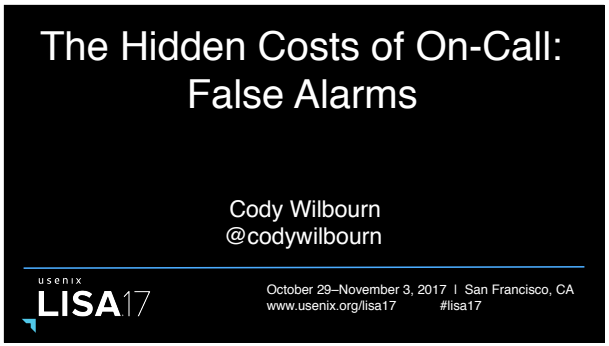


Presentation by Cody Wilbourn, Software Engineer @ parse.ly

Why do vendors charge more for 4-hour support than next business day? The reason is that there are costs associated with it. Availability, scheduling, vacation, benefits. Time spent responding to one customer can't be spent on another, and they don't know how long the issues will take to resolve.

Why aren't these costs taken into account when you put a salary-exempt employee on an on-call rotation? Employees are just expected to handle whatever alarms are thrown at them, including the false alarms. Businesses aren't hit by any additional direct costs because it's "part of the job responsibilities" in addition to the rest of their work.

As a sysadmin it's easy and tempting to ignore and set up more email filters to ignore the false alarms.

Just a warning, some of these numbers are going to be simplified to emphasize the point. We're also going to talk about costs in terms of costs to the business.

Investigation time is directly measurable by your wage - hourly wage * time spent.

Time spent isn't the only cost of investigation. You have to add the opportunity cost against other work. In order to investigate, you had to drop what you were doing, since on-call is in addition to your normal job responsibilities.

Opportunity Cost = (Return of most lucrative option) - (Return of chosen option)
Your most lucrative option is your normal job responsibilities
If the alert turns out to be a false alarm, you saved $0 against outages, and if you didn't fix the monitor causing it, you produced $0 of value, meaning your opportunity cost is the value of what you should have gotten done.

Switching back to what you what you were supposed to do, even a brief interruption has a cost — up to 40% of someone's productive time (American Psychological Association). Add another 40% of intended productivity on top of your investigative time and opportunity cost.

This snowballs quickly as you get bombarded with alerts.

http://www.apa.org/research/action/multitask.aspx

For easy math, let's say an employee's fully loaded cost is $100/hr. So that's salary plus benefits, not necessarily take-home pay.

Now we're going to make a leap into productivity. While every business will calculate expected productivity differently, taking into account ongoing projects and individual roles, a business seeks to make a profit on every employee. Otherwise, it mays no sense to hire the employee. Let's say again for easy math the business wants employees to create an amortized 2x their costs, this pays for the employee + profit.

2x employee costs = $200
Remember, a false alarm provides $0 in value if it was investigated and no other action occurred, like fixing the alert, which would have produced value.
Opportunity cost = $200 (Intended productivity) - $0 (Return on investigating the false alarm)

Context switching is an additional 40% of someone's productive time — that 24 minutes as you review what you were thinking about prior to the interruption and pick up where you left off. As we said, our productivity was valued at $200/hr; 40% of that.

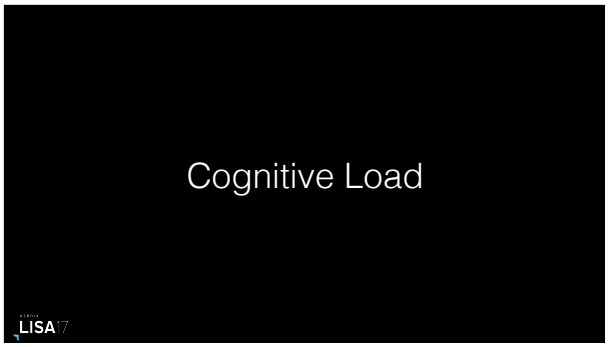
For just a single hour of investigating a false alarm, you're looking at a cost multiplier of almost 4x of the employee's cost. It only goes up from there as you increase the intended revenue per employee.

Employees may not spend an hour looking at a single issue, but might accumulate that over the course of days or weeks. Amortized costs, same as the productivity.

This is the cost over 1 year if that $100/hr person spends 5% of their time checking on false alarm notifications they've received being on-call.

5% time isn't actually that much. It's 1 day out of the month. It's 2 hours a week. Spend a minute on each email and you hit that with a rate of 3 emails per hour during the workday. Nobody notices, especially in an on-call rotation. For any individual employee in the rotation it's 2 hours out of their month, 3 emails an hour, why would they mention it?

