

OpsGenie to Spike parallel run plan

Run Spike next to OpsGenie for a short time, so you can confirm every alert reaches the right person before you switch.

UPDATED OCTOBER 2026

The plan in three phases

PHASE 1: OBSERVE

Send every alert to both tools. OpsGenie keeps paging people as it does today, and Spike posts incidents to a [Slack](#) channel only, so nobody gets a second page.

- Add the Spike webhook to each monitoring tool, next to the OpsGenie one. See [creating integrations and services](#).
- Give each Spike escalation policy a single Slack step for now. See [creating an escalation policy](#).

PHASE 2: PAGE FROM SPIKE

After a few days without a gap, start paging from Spike for one team or a few low-risk integrations. OpsGenie stays on as your backup.

- Add the [phone](#), [SMS](#), or [mobile app](#) steps to those escalation policies, and run [Test Escalation](#) on each one.
- Ask the people on-call to confirm they received their pages. Everyone should [set their alert preferences](#) first.

PHASE 3: SPIKE FIRST

Extend Spike paging to every team and integration, and turn off OpsGenie notifications for your responders so nobody gets duplicate pages. Keep OpsGenie receiving alerts, so you can still compare.

- Tell the team that pages now arrive from Spike.
- Keep doing the daily check until you have covered a full on-call rotation.

Then continue with the cutover steps in the [migration checklist](#).

How long to run both

Keep both running until you have covered a full on-call rotation. [HitPath](#), one of our customers, ran both tools for about a month before canceling OpsGenie.

The daily check

Compare yesterday's alerts in OpsGenie with the incidents in Spike.

- ☐ **Every alert matches.** Each OpsGenie alert has a matching Spike incident.
- ☐ **The right people were alerted.** The same responders were reached, on the same channels and at the same priority.
- ☐ **Acknowledge and resolve work.** Try them from each [channel your team uses](#).

☐ **Incidents close on their own.** Alerts that recover in your monitoring tool resolve in Spike.

☐ **Shift notifications arrive.** People are told when their on-call shift starts and ends.

Common gaps and how to fix them

WHAT YOU SEE	LIKELY CAUSE	WHAT TO DO
An alert in OpsGenie but no incident in Spike	The monitoring tool is not sending to the Spike webhook, or is sending to the wrong integration.	Check the webhook URL in the tool, and use the tool's test notification to send an alert.
The wrong person was alerted	A schedule <u>layer</u> , <u>slot</u> , or <u>policy step</u> is set differently from OpsGenie.	Compare the <u>on-call calendar</u> with your OpsGenie schedule, and check the order of the policy steps.
Too many incidents	Spike <u>groups incidents by title</u> , so a changing title creates new incidents.	Use <u>Title Remapper</u> to give repeated alerts the same title, and add an <u>alert routing rule</u> for noise you want to ignore.
An incident does not close	The tool is not sending a recovery event.	Enable the OK or resolved notification in the tool, or set a <u>Resolve Timer</u> on the integration.
Someone was not reached	Their phone or email is not verified, or Do Not Disturb blocked the call.	Ask them to <u>verify their contacts</u> and add Spike's <u>contact card</u> to their favorites.

When to switch

We suggest switching when all of these are true:

☐ You have covered a full on-call rotation.

☐ A week has passed with no gaps in the daily check.

☐ Everyone on call has acknowledged an incident in Spike at least once.

☐ The team knows the date and what changes.

Daily log

Note each day's check here, so you can see when a full on-call rotation is covered.

DATE	EVERY ALERT MATCHED	GAPS FOUND	FIXED

Need help with your migration?

Write to support@spike.sh, or book a call with our team at spike.sh/demo, and we will help you map your setup.