Three Reasons Developers Struggle with AppSec… and How to Make it Easier

Hint: None of the reasons are “Developers Don’t Care about Security”

SNOWFROC

$ whoami

Maecenas faucibus mollis interdum.

Scott Gerlach

- CSO/Co-Founder StackHawk, Inc
- CISO @SendGrid - 3 years
- Sr. Security Arch @GoDaddy - 9 years
- Husband, Dad, Brewer, Golfer, tinkerer
- @sgerlach
- linkedin.com/in/scott-gerlach-kaakaww

WARNING

AppSec Problem Overview

AppSec = Good; In Theory

Static Code Analysis

- Noisy, often lacks Application Context
- Language Dependant (Don’t get me started on IDE support)

Dynamic Code Analysis

- Better at actual app and context, but still somewhat noisy
- Hard to use

RASP, IAST, WAF

- Wait til someone else finds it… in Prod

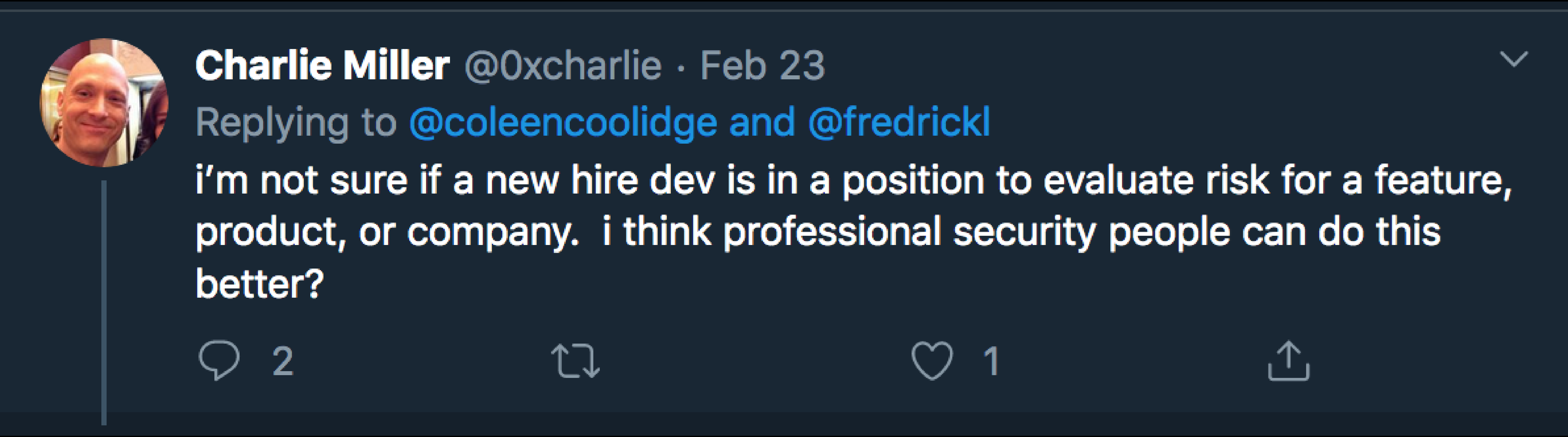
AppSec Is Hard to Scale

All solutions need people, specifically **security** people, to be effective *(ok, maybe SAST doesn’t necessarily)*

No try and buy; only ‘Contact us for a Demo’ and spend tens of thousands.

$10 000
10 000
$10 000
$10 000
$10 000
$10 000
$10 000
10 000
$10 000
$10 000
$10 000
$10 000

Trust Issues



Charlie Miller @0xcharlie · Feb 23

Replying to @coleencoolidge and @fredrickl

i’m not sure if a new hire dev is in a position to evaluate risk for a feature, product, or company. i think professional security people can do this better?

Trust and Support

> “ I wouldn’t want to put a new hire Developer in the position of making an uninformed risk decision ”

-Scott Gerlach

We care; differently

Devs Tend to Care About

(In this order)

- **Performance** (time, LoadTest)
- **Quality** (Unit/Integration Tests)
- **Easy of Use** (Linters, UX)
- **Efficiency** (Heap Inspection)
- **Security** (uhhh, google)

InfoSec Tends to Care About

(In this order)