$39,000 you can pull out of the couch cushions by reducing false alerts.



Cognitive Load is the idea that you expend effort to remember things. If you have a lot on your plate, it becomes difficult to manage everything, and you turn to something like ToDo lists. It's also one of the benefits of standardizing systems — instead of remembering every fact about every system, you memorize patterns. These patterns allow you to scale to many more systems, cattle vs pets.

Alerts have a cognitive load as well — Which alerts are real? Which alerts are noise?

All of the stuff you have to "know" to be on-call is actually tribal knowledge that has to be taught to someone joining your on-call rotation. How long do you have to spend to teach someone new this knowledge? Reducing the amount of tribal knowledge required will reduce the upfront costs of getting someone on-boarded without losing any of those future returns.

If that tribal knowledge training was skipped or forgotten, how many alerts does the new person have investigate in order to learn that alert is a false alarm? There's direct investigation impacts there — as we said, 4x their hourly costs.

But what if those false alarms never happened? You wouldn't need to spend time teaching the tribal knowledge of which alerts were false, meaning you can eliminate those investigation costs.

Have you ever had “phantom pager alert” where you swore your pager buzzed but it didn’t? It’s a hallucination caused by the stress of being on call.

Everything talked about so far adds to your levels of stress. There are other stressors of on-call too.

Urgency - On-call has a short window where they have to respond to an incident, and being late for anything is stressful for many. There’s an urgency in that being down is bad, so you need to act as quickly possible.

Uncertainty of when an alert may come. Want to run to the store or see a movie? You might have to cut that short. An alert could come as soon as you try to go to sleep, or in the middle of the night, or before your alarm clock rings.

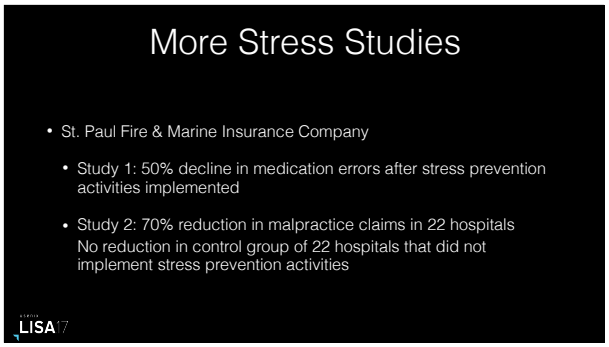
Duration An on-call rotation longer than about a week or so and you get worn down from the repeated alerts. If your on-call rotation only has 2 or 3 people, even an on-call “week” turns into being on call for the better part of half the year.

Expectations Whether this is internal or external, it’s easy to be pressured by the on-call. *It’s your job to keep everything up and running*. If the environment is down, that’s bad, so you *must not be doing your job*.

Stress can contribute to mistakes on the job and a variety of health problems. According to the Bureau of Labor Statistics, workers who must take time off work due to a stress, anxiety, or related disorder will be off the job for about 20 days.

Can the business afford to be down an employee for 20 days, not to mention any medical costs associated with insurance claims?

https://www.cdc.gov/niosh/docs/99-101/ - Sorry couldn’t find more recent, reporting format changed.
Tabular data, 1992-96: Number and percentage distribution of nonfatal occupational injuries and illnesses involving days away from work, by nature of injury or illness and number of days away from work. Date accessed: 1998.



Here we have studies performed in hospitals. A 50% decline in medication errors, and a 70% reduction in malpractice claims by reducing stress.

The way the study reduced stress was by

* Educating employees and management on job stress
* Changes in policies and procedures to reduce organizational sources of stress
* Established employee assistance programs, specifically help and counseling for work and personal problems

Would an organizational transformation like this translate to fewer bugs, fewer outages? Most likely.

Jones JW, Barge BN, Steffy BD, Fay LM, Kuntz LK, Wuebker LJ [1988]. Stress and medical malpractice: organizational risk assessment and intervention. Journal of Applied Psychology 73(4):727-735.
From <https://www.cdc.gov/niosh/docs/99-101/>

In a study published in the Journal of Occupational & Environmental Medicine, they put fatigue-related productivity losses at $1967 per employee annually.

In another study, researchers estimated that lost productivity due to poor sleep costs $3,156 per employee with insomnia, and averaged $2,500 for less severe sleep problems.

If employees are missing sleep due to on-call alerts, they're more likely to have trouble thinking, concentrating, and remembering. For the business, the impacts productivity. Free coffee in the break room won't fix because even with caffeine, the brain isn't functioning fully.

