2015

VictorOps

STATE OF ON-CALL

The second annual survey of those who keep the machines up & running

TL;DR - Summary of Data

- Alert fatigue is still getting worse. It was listed as the #1 pain point around being on-call, with almost 70% responding that it is an issue in their organization, up from 60% last year.
- 52% report that on-call is only sort of getting better or getting worse, down from 60% last year. The challenges are easing... slightly.
- Adoption of automation tools continues to grow.
- Similar to last year, email is still the #1 way people find out about problems.
- The use of a status page as a means to communicate during a firefight is growing.
- ChatOps is gaining in popularity, with reported use up to 40%, from 28% last year.

Motivation

Two years ago when we started collecting data for the first State of On-call Report, we were motivated by the trends started by Agile software development and the resulting pressure continuous deployment was putting on technical operations teams in companies of all sizes. Innovate your software or become irrelevant in a world of Software as a Service (SaaS).

Little did we know at the time, it was a narrative of operations work becoming more difficult and the beginning of the extinction of operations as a whole. What used to be people bolting boxes into racks is now engineers writing code. What used to be uptime via no change has become uptime by constant change.

The clear signal over the last year has been the death of operations as the lone protectors of highly technical infrastructures and systems. More and more organizations are placing developers on the front line of rapid response and resolution of technical problems. Alerting ‘someone’ to a problem is no longer acceptable and we are seeing this in many companies as they dismantle their Network Operations Centers (NOCs) in favor of routing specific system problems to the engineers that built the part of the system having issues.

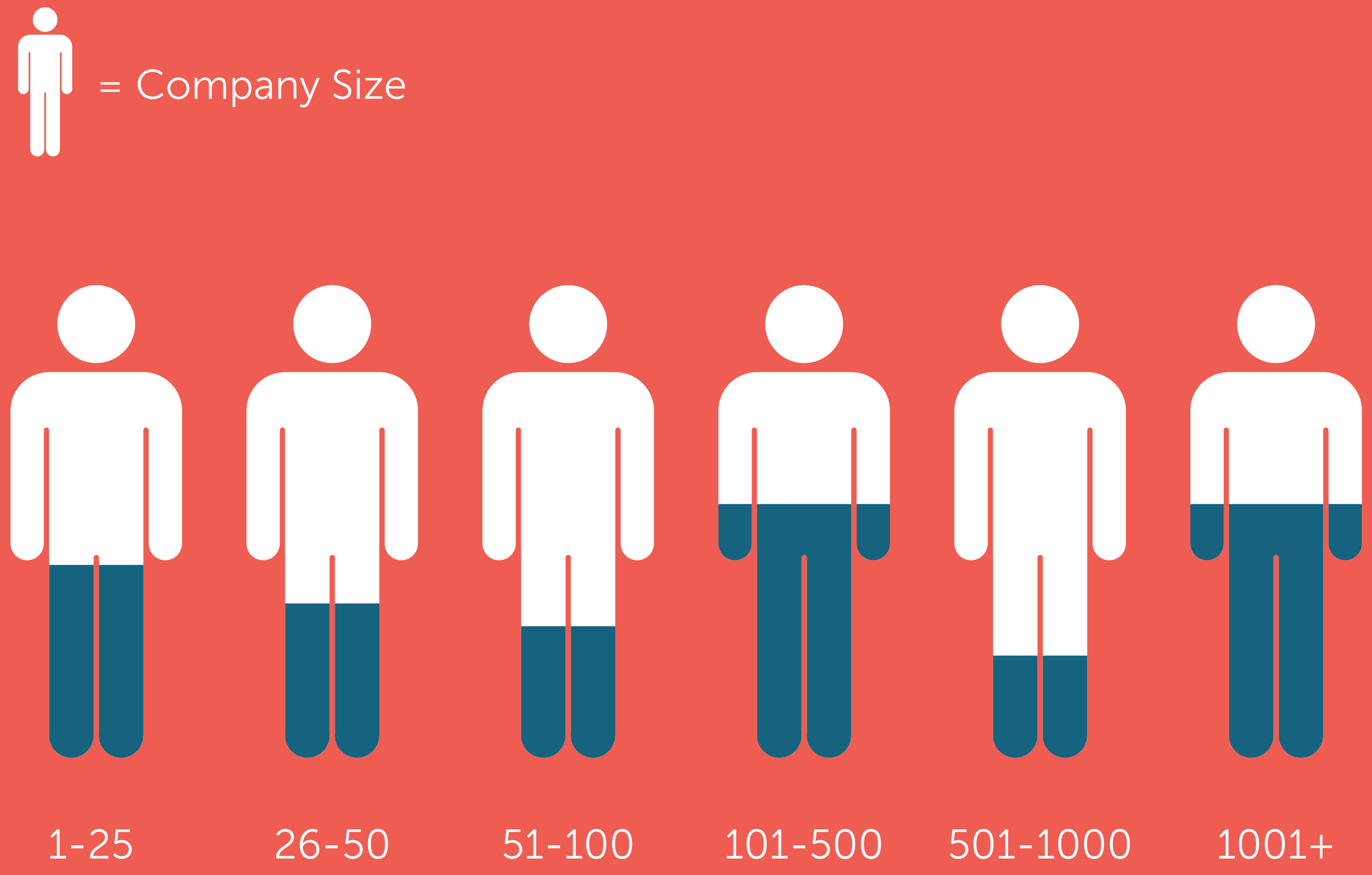
As you’ll see from the following survey results, technology is changing fast and progressive companies are changing faster.

