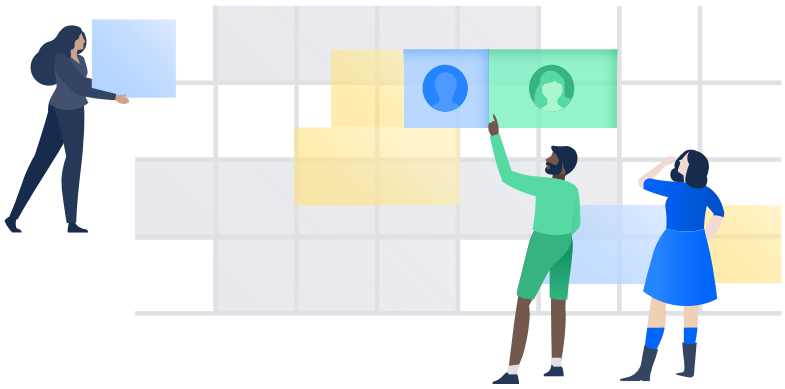
ATLASSIAN

Opsgenie

5 stages of incident management and how to improve them

Contents



Getting started

Simply put, effective incident management is an essential part of all enterprise business systems. Why? Because as tech tools and workflows become increasingly complex and interconnected, systems become increasingly vulnerable to unplanned downtime. Downtime that can hit any system at any time - with potential impact to both internal and external business operations. Costs for incidents are typically measured in tens, if not hundreds, of thousands of dollars per minute.

With such potential impact on the line, organizations are rapidly evolving incident response practices to ensure they can be managed as quickly and effectively as possible. This means taking a holistic approach to an incident, understanding how it evolves, and how to continually improve the resilience of systems. From an academic perspective, there are several opinions on how many stages are associated with a typical incident response workflow. While this may be different for varying organizations, we'll focus on the following five stages to represent the incident lifecycle:

1. Preparation
2. Detection & Alerting
3. Containment
4. Remediation
5. Analysis

Without consideration of each of these stages, organizations are exposing themselves to the risk that incidents will be mismanaged, resulting in unnecessary delays and associated costs. Below, we will look at each of these stages, and offer recommendations on practices that will help teams address incidents more efficiently.

Preparation

Even the most experienced IT professionals will say that Preparation is an essential, yet often overlooked, part of incident management. It's the stage where teams explore "what if" scenarios and then define processes to address them.

Leading organizations make a point of focusing on Preparation in the same way that athletes practice a sport. The goal is to build muscle memory around incident response so reactions can be faster.

> **Incident response methodologies typically emphasize preparation – not only establishing an incident response capability so that the organization is ready to respond to incidents, but also preventing incidents by ensuring that systems, networks, and applications are sufficiently secure."**
>
> NIST

Ideas for improvement

Always pack a jump bag.
A “jump bag” for incident responders is a repository of critical information that teams need to respond with the least amount of delay. By centralizing this material into a single location, teams have knowledge at their fingertips instead of needing to search for it. Depending on the structure of an organization’s teams and systems, this could include a variety of things:

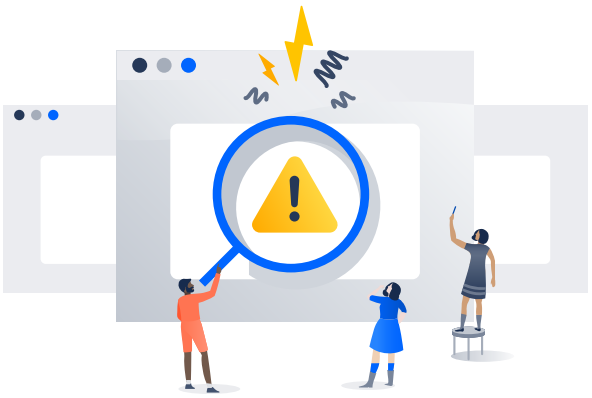
- Incident response plans
- Contact lists
- On-call schedule(s)
- Escalation policies
- Links to conferencing tools
- Access codes
- Policy documents
- Technical documentation & runbooks

- **Don’t run from Runbooks.**

Runbooks offer team members essential guidance on what steps to take in a given scenario. This is especially important for teams that work on rotational schedules and/or where a system expert may not be immediately available. Without runbooks in place, responders unfamiliar with a system are left spending cycles attempting to determine what steps need to be taken to begin remediation. A well maintained set of runbooks not only allows teams to respond faster, but also collectively builds a knowledge base that supports the continuous improvement of incident response practices.

- **Embrace chaos, promote stability.**

As a term “Chaos Engineering” seems like an oxymoron. It’s not. It’s the practice of experimenting with systems by knowingly injecting failure - in order to understand how systems can be built more robustly. An example of this is Chaos Monkey. Originally developed at Netflix, Chaos Monkey is a tool that tests network resiliency by intentionally taking production systems offline. While seemingly dangerous, the practice actually helps engineers continually test systems to ensure recoverability. Ultimately, Chaos Monkey helped teams at Netflix build a culture around system resiliency. With this success, many other organizations have followed suit in this practice.