- **Security**
- **Security**
- **Security**
- Feelings

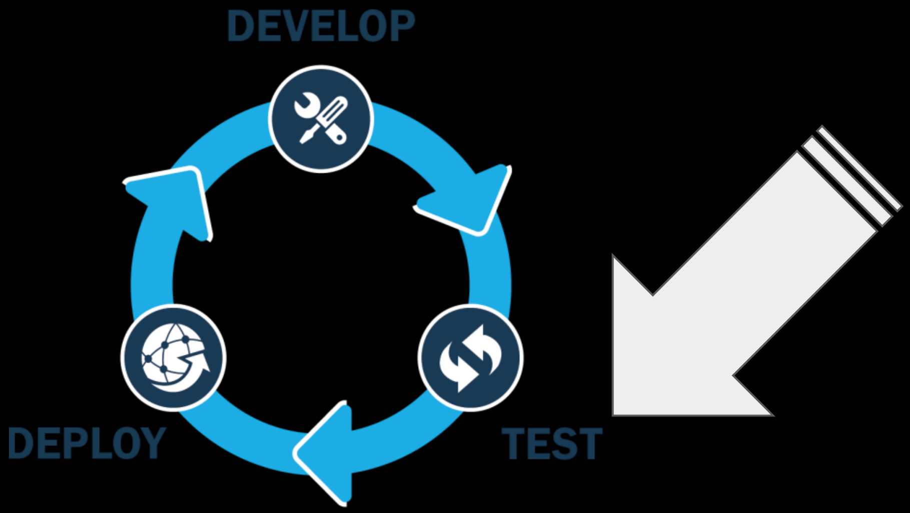
Notice how Devs care about stuff they can easily know and make decisions on

Evolution of Linting

Code Standards Documentation

Corporate hath deemed

- Thou shalt use spaces not tabs
- Thyne indentation shalt be 3.5 spaces
- Nary shalt thou employ semicolons in javascript
- Thy variable names shalt be snake_Camel_Case
- Thyn IDE is standard and pico is it’s name



Linters and Linting Rules

Someone else did the heavy lifting for us!

But we’ll add this little rule ourselves.

Hey the CI Pipeline can help enforce this!

Problem One: Knowing AppSec

Or at least being able to make decisions

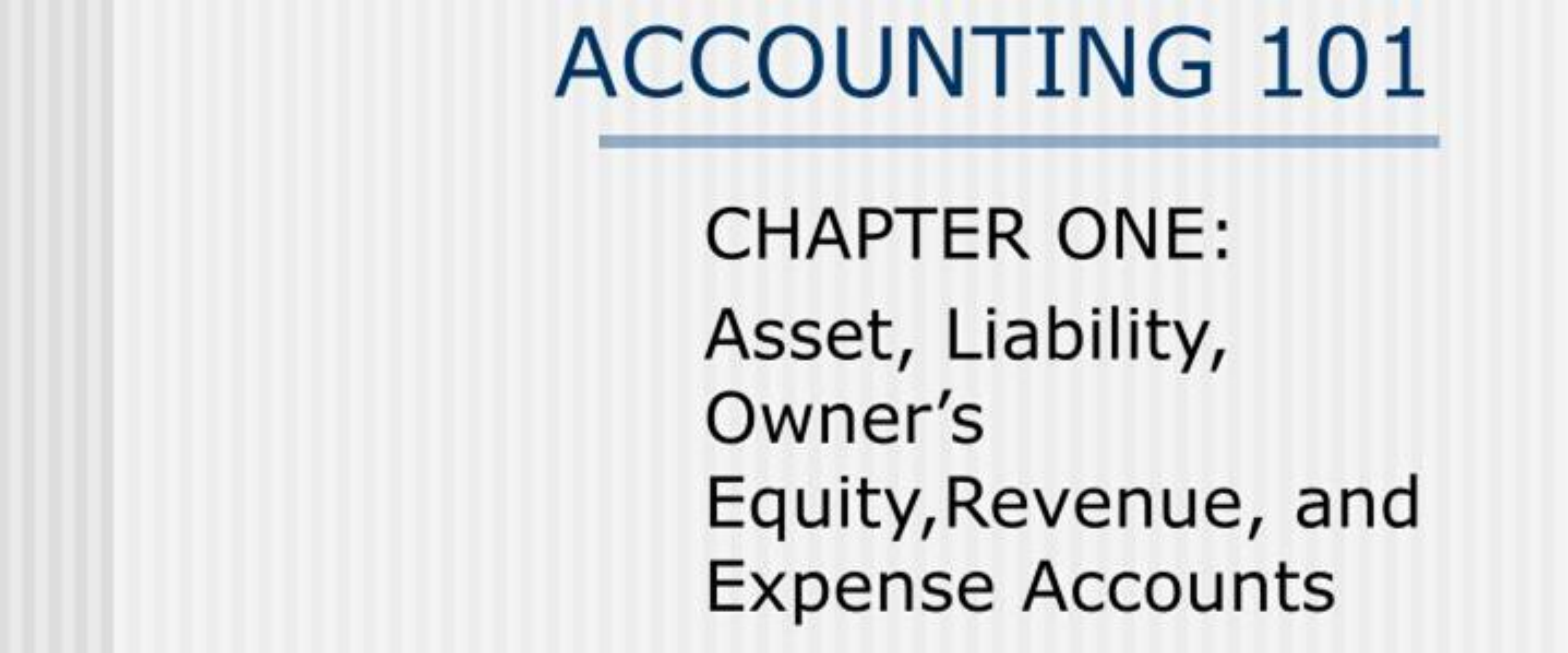
Let’s Teach Them AppSec

If they know how attackers think, they’ll be able to test like an attacker - Hack Yourself!