We received more than 600 responses, up from the 500 we received last year, making this what we believe to be the largest sampling of on-call practitioners in the tech space.

Who took this survey?

COMPANIES

The people answering this survey are the ones doing the hard job of being on-call for IT issues. Roles included operations, system administrators, developers, IT, and programmers.



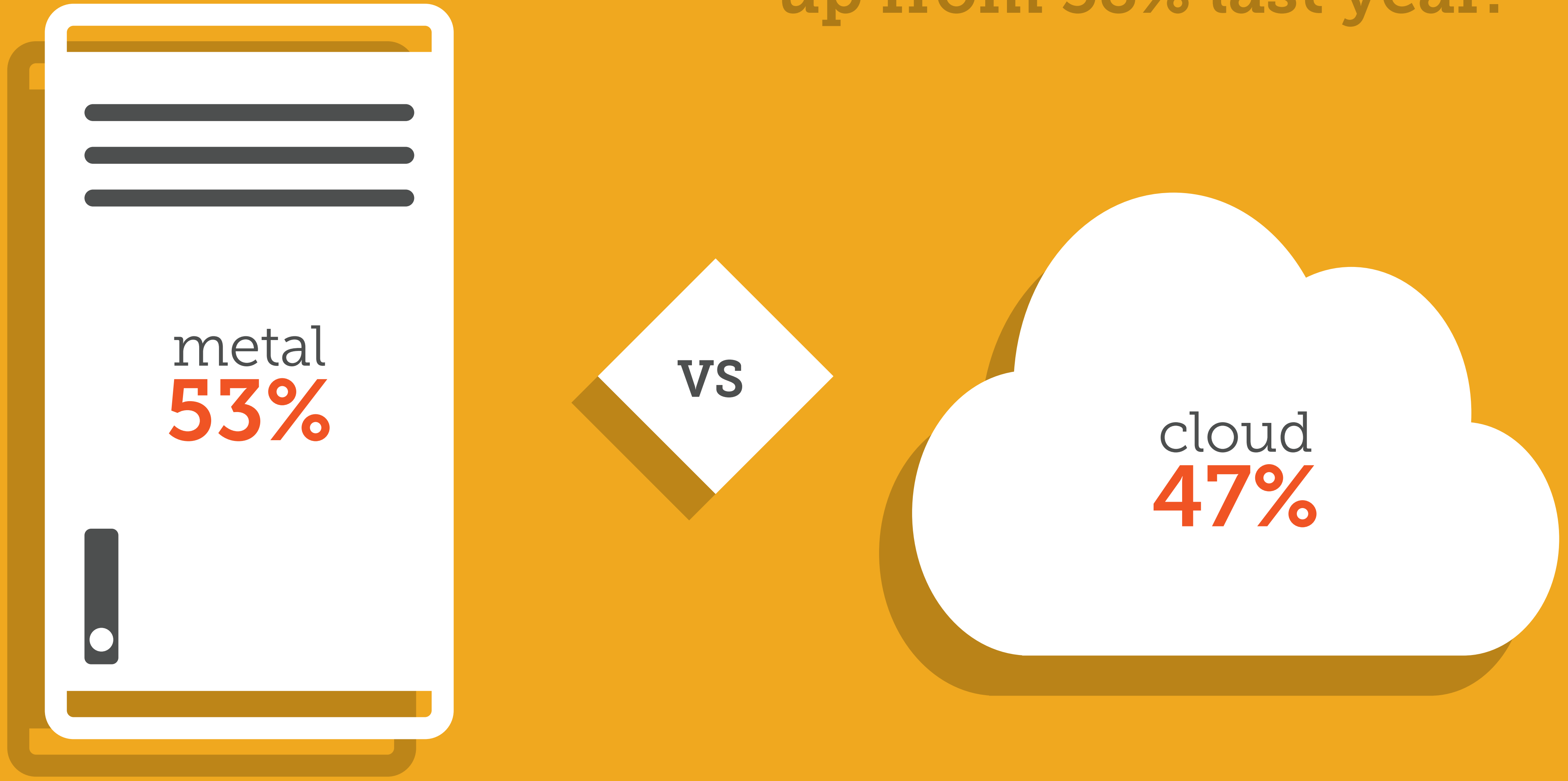
Consistent with last year, the majority of industries represented in this survey are internet-forward companies including software, internet service, and media & entertainment.

