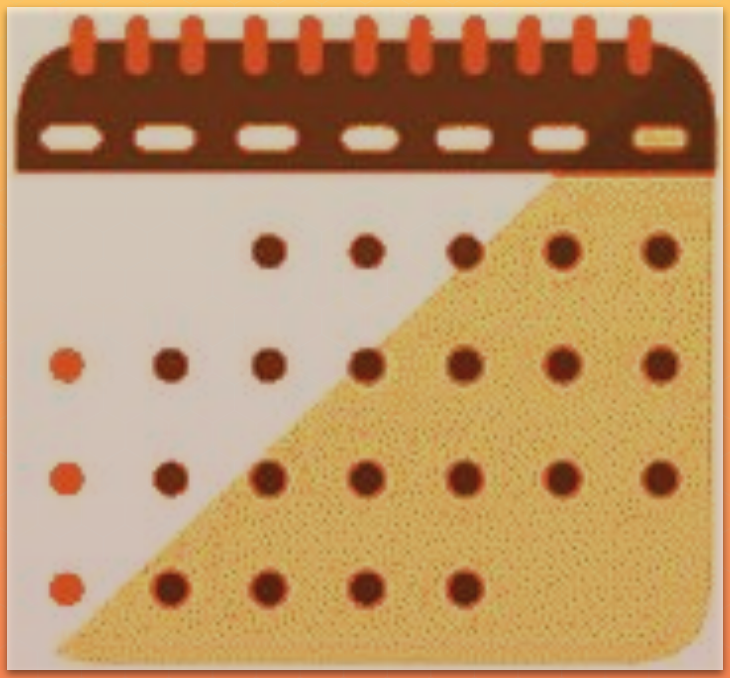
STOP ALERT FATIGUE MONITOR WHAT MATTERS

January 2025

Quality and Reliability Lead at RocketReach

Tester, Breaker, Fixer of all things

REWIND



2022 IN REVIEW

Sev I Incidents

Weekly

Alerts Triggered

100-1000 per day

Estimated Downtime

8-20 hours a month

FAST FORWARD

2024 IN REVIEW

Sev I Incidents

Every Other Month

Alerts Triggered

10-100 a day

Estimated Downtime

1-4 hours a month

COMPARING 2022 TO 2024

Weekly to Monthly

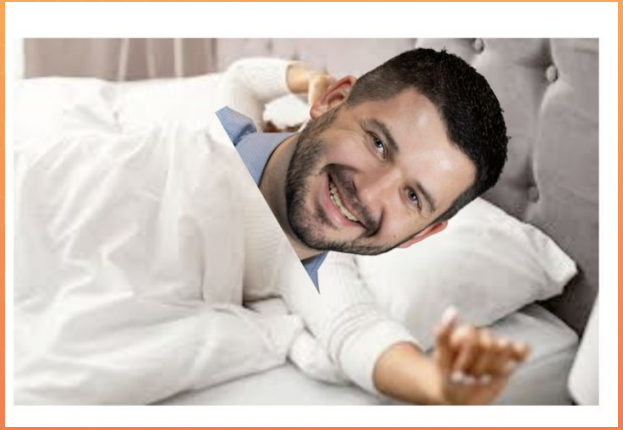
Sev I Incidents

1000s to Double Digits

Total Alerted Incidents

99.4% uptime

Total Downtime



HOW DID WE GET THERE

HOW DID WE GET THERE

Things happened,
lets fix them

REACTIONARY

HOW DID WE GET THERE

8

Reply hazy, try again later

PRECAUTIONARY

HOW DID WE GET THERE

"The best time to start was last year"

TIME AND FOCUS

HOW DID WE GET THERE

Things happened, lets fix them

REACTIONARY

8

Reply hazy, try again later

PRECAUTIONARY

"The best time to start was last year"

TIME AND FOCUS

DOCUMENT YOUR ALERTS

DOCUMENT YOUR ALERTS

Take a holistic inventory of what you have today

Not just fresh memory but dig deep

DOCUMENT YOUR ALERTS

Look through your systems

- messaging
- monitoring system
- log management
- on-call paging
- codebase
- etc…

DOCUMENT YOUR ALERTS

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	**Source**	**Alert Name**	**Description**	**Status**	**JIRA LInk**	**Last Triggered**	**Keep?**	**Recommended Changes**	**Channel**	**Slack LInk**	**Source Link**	**Frequency**	**Whose Alerted**	**How Alerted**	**When Alerted**
2	Datadog	High CPU Usage	ECS Task CPU Usage higher than X% over past 15 minutes	Reviewing	Link	1/13/2025 10:00 AM	Yes		alerts	Link	Link	Hourly	All cloud ops	Squadcast	When happens
3	Sentry	X Errors in Past 10 Minutes	Total errors in sentry for project over x	Done	Link	1/13/2025 10:00 AM	Yes		dev	Link	Link	Every weekday	All dev	Slack	When happens

Sample Link

DOCUMENT YOUR ALERTS

Build out runbooks for your alerts

MEASURE WHAT MATTERS

Just because you have the measurement doesn't mean it’s useful

MEASURE WHAT MATTERS

MEASURE WHAT MATTERS

Are your vitals easily obtainable

Know your system architecture

Some measurements you don’t know you need until after the fact

MEASURE WHAT MATTERS

MEASURE WHAT MATTERS

Inventorying our alerts allowed us to see what overlapped and could be reduced

EVALUATE THE URGENCY

EVALUATE THE URGENCY

Make sure your measurements are actually right to indicate someone needs to do something

EVALUATE THE URGENCY

Can you configure your paging system or your monitors to work when you need them to?