Risk of an employee being in an accident increases, as being drowsy can slow reaction time as much as being drunk. At an extreme level of sleep deprivation employees will undergo micro-sleeps, which are seconds to minutes long periods where a person goes unconscious. However, those people are often unaware of the microsleeps, instead believing they were awake the whole time or they temporarily lost focus.

Accidents impact worker's compensation rates after a claim. And you don't want an OSHA reportable event.

A variety of health problems (also correlated to stress) can occur like weakened immune system and high blood pressure, as well as increasing risk for obesity, heart attack, stroke, and diabetes. One study found participants who had fewer than seven hours of sleep were almost three times more likely to develop a cold than those who slept for seven hours or more. This comes back to health insurance rates besides employees requiring sick leave to visit a doctor.

Journal of Occupational & Environmental Medicine: January 2010 - Volume 52 - Issue 1 - pp 91-98. doi: 10.1097/JOM.0b013e3181c78c30
http://journals.lww.com/joem/Abstract/2010/01000/The_Cost_of_Poor_Sleep__Workplace_Productivity.13.aspx - Used a Work Limitations Questionnaire
https://hbr.org/2011/01/sleep-deprivations-true-workpl.html
https://www.webmd.com/sleep-disorders/features/10-results-sleep-loss#1
http://www.bbc.com/news/health-41666563

2014 Society for Human Resource Management survey on workplace flexibility found that 32% of companies saw a decrease in absenteeism after they implemented flex-time policies. 26% reported productivity increases. If people could get what they needed in their lives done, they became less distracted, more productive, and they didn't need to play hooky from work.

If between their typical workload and after-hours on-call, employees can't live their lives, their lives are going to suffer. They will mentally "check out" and not put forth more effort than the bare minimum. The stress and sleep deprivation will be compounding factors in the overall work-life balance.

At an extreme, beyond becoming unengaged with the company, they may become actively disengaged. Gallup's 2017 *State of the American Workplace* report puts this number at 16% of employees. Actively disengaged employees negatively impact the business — they negatively influence their coworkers, miss workdays, and drive customers away. They are also more likely to steal from or disrupt the business. Gallup estimates actively disengaged employees cost the US economy $483 billion to $603 billion in lost productivity annually.

What might this look like in practice? Beyond negatively impacting morale, a disengaged employee may intentionally ignore pagers. They'll frequently come up with excuses like they forgot they were on-call, didn't have cell service, or their phone ran out of battery.

These employees may ultimately leave the company, as 73% of actively disengaged employees are looking for jobs or opportunities, compared to 37% of engaged employees. A replacement hire costs 15-25% of first year's salary to a recruiter, plus training.

https://www.shrm.org/research/surveyfindings/articles/pages/2014-workplace-flexibility-survey.aspx
https://hbr.org/2016/06/how-a-flex-time-program-at-mit-improved-productivity-resilience-and-trust
Gallup "State of the American Workplace 2017 report": 33% engaged, 16% actively disengaged, and 51% unengaged.

Alarm fatigue, where you're put in a "boy who cries wolf" situation and ignore a problem because the previous alerts have been false. You become desensitized to the constant alarms and ignore new data.

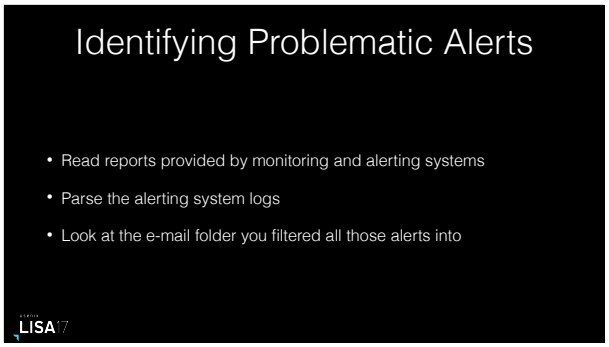
In 2010 a man suffered a fatal heart attack at Massachusetts General Hospital. 10 nurses on duty could not recall hearing the alarms or seeing the scrolling messages on three separate hallway signs that indicated the patient's condition.

State investigators attributed this to alarm fatigue after desensitization to constant alarms throughout the day. The information unintentionally became background noise, which downgraded the severity it represented.

