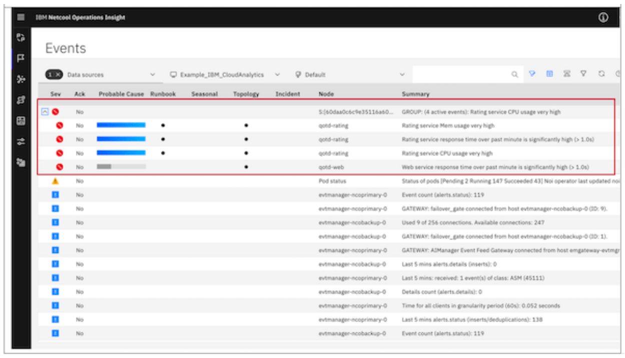
This is the Event Manager component of Watson AIOps. Here's the event group that we're working on. There are four events grouped together – these are the same ones we previewed earlier from the Slack notification. There's additional detail here, including the likelihood of each event being the root cause of the overall issue and if there is a runbook associated with the event.

There are three events that occurred on the Rating service. They're all rated as 99% likely to be the root cause - meaning that the Rating service is the source of the problem.

It also shows us that there's a runbook associated with these events, which is what was mentioned in the GitLab incident comment.



| ook | Seasonal | Topology | Incident | Node | Summary |
|---|---|---|---|---|---|
| | | | | S:[60daa0c6c9e35116a60... | GROUP: (4 active events): Rating service CPU usage very high |
| | | | | Pod status | Status of pods [Pending 2 Running 147 Succeeded 43] Noi operator la |
| | | | | evtmanager-ncoprimary-0 | Event count (alerts.status): 119 |
| | | | | evtmanager-ncoprimary-0 | GATEWAY: failover_gate connected from host evtmanager-ncobackup |
| | | | | evtmanager-ncobackup-0 | Used 9 of 256 connections. Available connections: 247 |
| | | | | evtmanager-ncobackup-0 | GATEWAY: failover_gate connected from host evtmanager-ncobackup |
| | | | | evtmanager-ncoprimary-0 | GATEWAY: AIManager Event Feed Gateway connected from host emg |
| | | | | evtmanager-ncobackup-0 | Last 5 mins alerts.details (inserts): 0 |
| | | | | evtmanager-ncoprimary-0 | Last 5 mins: received: 1 event(s) of class: ASM (45111) |
| | | | | evtmanager-ncobackup-0 | Details count (alerts.details): 0 |
| | | | | evtmanager-ncoprimary-0 | Time for all clients in granularity period (60s): 0.052 seconds |
| | | | | evtmanager-ncoprimary-0 | Last 5 mins alerts.status (inserts/deduplications): 138 |
| | | | | evtmanager-ncobackup-0 | Event count (alerts.status): 119 |
| | | | | evtmanager-ncoprimary-0 | Used 25 of 256 connections. Available connections: 231 |
| | | | | evtmanager-ncobackup-0 | Last 5 mins alerts.journal (inserts): 3 |
| | | | | evtmanager-ncoprimary-0 | GATEWAY: CEA Gateway Reader connected from host evtmanager-ea- |
| | | | | evtmanager-ncoprimary-0 | Last 5 mins alerts.journal (inserts): 3 |



