PagerDuty

The On Call Survival Guide

Best practices for being on call

Contents

SHARING OUR INTERNAL RESPONSE PROCESS 3

WHAT IS "ON-CALL"? 3

RESPONSIBILITIES 4

- Prepare 4
- Triage 5
- Fix 5
- Improve 5
- Support 6

NOT RESPONSIBILITIES 6

RECOMMENDATIONS 7

- Notification Method Recommendations 8

ON-CALL ETIQUETTE 9

ITERATE AND IMPROVE 10

- Alerting principles to follow 10

DON'T FEAR THE ON-CALL LIFE 11

Welcome to the on-call life

Being on-call can be a daunting experience for any new team member. There is only one thing worse than being woken up at 3am to discover that your systems are down — to wake up on your own at 8am and discover that your systems were down for 5 hours and nobody got the alert!

With the right approach, a culture of collaboration, infrastructure insight, and the right tools in place, being on-call doesn't have to be so bad.

Sharing Our Internal Response Process

Our internal incident response documentation is something we've built up over the last few years as we've learned from our mistakes. It details the best practices of our process, from how to prepare new employees for on-call responsibilities, to how to handle major incidents, both in preparation and after-work.

We'd like to share how we here at PagerDuty prepare our team members for going on-call. It is our hope that others will use the documentation as a starting point to formalize their own processes. In this guide, we'll talk about what being on-call actually means, what on-call responsibilities entail (and don't entail), and best practices for being on-call.

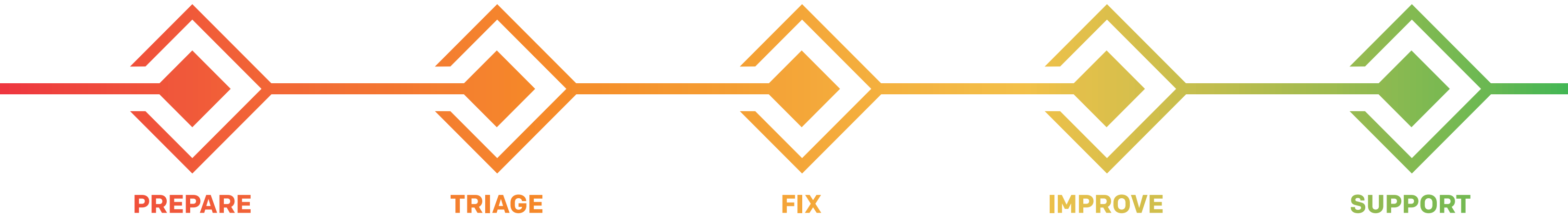
What is "On-Call"?

Being on-call means that you are able to be contacted at any time in order to investigate and fix issues that may arise for the system you are responsible for. For example, if you are on-call for a service at your organization, should any alarms be triggered for that service, you will receive an alert on your mobile device via email, phone call, push notification or SMS, providing you details on what's wrong and how to fix it. You're expected to take whatever actions are necessary to resolve the issue and return your service to a normal state.

On-call responsibilities extend beyond normal office hours and if you are on-call, you are expected to be able to respond to issues, even at 2am. This sounds horrible (and it sometimes can be), but this is what our customers go through, and is the exact problem that PagerDuty is trying to solve! PagerDuty exists to make on-call life less painful for everyone.

Responsibilities

Knowing exactly what your responsibilities are can make being on-call much more painless. Below are responsibilities as they relate to each step of the incident management process.



Prepare

For peace of mind, it's crucial that you're prepared with everything you need before going on-call.

Have your laptop and Internet with you (office, home, a MiFi, a phone with a tethering plan, a hotspot, etc).

Have a way to charge your MiFi.

Team alert escalation happens within 5 minutes. Be sure to set or stagger your notification timeouts accordingly.

Make sure PagerDuty can bypass your "Do Not Disturb" settings

Your environment should be set up and a current working copy of the necessary repos should be local and functioning.

Have your configured and tested environments on workstations.

Ensure your credentials for third-party services are current.

Understand how your organization handles serious incidents, as well as what the different roles and methods of communication are.

Be aware of your upcoming on-call time (primary, backup) and arrange swaps around travel, vacations, appointments etc.

Triage

When situations do arise, evaluate them and think broadly.

Acknowledge and act on incidents whenever you can (see the first "Not Responsibilities" point below)

Check Slack (or whichever chat platform you use) for current activity.

Ask yourself, does the alert and your initial investigation indicate a general problem or an issue with a specific service that the relevant team should look into? If it does not look like a problem you are the expert for, then escalate to the appropriate team.

Determine the urgency of the problem:

Is it something that should be worked on right now or escalated into a major incident?

Is it some tactical work that doesn't have to happen during the night? For example, if there's a disk utilization high water mark but there's plenty of space left and the trend is not indicating impending doom, snooze the alert until for a more suitable time, such as normal working hours and get back to fixing it then.