Infrastructure & Automation

Cloud has (almost) caught up to metal when it comes to infrastructures, with an **11%** increase over last year.

Automation tools, like *Puppet & Chef*, have taken off even more since last year, with 70% reporting regular use, up from 58% last year.

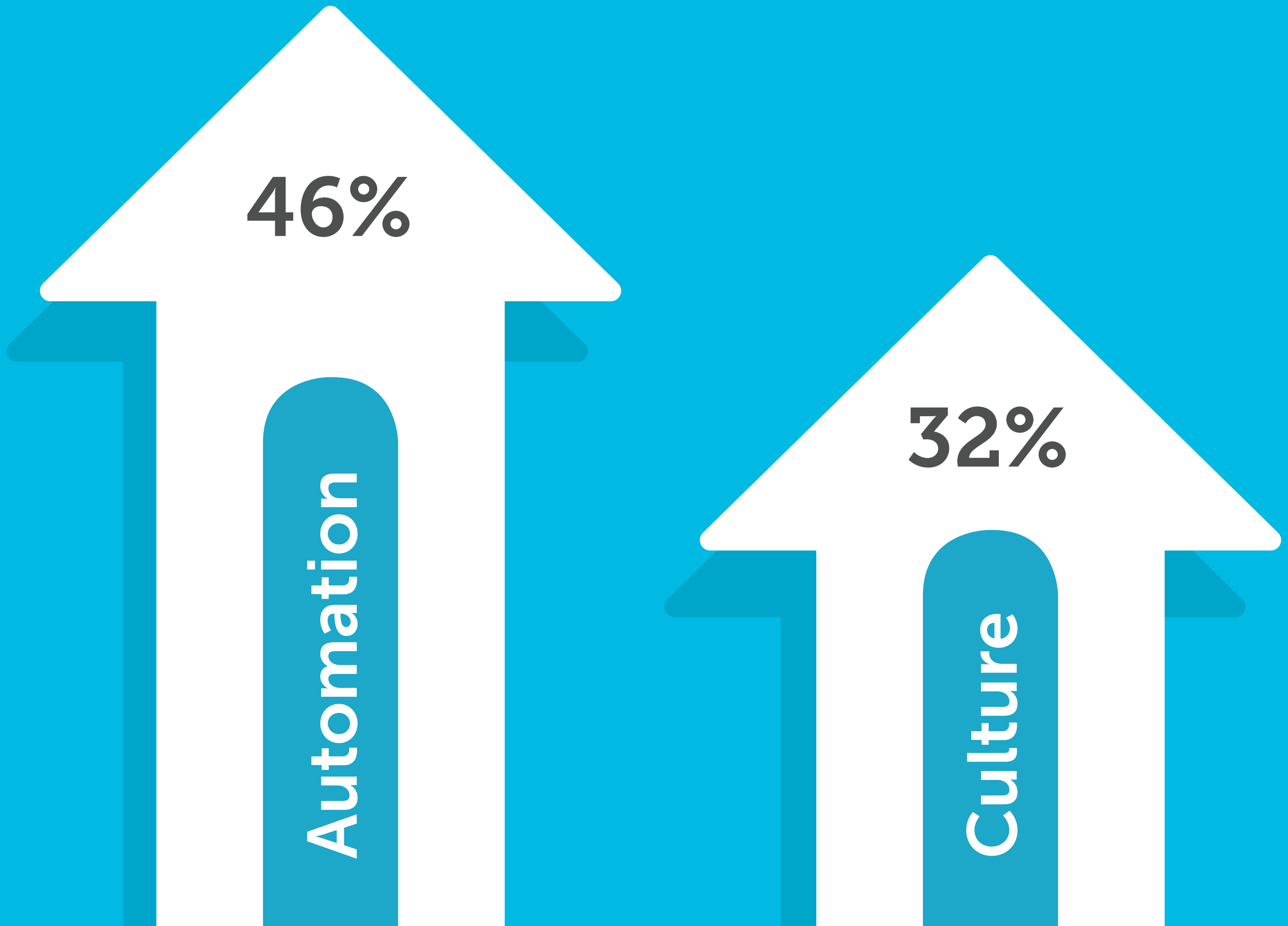
The infrastructure breakdown:



