Build vs buy incident management: how to decide

Not sure whether to build your own incident management tool or buy one? This build vs buy incident management guide compares both on real 3-year costs, setup time and reliability.

UPDATED NOVEMBER 2026

TL;DR

Buying is usually the better choice in the build vs buy incident management decision. Alerting, on-call schedules and escalations are solved problems, and vendors have spent years making them reliable. Building only makes sense if you sell incident tools yourself or if strict rules stop you from using an outside vendor. The first version is quick to build, but keeping it running costs far more over time.

Why do teams consider building their own incident management tool?

1. It fits your own workflows and terms. An in-house tool can follow your incident process exactly. It can use your own severity labels and incident stages, and it can connect to your internal tools.

2. There's no per-user bill. The engineers who build it are already on the payroll, so there's no subscription to approve. The cost doesn't disappear, though. It moves into engineering hours and hosting, which the next section covers.

3. You control the features and the data. You decide what gets built and when, with no feature request waiting on a vendor's roadmap. You also decide where incident data is stored, which matters for companies with strict data residency rules.

4. AI makes the first version quicker to write. AI coding tools can produce a working Slack bot or alert handler in days rather than weeks. That makes building look cheaper than it used to. The first version is still the smallest part of the work, though.

What are the hidden costs of building in-house?

Development time and resources

A basic version that opens a Slack channel and alerts one person can be ready in a few weeks. A tool people trust at 3 AM takes several months of building and testing. After that comes the ongoing work of bug fixes, new features and questions from the people who use it. That time comes from engineers who would otherwise work on your product.

Complexity over time

Alerting someone is only one part of incident management. A complete tool also needs:

- Escalation policies that call a backup when nobody responds
- On-call schedules with rotations, overrides and time zones
- Alerts by phone call, SMS, push, Slack and email
- Integrations with monitoring tools like Prometheus, Datadog or AWS CloudWatch

Alert delivery is the hardest of these to get right. A phone call can go unanswered, a carrier can filter an SMS and the calling provider can have its own outage. An in-house tool needs the same safeguards before anyone can rely on it at night.

Incident risks for your incident tool

What happens if the tool you built to manage incidents has an incident? If it runs on the same servers or login system as your product, it can go down in the same outage. The team then has two problems at once and no alerts for either.

Keeping the tool up means hosting it separately. It also needs its own deployment process, its own monitoring and someone on call for the tool itself.

Opportunity cost

Every hour spent on an internal incident tool is an hour not spent on the product your customers pay for. For a company that doesn't sell incident tooling, that trade is rarely worth it. The engineers best placed to build reliable alerting are usually the same ones your roadmap depends on.

The tool becomes a second product

Once people rely on it, an in-house tool becomes a product of its own. It needs an owner, updates, documentation and support. When the engineer who built it moves on, the knowledge often leaves with them if not documented well. Changes start to feel risky, so the tool stays as it is while the services around it keep changing.

How much does it cost to build vs buy incident management software?

Building usually costs several times more than buying over 3 years. Buying is a predictable monthly fee per user. Building means paying engineers up front, then paying for servers, security and maintenance for as long as the tool runs.

Here's a 3-year example for a team of 20:

Cost	Build in-house	Buy (Spike Business, 20 users)
First version	$75,000	None (set up in hours)
Maintenance per year	$45,000	Included
Servers, phone and SMS per year	$5,000	Included
Subscription per year	None	$3,360
3-year total	$225,000	$10,080

Buying Spike saves this team about $214,920 over 3 years, which is roughly 95% less than building. Spike prices are list prices as of October 2026.

What does the first version cost?

An incident tool needs more than a Slack bot. The first version must route alerts to the right team, follow escalation rules, make phone calls and send SMS, and keep a timeline for each incident. Building all of that takes months.

A simple way to estimate the cost is engineers × months × monthly salary. For example, one engineer earning $150,000 a year spends 6 months on it. That comes to about $75,000.

Big software projects also often grow beyond their plan. A McKinsey study of large IT projects found they ran 45% over budget on average and delivered 56% less value than expected.

What does it cost to keep it running?

The costs don't stop after launch. Three of them come back every year.