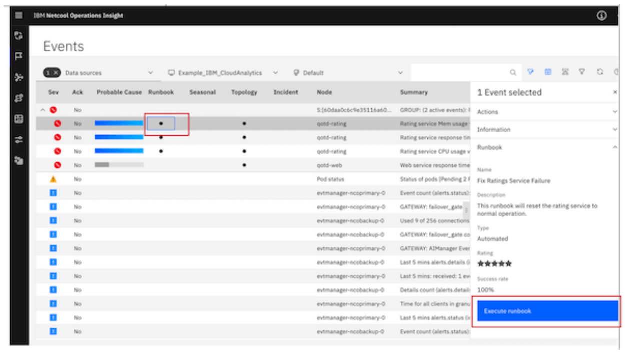
| Sev | Ack | Probable Cause | Runbook | Seasonal | Topology | Incident | Node | Summary |
|---|---|---|---|---|---|---|---|---|
| ⊘ | No | | | | | | S:[60daa0c6c9e35116a60... | GROUP: (4 active events): Rating service CPU usage very high |
| ⊘ | No | ▬ | ● | | ● | | qotd-rating | Rating service Mem usage very high |
| ⊘ | No | ▬ | ● | | ● | | qotd-rating | Rating service response time over past minute is significantly high (> 1.0s) |
| ⊘ | No | ▬ | ● | | ● | | qotd-rating | Rating service CPU usage very high |
| ⊘ | No | ▬ | | | ● | | qotd-web | Web service response time over past minute is significantly high (> 1.0s) |
| ⚠ | No | | | | | | Pod status | Status of pods [Pending 2 Running 147 Succeeded 43] Noi operator last updated noi |
| ■ | No | | | | | | evtmanager-ncoprimary-0 | Event count (alerts.status): 119 |
| ■ | No | | | | | | evtmanager-ncoprimary-0 | GATEWAY: failover_gate connected from host evtmanager-ncobackup-0 (ID: 9). |
| ■ | No | | | | | | evtmanager-ncobackup-0 | Used 9 of 256 connections. Available connections: 247 |
| ■ | No | | | | | | evtmanager-ncobackup-0 | GATEWAY: failover_gate connected from host evtmanager-ncobackup-0 (ID: 1). |
| ■ | No | | | | | | evtmanager-ncoprimary-0 | GATEWAY: AIManager Event Feed Gateway connected from host emgateway-evtmgr |
| ■ | No | | | | | | evtmanager-ncobackup-0 | Last 5 mins alerts.details (inserts): 0 |
| ■ | No | | | | | | evtmanager-ncoprimary-0 | Last 5 mins: received: 1 event(s) of class: ASM (45111) |
| ■ | No | | | | | | evtmanager-ncobackup-0 | Details count (alerts.details): 0 |
| ■ | No | | | | | | evtmanager-ncoprimary-0 | Time for all clients in granularity period (60s): 0.052 seconds |
| ■ | No | | | | | | evtmanager-ncoprimary-0 | Last 5 mins alerts.status (inserts/deduplications): 138 |
| ■ | No | | | | | | evtmanager-ncobackup-0 | Event count (alerts.status): 119 |

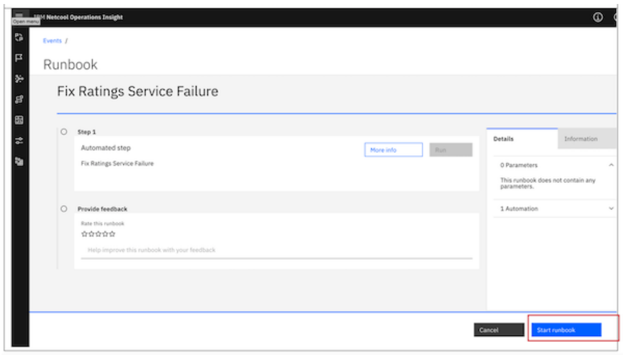
**Actions**

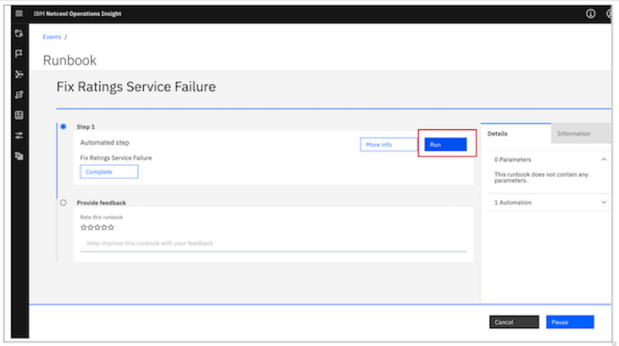
- Click on the second row to select it (**Summary** column: **Rating service Mem usage very high**)
- Click on **Execute runbook**
- Click on **Start runbook**
- Click on **Run**

**Narration**

Now we'll execute the runbook that should resolve the underlying failure of the Rating service. That should fix the problem.