DevOps is maturing

The majority of respondents consider themselves at least minimally knowledgeable about DevOps, with 76% reporting a year or more of DevOps experience, up from 52% last year.

When asked about what DevOps principles they're currently employing, respondents indicated that the top two are **automation (46%) and culture (32%).**



VO INTEL: With our alert annotations and easy collaboration, VictorOps can help with your DevOps goals.

Being on-call still sucks...but is improving

80% SAY THE PROBLEMS AROUND BEING ON-CALL ARE, OR ARE SORT OF, GETTING BETTER.

Top 4 pain points of being on-call:

- Alert fatigue (constantly getting paged for non-actionable alerts)
- Not sure what's happening across all the systems when I get alerted
- Leaders in the organization don't have context around what's happening during the firefight
- Not having the relevant information to solve the problem myself

The best part of being on-call...?

- "The ability to experience new challenges."
- "Creating customer happiness ;)"
- "Solving issues, getting things operational and online again.
- **"Good scotch."**
- "We leverage the insight gained from our on-call rotations as the primary catalyst and feedback loop to drive improvements in our DevOps capabilities."
- "The on-call system has created a shared-accountability system within the engineering org where one of the quickest ways for a fresh engineer to gain understanding of our services and how they fit together, as well as to gain the friendship and comradery of their coworkers, is to go on-call as a shadow for their service."

Managing On-call

THE MAJORITY OF RESPONDENTS (54%) HAVE 2-5 PEOPLE THAT MAKE UP THE ON-CALL TEAM AND ARE ON-CALL FOR A WEEK AT A TIME.

In support of something we saw last year (and a growing trend, apparently), there is an increasing practice of sharing the on-call responsibilities between different disciplines.

- Operations
- IT
- Developers
- DevOps
- Support

VO INTEL: We put together a guide to getting your devs added to the on-call rotation.

SaaS on-call management tools are taking off

Only **25%** of respondents reported having built an internal system for managing their on-call process, a dramatic decrease from the **70%** saying the same thing last year.

A few downfalls of using a homegrown solution include...

- Not being able to scale it easily
- Not knowing how much you're spending in time & money
- The inability to pull reports

In their words...