Missing alarms in tech normally doesn't have this kind of life threatening impact, it's generally just a service outage. But what does missing an outage mean for your business? How is it impacted? Generally outages are costly, not only in terms of immediate lost revenue, but also customer perception of your brand.

http://archive.boston.com/lifestyle/health/articles/2011/02/13/patient_alarms_often_unheard_unheeded/

Solutions

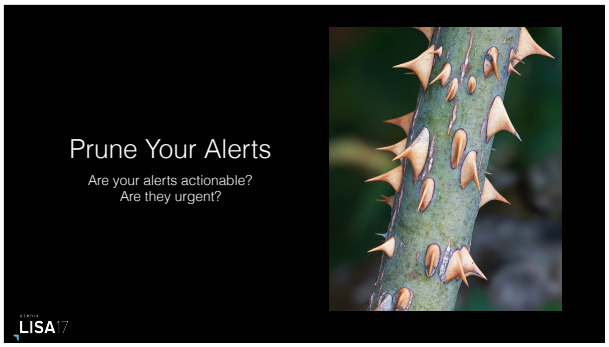


First we have to identify the problematic alerts. How do you find these alerts?

You can see if your monitoring or alerting tool has a canned report you can run. Most do, including Pagerduty if you're using that for contacting you.

You could go parse the logs of your monitoring system, to see what alerts it sends out

You could go spelunking through your inbox and look at that pile of e-mails your filters put aside for you



Next we're going to prune your alerts.

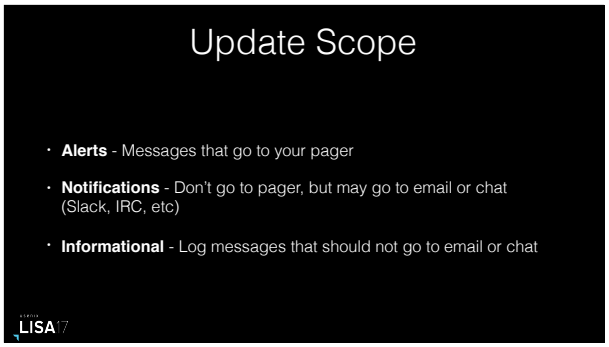
University of Massachusetts Memorial Hospital found that 40% of cardiac monitor patients didn't need to be on monitors at all, based on American College of Cardiology criteria.

Nor do your systems need as much monitoring as you think.

Determine how actionable your alerts are. If this is something you wait for self-resolution, get rid of it.

Determine the urgency. If you don't have to drop everything to solve this, it's not worth of paging you

http://archive.boston.com/lifestyle/health/articles/2011/02/14/no_easy_solutions_for_alarm_fatigue/?page=3
img: https://commons.wikimedia.org/wiki/File%3ARose_Prickles.jpg



For the monitors you do feel are necessary, but not urgent, make sure they have the right scope.

You've got 3 main categories of messages your monitoring system generates

- Pager-worthy alerts
- Notifications, like "warning" level thresholds that would help you to cut the alert off before it happens
- Informational messages that need to be logged somewhere

Monitors that alert when they don't have to are noise, and likewise, critical information being sent as anything other than pagers won't be seen because of your email filters.

Informational messages should not be sent to your inbox. These are all the messages you're currently filtering off to trash. Log this to a syslog, ELK or Splunk server and not your inbox. If you really must use e-mail because you don't have the infrastructure, send these messages to a separate shared email account so it doesn't clog your main account.

With today's large complex systems, we've mostly moved past the single-host catastrophic point of failure scenario.

One dead host or disk isn't as urgent as it used to be. So why wake up for one host, or one disk?

Prefer cluster metrics to host metrics, and user-experience metrics over both of those.
Individual database process checks < Database cluster checks < Checking that a user can access their account information from the database
Consider downgrading the individual node data to informational messages, so it's still available if you needed to query it.

Getting rid of those host-level alerts in a cluster will reduce huge amounts of chatter as the cluster handles most failures for you.

“Alarm flood” is an ANSI & ISA standard defined as 10 or more alarms in any 10 minute window, per operator (on-call engineer when applied to tech). After 10 or so alarms, an operator cannot functionally understand the alerts coming in, because they’re still dealing with another one. Metrics you can track in your environments relating to alarm flood is time in alarm flood and volume of flood.

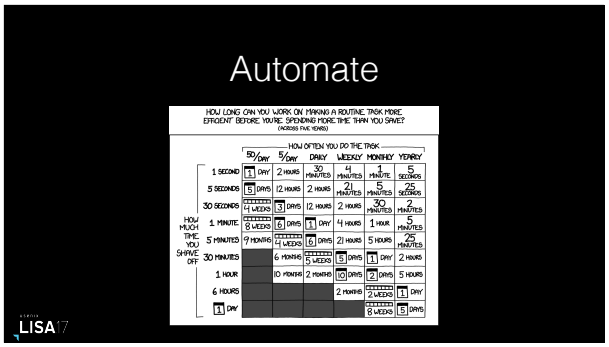
Reminder alerts generate alarm flood because the operator now has to determine that these alerts are repeats and not new information. Reminder alerts should be sent on the order of hours or days, not minutes.