IBM Netcool Operations Insight
Events
1 Event selected
Actions
Information
Runbook
Fix Ratings Service Failure
This runbook will reset the rating service to normal operation.
Automated
100%
Execute runbook





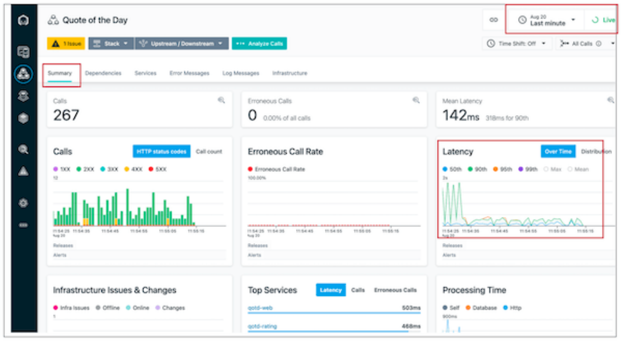
## Actions

- Navigate to **Instana**
- Click on the **Summary** tab
- Set the time period to **last 15 minutes**, and click on the **Live** button (you will be showing the latency before, during, and after the incident)

## Narration

To check that the runbook is fixing the problem, we can look at the health metrics in Instana. We'll set the time period so we can see data from when the incident began until now.

We can see that the latency of the application started increasing when the problem began. Now that we've run the runbook, latency is going back down. This confirms that the runbook was successful.



# 6 - Closing out the incident

## 6.1 - Provide feedback on the runbook and close incident

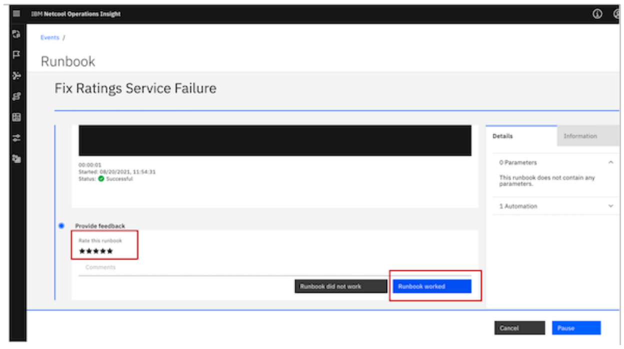
**Actions**

- Go back to the Runbook screen
- If you navigated away from it, go back with these steps:
- Automations -> Runbooks -> click on the three dots at the far right for the **Reset Rating Service** runbook -> Choose **View history** -> Click on **Resume on most recent**
- Scroll down to the bottom where there are stars/ratings

**Narration**

We can rate the runbook to preserve the information that it was successful – this will be helpful for the next person who comes along looking at this runbook, as well as feedback to the author of the runbook.

This enables a collaborative approach to organically improve incident resolution over time. As the system matures it can evolve into an increasingly automated self-healing system.



Runbook

Fix Ratings Service Failure

00:00:01
Started: 08/20/2021, 11:54:31
Status: Successful

0 Parameters

This runbook does not contain any parameters.

1 Automation

**Actions**

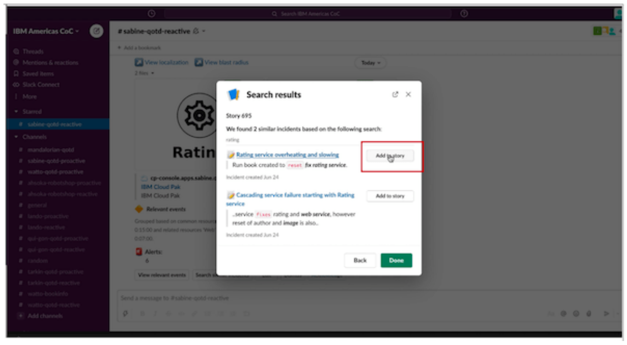
- Navigate to **Slack**
- Add details about the resolution by clicking the **Add to story** button

**Narration**

Now we are ready to close out the incident.