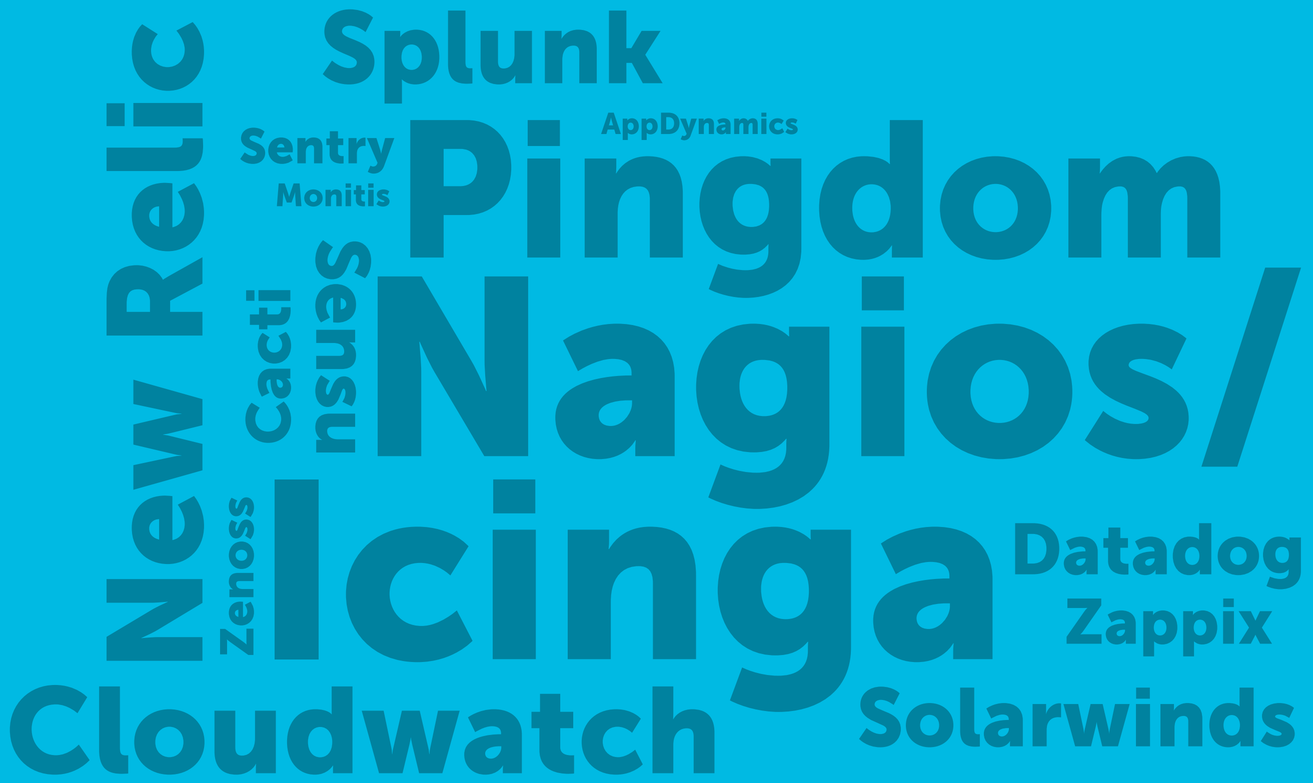
"I would recommend not using a homegrown solution."

"Homegrown solutions are not worth the money invested in them."

"We have to support the thing we created."

Monitoring & Alerting

There are more and more monitoring solutions delivered every year, with most companies using a variety of these tools. When asked what monitoring systems they're using, here's how our respondents answered:



Email is still the #1 way that people find out about issues with their infrastructure or applications. Other ways include...

- SMS
- Push Notification
- Telephone Call
- Dashboard

Alert Fatigue is getting worse

65% percent reported that alert fatigue is a problem in their organization. In addition, almost **31%** of respondents believe that over a quarter of alerts are false alarms.

The following were the most popular steps taken to reduce alert noise:

1. Adjust alert thresholds
2. Regularly evaluate and delete superfluous alerts
3. Route incidents to specific people or teams

VO INTEL: Our transmogrifier can help to reduce fatigue and quiet unactionable alerts by ensuring that alerts get sent to only those people who can resolve them.

In the Heat of the Moment

Survey respondents ranked the importance of the following tools used during the firefight (and were asked to choose all that apply). Below is the list from most important to least:

- Synchronous chat & collaboration tools
- Wiki articles
- Graph tools
- Runbooks
- 1:1 phone calls
- Conference calls