Another way you can reduce alarm flood is by combining related alerts. Composite monitors combine lower level monitors into a higher level one which you can route to a more actionable alert.

Apache Storm runs multiple topologies (jobs) simultaneously. These jobs have shared infrastructure dependencies, like databases. The standard approach is to instrument each job and send an alert if the job isn’t operating properly. I’d get an alert for each job that slows down or stops.

But since these jobs have shared dependencies which can cause shared failures (and frequently do), I can set up a composite monitor that triggers if at least one of the sub-monitors are triggered. So long as the composite alert hasn’t been cleared, on-call needs to continue remediating. The composite monitor provides context of the active sub-monitors for on-call to investigate.

Go back to that statement I made about checking user experience. If the check user can’t access their account, is it the database or web server? Create a composite monitor that would combine this user experience check with the database health check. Create a second monitor combining the user experience check with a web server health check. Now on-call has better context. Downgrade sub-monitors to only send informational messages or low priority notifications instead of pager alerts like they used to.



If your warning or alert has a runbook, why not have the computer fix it for you? Perfect computer task.

Behaviors like restarts, reboots, and `rm` commands are destructive and you'll want to be very sure this automation will always do what you want it to. Leave it in debug mode and just print what steps would have happened before you turn it live.

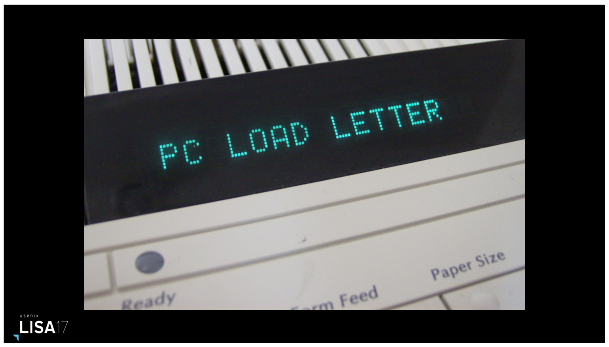
Consider: your entire environment attempts this automated repair at once

img: https://xkcd.com/1205/

If your system is alerting because it's fragile, this needs to be fixed.

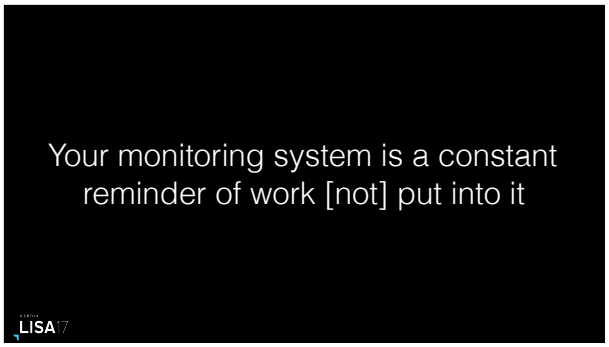
Add high-availability
Add capacity
Get development time to fix brittle code

These things cost money, but this presentation is about the costs. Use this to help justify projects.



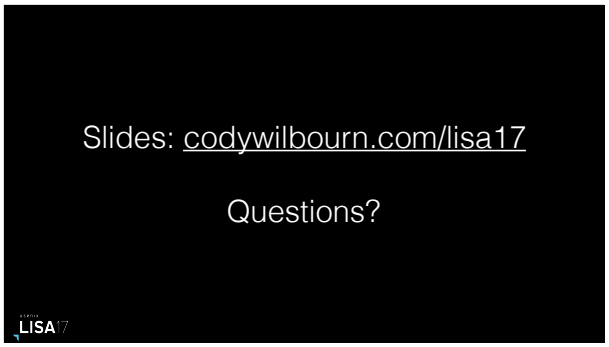
The Printer Rule: As horrible as printers are, you don't need to service them 24 hours a day. Likewise, not everything in your environment needs to be monitored and managed 24/7. No one will care if the issue doesn't get fixed until the next morning.

I've worked with a QA lab to reduce their support to business hours only, except during their month of "crunch mode" where they staffed QA 24/7. QA got the support they needed when they needed it, but for the rest of the year on-call got a huge reduction in alerts outside of business hours.



Your monitoring system is a constant reminder of the work [not] put into it. Default alerts, un-tuned alerts all contribute to this noise that's going to eat into your time, and at worst you'll ignore an actual problem.

Your alert problem won’t end tomorrow. It will be a process, but just take a little bit at a time, get your team involved, and you’d be surprised at what accomplish.



You can find the slides, citations and related readings on my website with the URL here.

Thank you!