EVALUATE THE URGENCY

Can you set work and non-work hours?

Do you have off hours where an alert is not necessary?

EVALUATE THE URGENCY

Is the action that failed necessary to alert about or can it be a lower priority notification?

DON'T LET THE TOOLS DICTATE YOUR STRATEGY

Before throwing money at a tool, figure out what your ideal world looks like

DON'T LET THE TOOLS DICTATE YOUR STRATEGY

Evaluate your toolset
Does a higher edition contain what you need?
Does a competitor do what you need?

DON'T LET THE TOOLS DICTATE YOUR STRATEGY

System doesn’t have an integration with your paging solution… Build One

DON’T LET THE TOOLS DICTATE YOUR STRATEGY

Create custom system alerts in your application code

DON'T LET THE TOOLS DICTATE YOUR STRATEGY

THINK SMALLER

THINK SMALLER

Prove out your hypothesis first

Don’t just flip a switch and enable the world

THINK SMALLER

Focus on a the highest impact first, iterate if necessary and move on

BE FRUGAL BUT DON'T BE CHEAP

Don’t have the metric you need, figure a way to add it even if it means a couple bucks a month

BE FRUGAL BUT DON'T BE CHEAP

Reallocate money to another tool that has more value than another

BE FRUGAL BUT DON'T BE CHEAP

Don’t need top tier enterprise solution, but maybe that next level solves your issues

BE FRUGAL BUT DON'T BE CHEAP

PRACTICE MAKES BETTER

PRACTICE MAKES BETTER

Things run smoothly and then all of a sudden, 3 AM phone calls happen

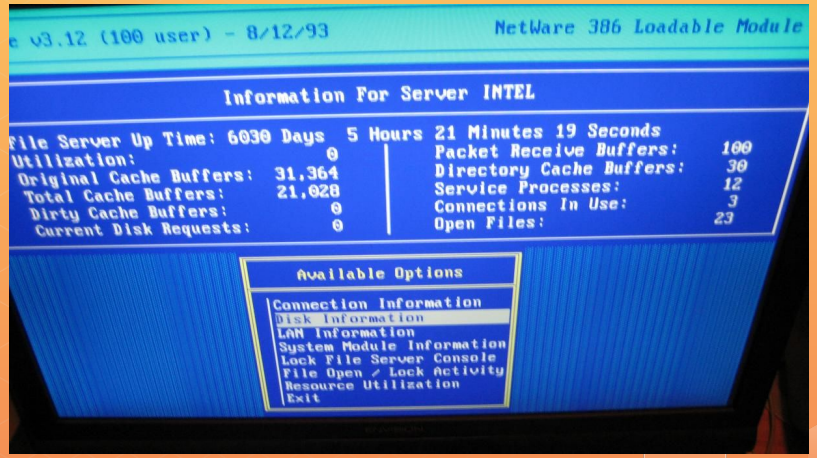
PRACTICE MAKES BETTER

Schedule time to complete mock incidents

PRACTICE MAKES BETTER

Chaos experiment, not just a devops check but also a good way to test your processes

PRACTICE MAKES BETTER



PRACTICE MAKES BETTER

Build automation to not make it on-calls responsibility

ALERT THE RIGHT FOLKS

Associate alerts to the right owners, not just anyone who is on-call

ALERT THE RIGHT FOLKS

ALERT THE RIGHT FOLKS

Have the proper escalations in place

Have everyone who can fix something in your alerting system

One person who knows how to fix it all? Stop that and start sharing the wealth. Bus factor is a real thing.

ALERT THE RIGHT FOLKS

WHO KNOWS YOUR SYSTEM BEST?

WHO KNOWS YOUR SYSTEM BEST?

Work with Customer Support, Sales, Customer Success, etc… to understand vital flows

WHO KNOWS YOUR SYSTEM BEST?

A process you think is important and crucial to the system might actually not matter to the end user

WHO KNOWS YOUR SYSTEM BEST?

Understand your user usage patterns, both how they use the app and when they use it

BUILD INTO SDLC

BUILD INTO SDLC

Make monitoring and alerting part of your process, not an afterthought

If you're not measuring your system, how do you know it's impact?

BUILD INTO SDLC

DON'T JUST FIX IT, PREVENT IT

DON'T JUST FIX IT, PREVENT IT

Incidents happen, after they happen make sure they don’t bite you again

DON'T JUST FIX IT, PREVENT IT

Build tests for what broke

Take your post mortems seriously and have traceable takeaways

DON'T JUST FIX IT, PREVENT IT

Act on your takeaways ASAP from a post mortem, don’t let them get buried in a backlog

TOOL SHOUTOUTS

Checkly

Synthetic monitoring that allows you to do API heartbeat style checks as well as Playwright E2E tests

Squadcast

All in one incident management tool

Datadog

Observability service for monitoring servers, databases, and other aspects of your products

Sentry

Application performance monitoring and error tracking for all of your systems

THANK YOU

Ways to reach me:

clymerrm