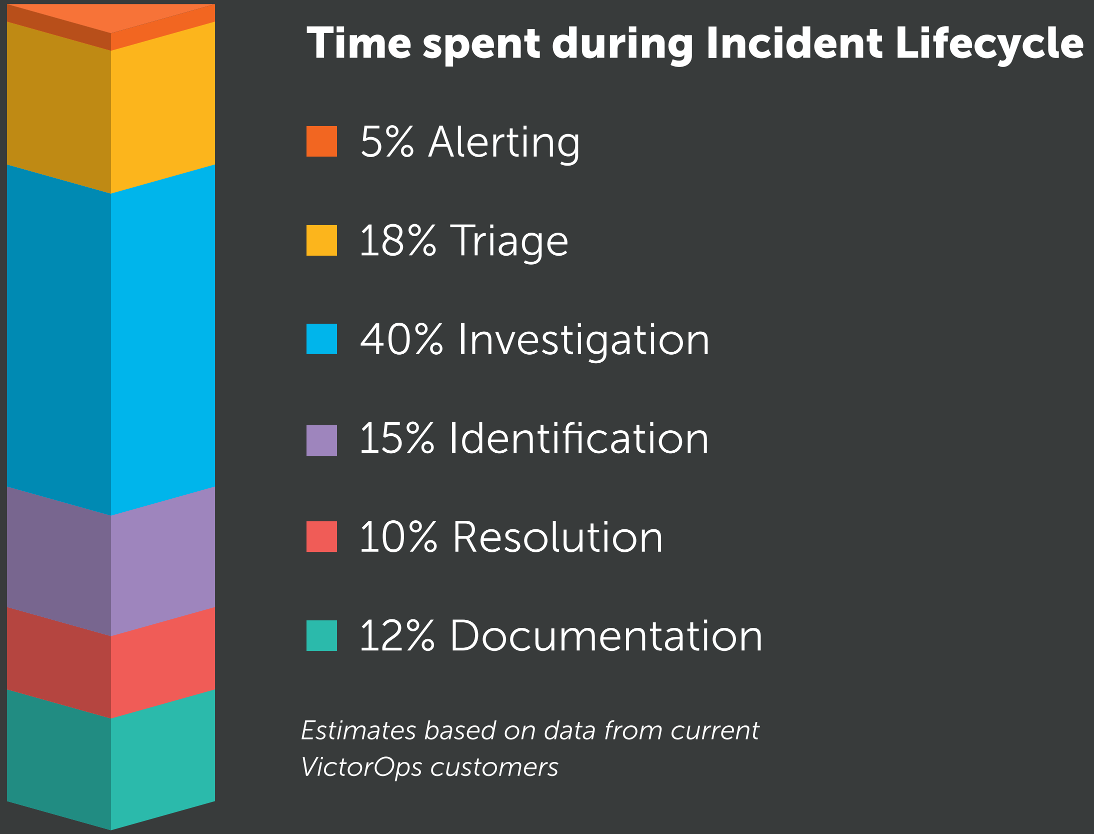
VO INTEL: Our StatusPage integration makes it easy to update stakeholders from the middle of a firefight, without leaving the tool you're working in.

When asked how they communicate externally during an incident, email still reigns supreme, with a status page, website and Twitter rounding out the list.

Incident Lifecycle & Being On-call

The incident lifecycle goes from the moment you're first alerted to a problem through to when an incident is resolved.

32% reported that triage (gaining situational awareness & getting others involved) presents the most challenges while **19%** believe that the investigation phase of the incident lifecycle is most difficult.