Detection & Alerting

Incident Detection is not only focused on knowing that something is wrong, but also on how teams are notified about it. While these two may seem like separate processes, they are in fact very connected. The challenge is that while the proliferation of available IT monitoring tools has greatly improved the ability for teams to detect abnormalities and incidents - monitoring tools can also create “alert storms” or false positives that complicate the response process.

Top IT teams add a layer onto the monitoring process to ensure alerts are managed properly. This layer acts to centralize the alerting process, while also building in additional intelligence to the way alerts are delivered.

> Detection should lead to the appropriate response... This primarily call for the need to clearly identify and communicate the roles, responsibilities as well as the initia approach for incident handling. It should include determination of who shall identify the incident and determine its severity as a means to handle the incidenteffectively within the organisational context.”
>
> MITA

Ideas for improvement

- **Think outside the NOC.**

Historically, Network Operations Centers (NOCs) acted as the monitoring and alerting hub for large scale IT systems. The challenge is that a typical NOC engineer can be responsible for the triage and escalation of incidents from anywhere in the system. Modern incident management tools allow for this process to be streamlined significantly. By automating alert delivery workflows based on defined alert types, team schedules, and escalation policies, the potential for human error and/or delays can be avoided.

- **Aggregate, not aggravate.**

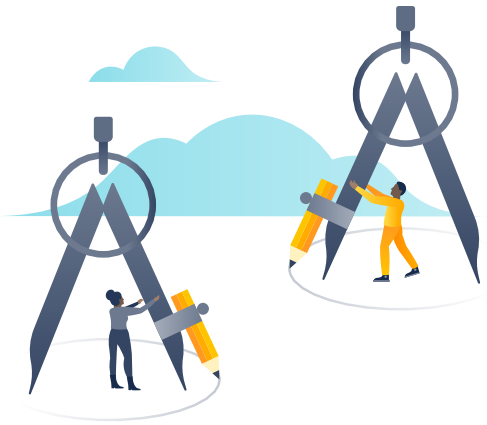
Nothing is worse than receiving a continual barrage of alerts coming from multiple monitoring tools. By centralizing the flow of alerts through a single tool, teams are able to better filter the noise so they can quickly focus on matters that need attention.

- **Knowledge = power.**

A basic alert conveys something is wrong, but it doesn't always express what. This causes unnecessary delays as teams must investigate and determine what caused it. By coupling alerts with the technical details of why it was triggered, the remediation process can begin faster.

- **Quis custodiet ipsos custodes?**

The latin phrase "Who's guarding the guards?" identifies a universal problem faced by all IT teams. This is because the monitoring tools they employ are as equally vulnerable to incidents and downtime as the systems they are designed to protect. Without a way to ensure monitoring tools are functioning properly, systems could easily go offline without notification. Holistic alerting processes ensure that both the systems, and the tools that monitor them, are continually checked for health.



Containment

The triage process for an IT incident is similar to processes deployed in medical fields. The first step is to identify the extent of the incident. Next, the incident needs to be contained in order to prevent the situation from getting worse. All actions taken in this phase should be focused on limiting and preventing any further damage from occurring.

> Short-term containment is not intended to be a long term solution to the problem; it is only intended to limit the incident before it gets worse."
>
> R. BEJTLICH, THE BROOKINGS INSTITUTE

Ideas for improvement

- **Stop the bleeding.**

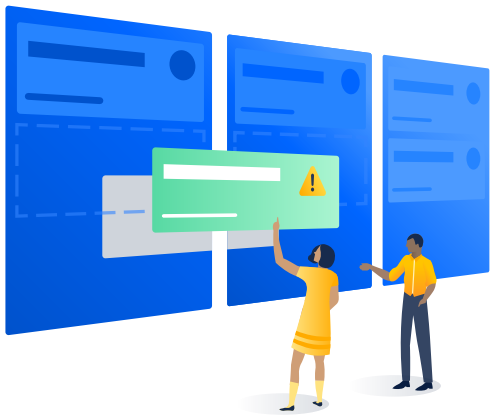
A triage doctor knows that they are risking greater harm if they get bogged down in attempting to resolve all situations as they arrive. Their focus is on short term actions that stabilize a patient enough to move them along to more acute care. In tech fields, containment actions focus on temporary solutions (isolating a network, regressing a build, restarting servers, etc.) that at a minimum limit the scope of the incident or, more ideally, bring systems back online. If incident management efforts focus purely on remediation, and not containment, an outage can be extended unnecessarily while a permanent solution is being found.

- **Don’t go it alone.**

Hero culture in IT teams is a dying philosophy. No longer is it fashionable to be the lone engineer who works endless evening and weekend hours because they are the only person who can bring systems back online. Instead, teams are working as just that, teams. Collaborating on issues because they understand that incidents can be resolved faster through shared knowledge. Conference lines, chat tools, and live video feeds therefore become essential elements of the incident management toolbox. These can quickly bring teams together so they can collaborate in real time. It’s also common for teams to integrate chat tools with incident management tools so incidents can be triggered, acknowledged, and resolved from a single platform.