We found the runbook because it was mentioned in the similar incident, so we want to preserve that information. To do so, we can click add to story and Watson AIOps adds a comment with the information about the similar incident.

This keeps all the information in one place so it's easy to find later or if another similar incident occurs in the future.



Search results

Story 695

We found 2 similar incidents based on the following search:

rating

Rating service overheating and slowing

Run book created to `reset` fix rating service.

Incident created Jun 24

Cascading service failure starting with Rating service

..service `Fixes` rating and web service, however reset of author and image is also..

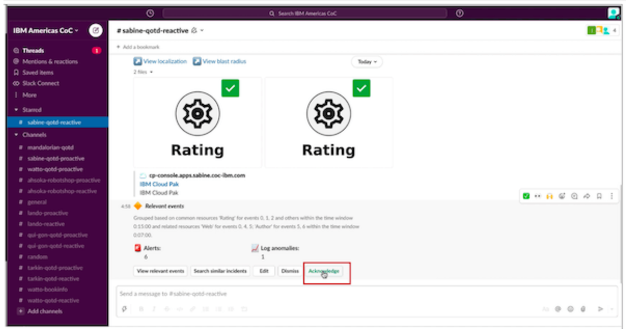
Incident created Jun 24

## Actions

- At the bottom of the story notification, click on **Acknowledge**
- Choose **Keep in current channel** and **Closed**
- Optionally add a comment, such as 'Executed runbook to fix issue,' and then click on **Acknowledge**

## Narration

Then we will acknowledge the incident and mark it as closed. Watson AIOps archives all the information from the incident in the comments of the original message. So we can go back to it whenever needed to see exactly what happened and how it was fixed.



cp-console.apps.sabine.coc-ibm.com

IBM Cloud Pak

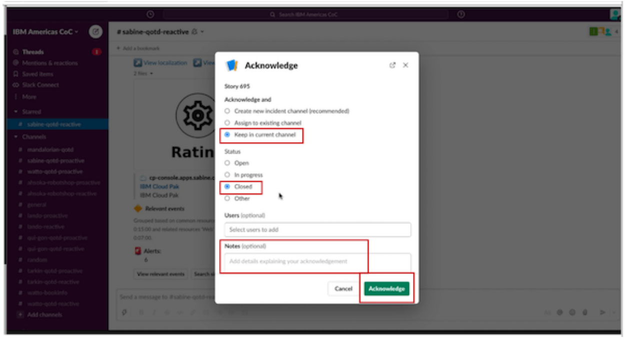
IBM Cloud Pak

Relevant events

Grouped based on common resources 'Rating' for events 0, 1, 2 and others within the time window 0:15:00 and related resources 'Web' for events 0, 4, 5; 'Author' for events 5, 6 within the time window 0:07:00.

Alerts: 6

Log anomalies: 1



Acknowledge

Story 695

Acknowledge and

○ Create new incident channel (recommended)

○ Assign to existing channel

● Keep in current channel

Status

○ Open

○ In progress

● Closed

○ Other

Users (optional)

Select users to add

Notes (optional)

Add details explaining your acknowledgement

## Summary

In this demo we have demonstrated how Watson AIOps enables you to avoid business impacting outages by applying AI to data gathered from your existing disparate tools. It helps you quickly determine the root causes of a failure and proactively alleviates outages – therefore minimizing the business impact of these IT issues on revenue or client experience.

The anomaly detection capabilities alerts you early to potential issues enabling the SRE take quick remedial actions. The intelligent event analytics examines logs, metrics, tickets and topology and provides a useful correlation that would otherwise be very challenging. Lastly, the AI-driven remediation based on recommendations of previous incidents accelerates problem resolution and significantly reduces the mean-time-to-repair.