Time spent during Incident Lifecycle

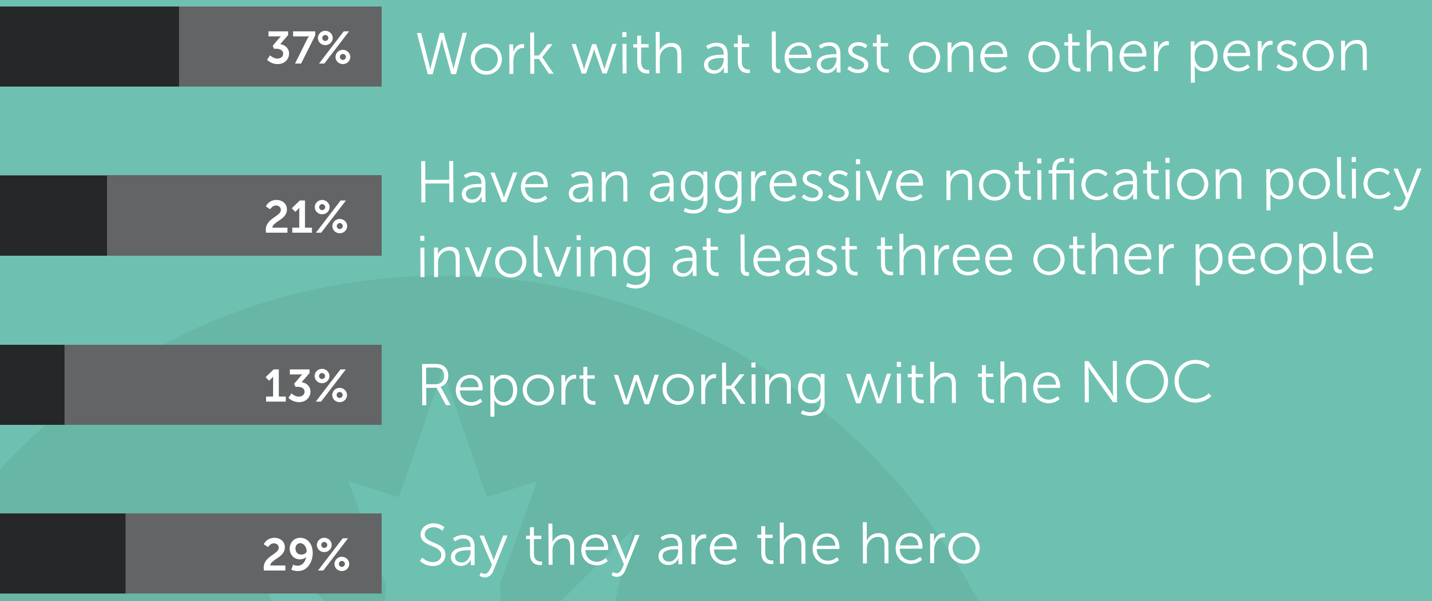
- 🟧 5% Alerting
- 🟨 18% Triage
- 🟦 40% Investigation
- 🟪 15% Identification
- 🟥 10% Resolution
- 🟩 12% Documentation

Estimates based on data from current VictorOps customers

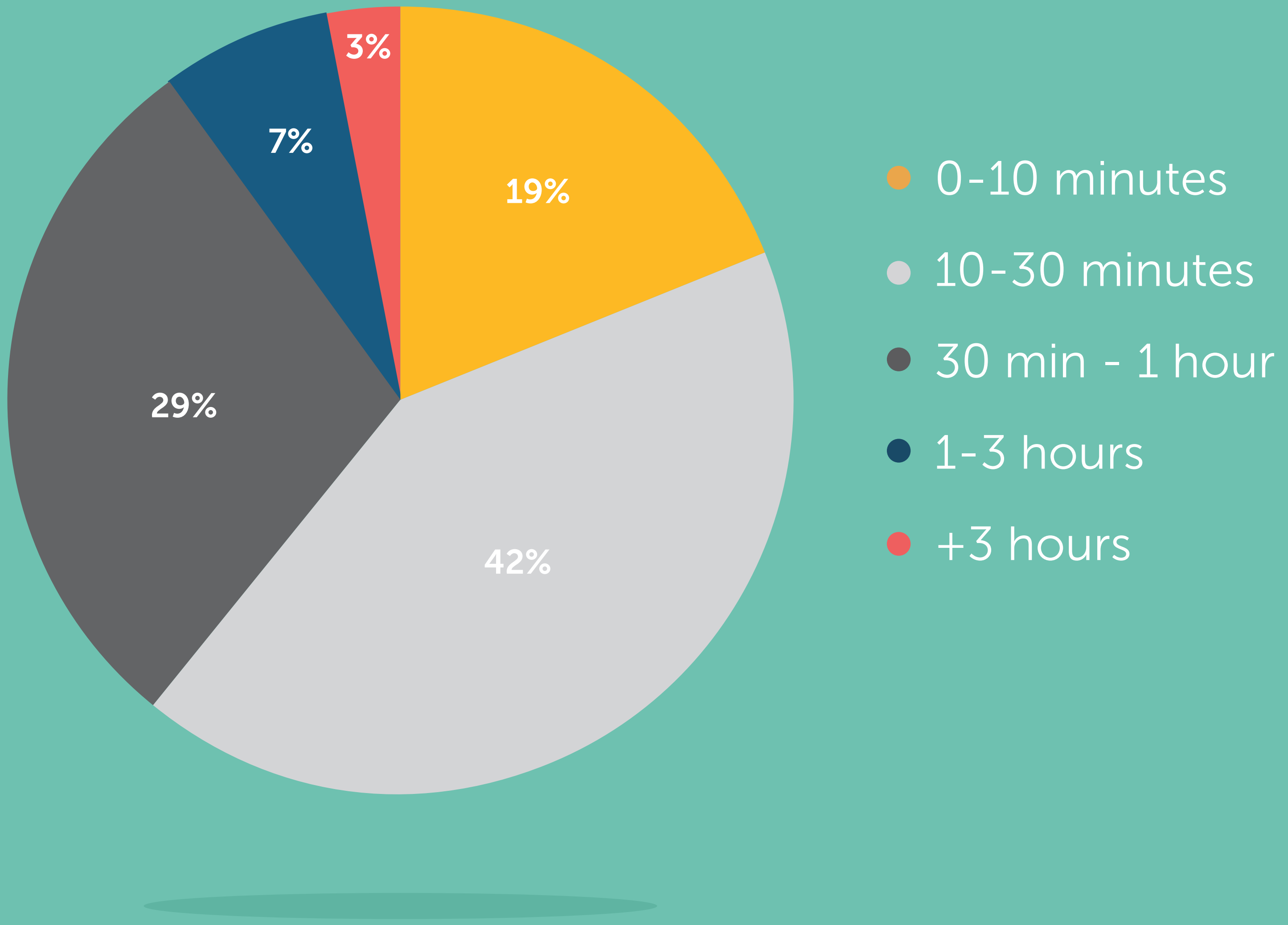
VO INTEL: Want to learn more about the Incident Lifecycle? We've got you covered.