- **Be transparent.**

The digital age makes seemingly endless amounts of information available at any time. In the midst of an IT meltdown, this can be an advantage - or disadvantage. If users are met with a service disruption, it's common for the incident to be made public in short order. To stay ahead of this, teams should have an incident communication plan in place. The goal is to build trust with customers by publicly acknowledging that a disruption is taking place, and to ensure them that steps are being taken to resolve it. Tools like Twitter, StatusPage, and user forums are great places to share this information. Importantly, this process should be designed to continue through the remediation and analysis phases to further grow trust with users that may otherwise abandon a system.



Remediation

Closely tied to Containment is Remediation. Here is where long-term solutions are implemented that ensure the incident has been addressed completely and effectively. Where in Containment, the goal may be to bring systems back online, in Remediation the goal shifts to understanding what caused the problem and how it can be corrected to prevent similar incidents from occurring in the future.

> Prior to full system recovery, remediation efforts should be performed to fix the source of the problem. The final stage of recovery is to not just restore the system to where it was, but rather to make it better and more secure. The system should have the same operational capabilities, but it also should protect against what caused the incident in the first place."
>
> US DEPARTMENT OF HOMELAND SECURITY

Ideas for improvement

- **Cynefin.**

A decision making framework, Cynefin (pronounced “KUN-iv-en”) provides a structured way to approach problems that helps incident responders determine the best course of action based on the nature of the problem itself. Depending on the type of incident (simple, complex, complicated, chaotic), an approach to solving it can be defined.

  - Does the incident have a known cause and solution?
  - Do I need to involve additional people to help address an incident?
  - Is there time to probe the problem to identify the best response, or does the situation require immediate action?

- **Automate much?**

Chat tools have become a defacto tool for organizations to improve communication and collaboration. Yet chat tools have also evolved far past simply enabling teams to send messages. The software development team at GitHub pioneered the evolution of chat tools when they released the open source tool, Hubot. Hubot allows users to trigger actions and scripts directly from a chat environment. This allows teams to simplify operations by creating bots that automate processes (initiating a server restart, deploying a snippet of code, etc).

Analysis

Incident management workflows don't end once the dust has settled and systems have been restored. Now begins one of the most important phases of the incident management lifecycle: Analysis. The intent of a "postmortem" analysis is to clearly understand both the systemic causes of an incident along with the steps taken to respond to it.

From here, leading teams work to identify improvement opportunities around the systems and the processes defined to maintain them. By evaluating this information, teams can develop new workflows that support higher system resilience and faster incident response.

> The (post-incident analysis) should be written in a form of a report to provide a play-by-play review of the entire incident; this report should be able to answer the: Who, What, Where, Why, and How questions that may come up during the lessons learned meeting. The overall goal is to learn from the incidents that occurred within an organization to improve the team's performance and provide reference materials in the event of a similar incident."
>
> SANS INSTITUTE

Ideas for improvement

- **Learn from failure.**

Overwhelmingly, IT teams will say that they only take the time to review "major outages." While this is a good start, it often overlooks smaller incidents that may have a lingering impact. A detailed postmortem report may not be necessary for all incidents, but a brief review of the details should always be done. This way, awareness of a situation supports the advancement of communal knowledge and continuous improvement.

- **There is no root cause!**

Or is there? When analyzing an incident, it is rare that a single identifiable "root" cause can be named. According to the Cynefin model, these would fall into the category of "simple" incidents where the cause and necessary response are known and repeatable. It's rarely that easy. Often systems are far too complex and interdependent to define a single root cause of an incident. Even if the root cause seems apparent (say a keystroke error that crashes an application), there is usually cause to understand what external factors may have allowed the application to crash (or not prevented it).

- **Be blameless.**

The goal of every postmortem should be to understand what went wrong and what can be done to avoid similar incidents in the future. Importantly, this process should not be used to assign blame. That's because teams that focus on the "who" and not the "what," let emotions pull the analysis away from truly understanding what happened.

In Summary

In modern IT environments, change is the only constant. This means systems will continually be stressed in new and different ways. Teams that understand this, also understand that it's not a matter of if - but when - systems will fail. Taking steps to prepare for these failures should be recognized as a critical element of ongoing success, and integrated into the DNA of engineering teams.

About Opsgenie

Opsgenie is a modern incident management platform for operating always-on services, empowering Dev & Ops teams to plan for service disruptions and stay in control during incidents. With over 200 deep integrations and a highly flexible rules engine, Opsgenie centralizes alerts, notifies the right people reliably, and enables them to collaborate and take rapid action. Throughout the entire incident lifecycle, Opsgenie tracks all activity and provides actionable insights to improve productivity and drive continuous operational efficiencies.

See Opsgenie in action

Get started for free today

Sign up

Resources

Alerting & Incident Management:

- Supporting custom alert properties
- How to enhance collaboration during an incident
- 5 Common incident response problems (and their solutions)

Cynefin

- The Cynefin Framework
- The Cynefin Framework video

ChatOps

- Slack > Opsgenie integration video
- ChatOps and Hubot at GitHub

Chaos Engineering

- http://principlesofchaos.org/
- Chaos Monkey

Post-incident Analysis

- Incident Tracking with Opsgenie

ATLASSIAN

Opsgenie