- **Maintenance:** bug fixes, security updates and new features usually take a quarter to a third of one engineer's time. At 30% of a $150,000 salary, that's about $45,000 a year.
- **Servers:** the tool has to run apart from your main product, so it keeps working when your product goes down. That means separate hosting, backups and monitoring, plus a phone and SMS provider like Twilio. This example budgets about $5,000 a year for it.
- **Security:** incident data often describes weak spots in your systems. That makes security reviews and strict access controls necessary, and both add cost.

Outside tools keep changing as well. When Slack or Twilio updates its API, your team has to fix the connection.

What does buying cost?

Buying is a subscription priced per user each month. Spike costs $7 per user a month on the Standard plan and $14 on the Business plan, which includes unlimited phone and SMS alerts. For a team of 20 on Business, that's $280 a month or $3,360 a year. Maintenance, servers and phone delivery are all part of the subscription.

Why is buying usually the better choice?

1. Fast setup. A bought tool can be running in hours. Connecting a monitoring tool, adding an on-call schedule and setting an escalation policy can usually be done in an afternoon.

2. Reliability that doesn't depend on your servers. A vendor runs the tool apart from your product, so it keeps working when your systems go down. A public status page shows how well it has held up. Spike's status page tracks phone call and SMS alerts as separate parts.

3. Continuous improvements. New features, security fixes and performance updates arrive without any work from your team.

4. Ready-made integrations. Spike connects to 180+ monitoring tools, cloud platforms and error trackers, including Prometheus, Datadog and Sentry. It also works with Slack, Microsoft Teams, Zendesk and Freshdesk from day one.

5. More time for your own product. Engineers stay on the work customers pay for, while the vendor's team looks after the incident tool.

What are the downsides of buying?

Subscription cost

A subscription is a bill that arrives every month and grows with every user. It helps to compare what each plan includes, since phone and SMS alerts are often limited on cheaper plans. Spike's Standard plan caps phone and SMS alerts at 100 a month, while Business has no limit.

Adapting to the vendor's setup

A bought tool comes with its own way of working. Your team might need to adapt to how the vendor handles severities, rotations or postmortems. Before choosing one, it's worth checking whether you can set your own severity levels, rotation patterns and escalation rules. Settings like these usually cover what people think they need to build.

Vendor lock-in

Once your team relies on a tool, leaving it gets harder. Moving incident history and settings to another tool takes work. If you need a niche workflow the tool doesn't support, you also wait on the vendor's roadmap instead of building it yourself. A tool with an open API, webhooks and an easy way to export data softens both problems, so these are good things to check early. In Spike, a REST API and webhooks let you add your own steps on top.

The vendor shuts down the product

It does happen. Atlassian stopped selling OpsGenie in June 2025, and support ends in April 2027. Any data that hasn't been moved by then gets deleted. OpsGenie customers now have to migrate whether they like it or not. And if you happen to be an OpsGenie user looking to migrate, we've got you covered with OpsGenie migration resources.

When does it make sense to build your own incident management tool?

1. Your needs are unique. Some workflows are so specialized or so tied to your own systems that no ready-made or open-source tool can adapt to them. Strict rules are one example: some defence, government and banking systems run on networks with no internet access or must keep all data in one country. Consider checking these rules with your security team first, since they're sometimes assumed rather than written down.

2. You operate at a very large scale. Companies the size of Google or Meta run incidents across thousands of engineers and services. At that size, per-user pricing adds up fast and a custom tool built around their own systems can work better.

3. You have a dedicated team to run it. Large companies sometimes have DevOps or SRE teams whose full-time job is internal tools. With that team in place, the tool always has an owner and the work doesn't take time away from your product.

4. Cost isn't a concern. Building only makes sense on budget grounds if you can pay for years of maintenance, updates and redesigns, not just the first version. With that budget in place, building gives you full control over the code and data.

Build vs buy incident management: a side-by-side comparison

Factor	Build in-house	Buy
Initial cost	Low cash outlay, paid in engineering hours	A monthly subscription per user
Maintenance effort	High and ongoing: bug fixes, API changes and feature requests	Handled by the vendor
Customization	Fully customizable	Configurable within the vendor's settings
Time to value	Several months for a reliable version	Hours to days
Scalability	Needs more development as you grow	Grows with your plan
Reliability	Depends on your own hosting and on-call cover	Runs apart from your product
3-year cost for 20 people	About $225,000 in engineering time and hosting	About $10,080 on Spike Business
Integrations	Each one built and maintained by your team	Ready-made for monitoring and chat tools
When the builder leaves	The knowledge often leaves too	Documented product and vendor support

Key considerations before you decide

Is incident management part of your core business?