If you decide the work doesn't need to happen immediately, make sure to follow up and change your alerting so that you don't get woken by similar incidents in future.

Fix

You don't have to necessarily fix a problem on your own right away. Draw on the appropriate resources as needed.

You are empowered to dive into any problem and act to fix it.

Involve other team members as necessary: do not hesitate to escalate if you cannot figure out the cause within a reasonable timeframe or if the service/alert is something you have not tackled before.

If the issue is not very time sensitive and you have other priority work, create a ticket with appropriate severity to keep a track of it.

Never hesitate to escalate

Improve

Always keep an eye out for ways to improve your processes.

If a particular issue keeps happening or if an issue alerts often but turns out to be a preventable non-issue, consider improving it as a longer-term task.

This can include alerts for disks that fill up, logs that should be rotated, noisy alerts, etc.

If information is difficult to find, write it down. Constantly refactor and improve your knowledge base or documentation.

Support

Be courteous to your team.

When your on-call "shift" ends, let the next on-call know about issues that have not been resolved yet and other experiences of note.

If you are making a change that impacts the schedule, let others know since many people make arrangements around the on-call schedule well in advance.

Support each other: When doing activities that might generate plenty of "pages", such as maintenance, it is courteous to "take the pager" yourself while performing the actions that will page out. Take the pager from the on-call team by notifying them and scheduling an override for the duration.

Not Responsibilities

Just as it's important to know what your responsibilities are and what's expected of you, it's also important to know what's not expected.

You're NOT expected to....

BE THE FIRST TO ACKNOWLEDGE ALL OF THE INCIDENTS

Commuting and other necessary distractions are facts of life, and sometimes it's not possible to receive or act on an incident before it escalates.

And that's okay — that's what we have the backup on-call and schedule for.

FIX ALL ISSUES BY YOURSELF

No one knows everything and your entire team is here to help. There is no shame in not knowing something and much to be learned by escalating issues you are not certain about.

Service owners will always know more about how their stuff works. Measure twice, cut once – and it's often best to let the subject matter expert do the cutting.

Recommendations

If your team is starting its own on-call rotation, here are some recommendations from our operations team.

ALWAYS HAVE A BACKUP SCHEDULE

GET THE TEAM INVOLVED

INVITE TEAM MANAGERS

SHADOWING FOR NEW MEMBERS

5 MINUTE ESCALATION TIMEOUTS

OFF-CALL SUMMARIES

ALWAYS HAVE A BACKUP SCHEDULE

Yes, this means two people being on-call at the same time; however, it takes a lot of the stress off of the primary if they know they have a specific backup they can contact, rather than trying to chose a random member of the team. A backup shift should generally come directly after a primary shift. It gives chance for the previous primary to pass on additional context which may have come up during their shift. It also helps to prevent people from sitting on issues with the intent of letting the next shift fix it.

GET THE TEAM INVOLVED

The third-level of your escalation (after backup schedule) should probably be your entire team. This should hopefully never happen, but when it does, it’s useful to be able to get a hold of the next available person.

INVITE TEAM MANAGERS

Team managers can and should be part of your normal rotation. It gives better insight into what has been going on.

SHADOWING FOR NEW MEMBERS

New members of the team should shadow your on-call rotation during the first few weeks. They should get all incidents and should follow along with what you are doing. At PagerDuty, all new employees shadow the operations team for one week of on-call, but it’s useful to have new team members shadow your team rotations also.

5 MINUTE ESCALATION TIMEOUTS

We recommend you set your escalation timeout to 5 minutes. This should be plenty of time for someone to acknowledge the incident if they’re able to. If they’re not able to within 5 minutes, then they’re probably not in a good position to respond to the incident anyway.

OFF-CALL SUMMARIES

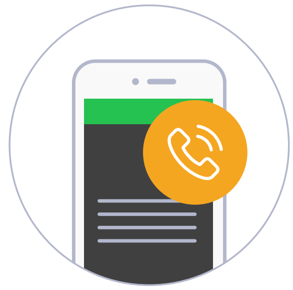
When going off-call, you should provide a quick summary to the next on-call team member about any issues that may come up during their shift. This can be a written report via email or a verbal summary.

Notification Method Recommendations

You are free to set up your notification rules as you see fit, to match how you would like to best respond to incidents. If you're not sure how to configure them, our operations team has some recommendations,

FIRST METHOD:
PUSH NOTIFICATIONS

Use Push Notification and Email as your first method of notification. Most of us have phones with us at all times, so this is a sensible first method and is usually sufficient.



SOMETHING STRONGER:
PHONE CALLS

Use Phone and/or SMS notifications for each minute after, until the escalation time. If Push didn't work, then it's likely you need something stronger, like a phone call. Keep calling every minute until it's too late. If you don't pick up by the third time, then it's unlikely you are able to respond, and the incident will get escalated away from you.

On-Call Etiquette

Being on on-call can definitely be rough. Here are some general guidelines to keep in mind to ensure a healthy relationship with your on-call team.