Resolution

In a given incident, **71%** reported that they work with at least one other person from their team to solve the problem.



42% of respondents said it takes 10-30 minutes, on average, to resolve an incident.



More Resolution

The **top 3 ways** that information is accessed to solve a problem:

- Internal wikis
- Runbooks
- Graphite graphs

VO INTEL: Our transmogrifier allows users to append important information to incoming alerts, while our control call feature saves time in launching a conference call right from the firefight.

What's been the most valuable in helping you to solve problems faster?

"Having chat / conference line is key to coordinating troubleshooting efforts and direction."

"DevOps culture. You write the code you own the alerts."

"Runbooks, as they allow me to solve problems outside of my knowledge domain."

"Having the right person with the right knowledge involved or having the resolution already documented."

Use of ChatOps is increasing

ChatOps continues to prove that it's not a passing fad or trend. **40%** report practicing ChatOps during remediation, up from **28%** last year

Tools that respondents are using to facilitate ChatOps include:

- Flowdock
- HipChat
- Hubot
- Litabot
- Slack
- VictorOps
- IRC

VO INTEL: If you're looking to learn more about ChatOps, be sure to check out the ChatOps for Dummies ebook, authored by Jason Hand, our DevOps evangelist.

Post-mortems

Responses were split evenly down the middle on this question, with half saying they did have a defined post-mortem process and half saying they did not. **66%** responded that they only perform them after significant outages.

Fortunately, **81%** report that they strive for blameless post-mortems, up from **65%** last year.

The majority of respondents use post-mortems to share with their team & executive shareholders.

VO INTEL: Our reporting tools make it easy to conduct thorough post-mortems with little effort, while our Getting Started guides provide tips and resources for how to practice blameless post-mortems.

Work Smarter

Post-mortems jumped to the top of the list this year when respondents were asked what they look at in order to improve their process. Additional things that people are viewing as a means to get better include documentation, reporting, alert accuracy and MTTR.

VO INTEL: If you're interested in getting more team members to share with the on-call responsibilities, check out our guide to making it happen.

Final thoughts about being on-call...

"Getting more team members to share in the responsibility helps."

"It's about the most painful and stressful thing a job can ask you to do."

"Knowledge sharing is a must. You can't expect everyone on the team to remember how one person fixed any specific problem. It has to be documented and the documentation has to be accessible and easy to find."

"On-call is only as good as whatever is generating the alerts."

"If people are responsible for ongoing development of the systems they're on-call for, that alignment of incentives makes things improve over time."

Thanks to those that helped us spread the word:

DevOps.com

O'REILLY®

StatusPage.io

LINUX JOURNAL

FOUNDRY GROUP

DEVOPS WEEKLY

New Relic®

If you need help improving your on-call process, we've got a trial to show you just how we're making the process suck less.

SEE HOW HERE