Incident tooling is core only when it's what your company sells, like an alerting or monitoring product. For everyone else, it supports the product rather than being the product. Alerting the on-call engineer, opening a Slack channel and updating a status page are standard jobs that ready-made tools already do well. Building only fits strict needs no vendor supports, such as an air-gapped network with no internet access or a regulatory framework unique to your company.

How soon do you need it?

A bought tool can be set up in hours or days. Building one can take 6 to 12 months of design, coding and testing. Until it's ready, your team handles incidents without a proper workflow.

What will it cost to keep it running?

An in-house tool is never really finished. Each time your chat app, monitoring tools or cloud setup changes, the tool's connections can break and someone has to fix them. If the engineer who built it leaves, that work falls on people who never planned for it. A subscription is easier to plan for, and it usually costs far less than the engineering time a built tool needs each year.

How reliable does it need to be?

Your incident tool has to keep working when your main systems go down. A vendor runs it on separate servers with backups built in. If you build it, its reliability depends on how much time your own team can give it.

Which tools does it need to connect to?

A bought tool comes ready to connect with monitoring tools, ticketing systems, deployment pipelines and chat apps. Building means writing and maintaining a new connector every time your team adopts a new tool.

Spend your engineering hours where they count

The build vs buy incident management question usually comes down to one thing: where your engineers' time is best spent. Unless your company sells incident tooling, that time is worth more on your product than on an internal tool. A practical next step is to list your must-have features and run the cost example above with your own numbers.

If buying comes out ahead, Spike lets you test it on real alerts first. You can connect a monitoring tool, add an on-call schedule and hear a real test call within minutes.

FAQs

What is the build vs buy decision in incident management?

It's the choice between creating your own tool for alerts, on-call and incident response, or paying for a ready-made one. The decision weighs cost, setup time, upkeep and how much control you need.

Is it cheaper to build or buy incident management software?

Buying is usually cheaper once engineering time is counted. A 20-person team pays $3,360 a year for Spike's Business plan, while upkeep alone on an in-house tool can take a quarter of an engineer's time.

How long does it take to build an incident management tool?

Usually 6 to 12 months for a tool your team can rely on. A basic Slack bot that alerts one person can be ready in a few weeks, but escalations, on-call schedules, reliable phone and SMS alerts and separate hosting take much longer. The work doesn't stop at launch either: keeping it running takes a quarter to a third of an engineer's time every year. A ready-made tool like Spike can be set up in a few hours.

Are there free, open source incident management tools?

Yes, a few are still in active development. GoAlert was built at Target and handles on-call schedules, escalations and phone or SMS alerts. OneUptime combines monitoring, status pages, on-call, and incidents in one platform. Keep groups and filters alerts before they reach the on-call engineer. All three are free to download but still need hosting, upkeep and your own phone and SMS provider.

What happens if an incident management vendor shuts down?

You move to another tool before the deadline. When Atlassian ended OpsGenie, customers got nearly two years' notice and a way to export schedules, escalation policies and alert history. Spike's OpsGenie migration guide walks through the move.

Does team size change the build vs. buy decision?

Yes, but mostly at the far ends. Small and mid-sized teams rarely have spare engineers to build and maintain an incident tool, so buying almost always fits. Very large companies with thousands of engineers and a dedicated platform team are the exception. At that size, per-user fees add up and a custom tool can pay for itself.

Can a bought incident management tool meet compliance needs like SOC 2 or HIPAA?

Yes, in most cases. Established vendors usually publish their security and compliance details: SOC 2 reports, ISO 27001 certificates or HIPAA support. It can be useful to ask for these during a trial and to check where the vendor stores your data. Building only makes sense when a rule goes beyond what any vendor offers, such as a network with no internet access. Spike's security page lists its own measures, including encryption at rest and GDPR compliance.

Need help setting this up?

Write to support@spike.sh, or book a call with our team at spike.sh/demo, and we will help you set up Spike.

Read this online: spike.sh/build-vs-buy