If the current on-call comes into the office at 12pm looking tired, it's not because they're lazy. They probably got paged in the night. Cut them some slack and be nice.

Don't acknowledge an incident out from under someone else. If you didn't get paged for the incident, then you shouldn't be acknowledging it. Add a comment with your notes instead.

If you are testing something, or performing an action that you know will cause a page, it's customary to "take the pager" for the time during which you will be testing. Notify the person on-call that you are taking the pager for the next hour while you test.

Never hesitate to escalate. Never feel ashamed to rope in someone else if you're not sure how to resolve an issue. Likewise, never look down on someone else if they ask you for help.

Always consider covering an hour or so of someone else's on-call time if they request it and you are able to. We all have lives which might get in the way of on-call time, and one day it might be you who needs to swap their on-call time in order to have a night out with your friend from out of town.

If an issue comes up during your on-call shift for which you got paged, you are responsible for resolving it. Even if it takes 3 hours and there's only 1 hour left of your shift. You can hand over to the next on-call if they agree, but you should never assume that's possible.

Iterate and Improve

Alerting principles to follow

Being woken up for something non-urgent is a sure-fire way to burn out your on-call team. Prevent burnout by making sure your alerts and incidents are relevant for your on-call team.

We manage how we get alerted based on a simple principle — an alert is something which requires a human to perform an action. Anything else is considered a notification, which is something that we cannot control, and for which we cannot perform any action to affect it. Notifications are useful, but they should not be waking people up under any circumstance.

If you're on-call and getting paged for something that doesn't need immediate attention, it's recommended that you adjust the alert appropriately. If you get alerted for something non-urgent, make sure to go back and adjust the alert so you don't get alerted again.

HIGH PRIORITY ALERTS

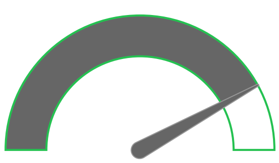
Anything that wakes up a person in the middle of the night should be immediately "human actionable." If it's not, then you should adjust the alert to not page at those times.

If you're setting up a new alert or notification, consider the chart below for how you might want to alert people.

Priority	Alerts	Response
High	High-Priority PagerDuty Alert 24/7/365.	Requires **immediate human action**.
Medium	High-Priority PagerDuty Alert during **business hours only**.	Requires human action within 24 hours.
Low	Low-Priority PagerDuty Alert 24/7/365.	Requires human action at some point.
Notification	Suppressed PagerDuty Event.	No response required. Informational only.

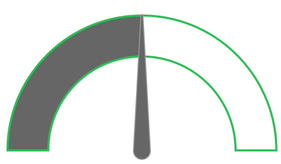
*Always be mindful of not creating new high-priority alerts if they don't require an immediate response.

EXAMPLES



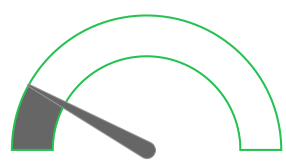
"Production service is failing for 75% of requests, automation is unable to resolve."

This would be a **High** priority page, requiring immediate human action to resolve.



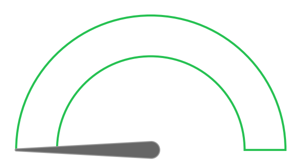
"Production server disk space is filling, expected to be full in 48 hours. Log rotation is insufficient to resolve."

This would be a **Medium** priority page, requiring human action soon, but not immediately.



"An SSL certificate is due to expire in one week."

This would be a **Low** priority page, requiring human action some time soon.



"A deployment was successful."

This would be a **Notification**, and should be sent as a suppressed event. It provides useful context should an incident occur, but does not require notifying a human.

Don't Fear the On-Call Life

Being on-call will inevitably be a fundamental expectation of you as a developer (if it isn't already), regardless of your organization's size or operational framework. It is now best practice for those who build services to also be accountable for the success of those services in production.

Keep in mind that with the proper preparation and a culture of collaboration, along with infrastructure insight, and the right tools in place, being on-call doesn't have to be so bad.

Being on-call should be a huge source of empowerment, not stress. By making on-call seamless through by implementing the best practices outlined above will position you to do the most impactful work of your career.

We hope that by sharing our internal processes of how we prepare our team members for going on-call, you're able to formalize your own processes and make being on-call easier for you and your team. Here's to making on-call be painless!

About PagerDuty

PagerDuty is the leading incident management platform for digital businesses. PagerDuty empowers developers, DevOps, IT operations and business leaders with the insight to intelligently respond to critical disruptions for exceptional customer experience. Over 8,000 small, mid-size and enterprise global customers such as Comcast, eHarmony, Slack and Lululemon use and trust PagerDuty to increase business response and efficiency. Headquartered in San Francisco, the company was recently included in the 2016 Deloitte Technology Fast 500, Inc. 500 and Forbes Cloud 100 lists.

PagerDuty