- Here’s 11ty Billion new Acronyms to learn
- Also, let’s talk about risk
- But wait before that, do you know the Internet is a bad place?
- If you have done a Dev Security Training program, who usually gets selected to go?

“We Need to Model Out a Price Increase”

Have you ever seen the FP&A team teach the basics of accounting to the Exec Team



ACCOUNTING 101

CHAPTER ONE:

Asset, Liability, Owner’s Equity,Revenue, and Expense Accounts

There are Good AppSec Dev Tools Out There

Developer native tools (in context, how they work)

- Snyk
- Fossa
- npm audit
- GitHub (package inspection, PR Bot)
- At least one more coming (cough cough) ;)

Problem Two: Job Roles

We are misaligned

EVERYTHING IS

BROKEN!

Hey! I broke the crap out of your thing. Cool huh!

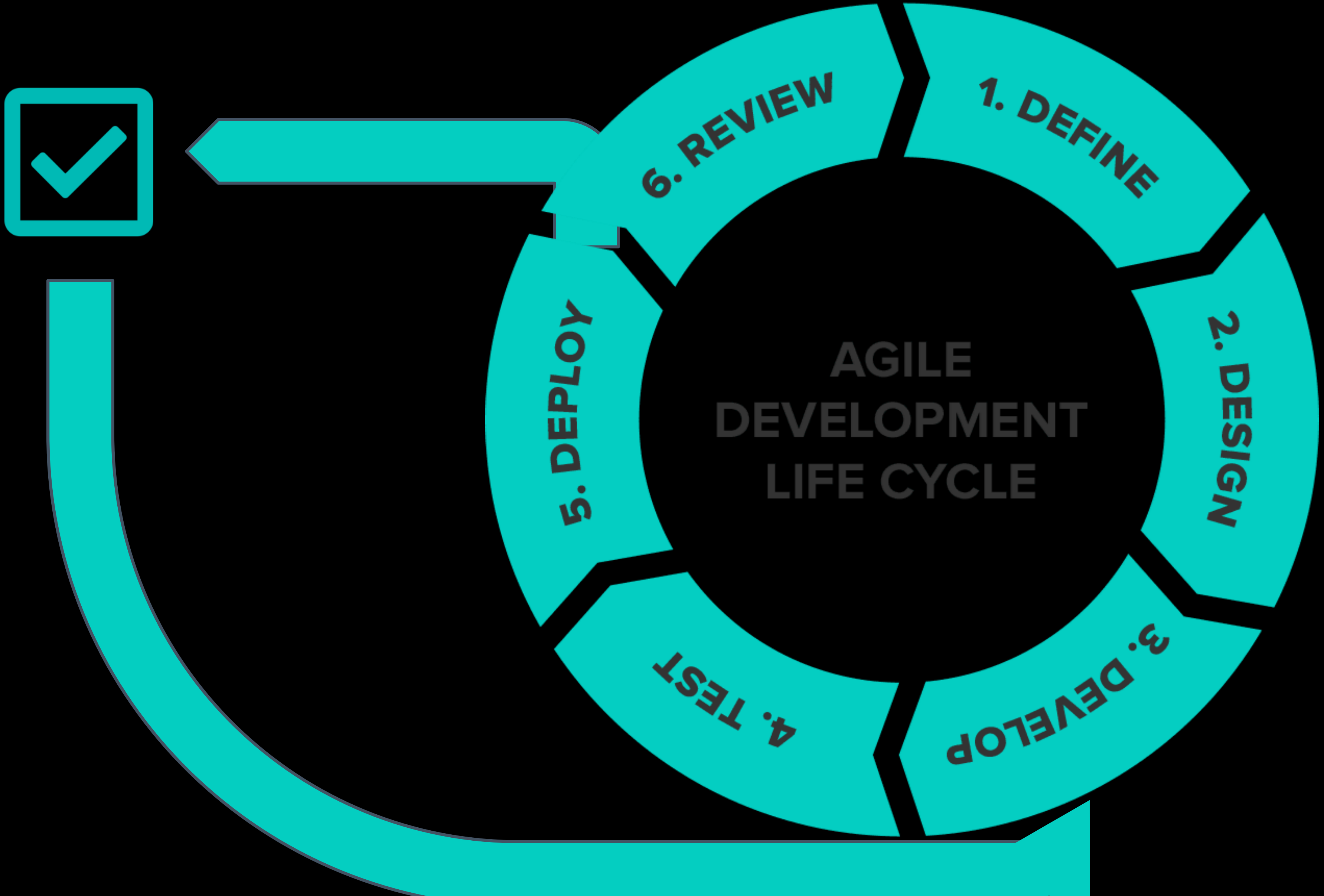
We often forget we play on the same team, but have different roles

- Security teams are often **super** excited when they find an issue, especially a hard to find one

And they should be because that’s the job, but remember what you just found is probably expensive

The AppSec Expense

"Ohh hey. We'd like to review this thing... also, when did you deploy this?"



You say it’s the internal directory huh

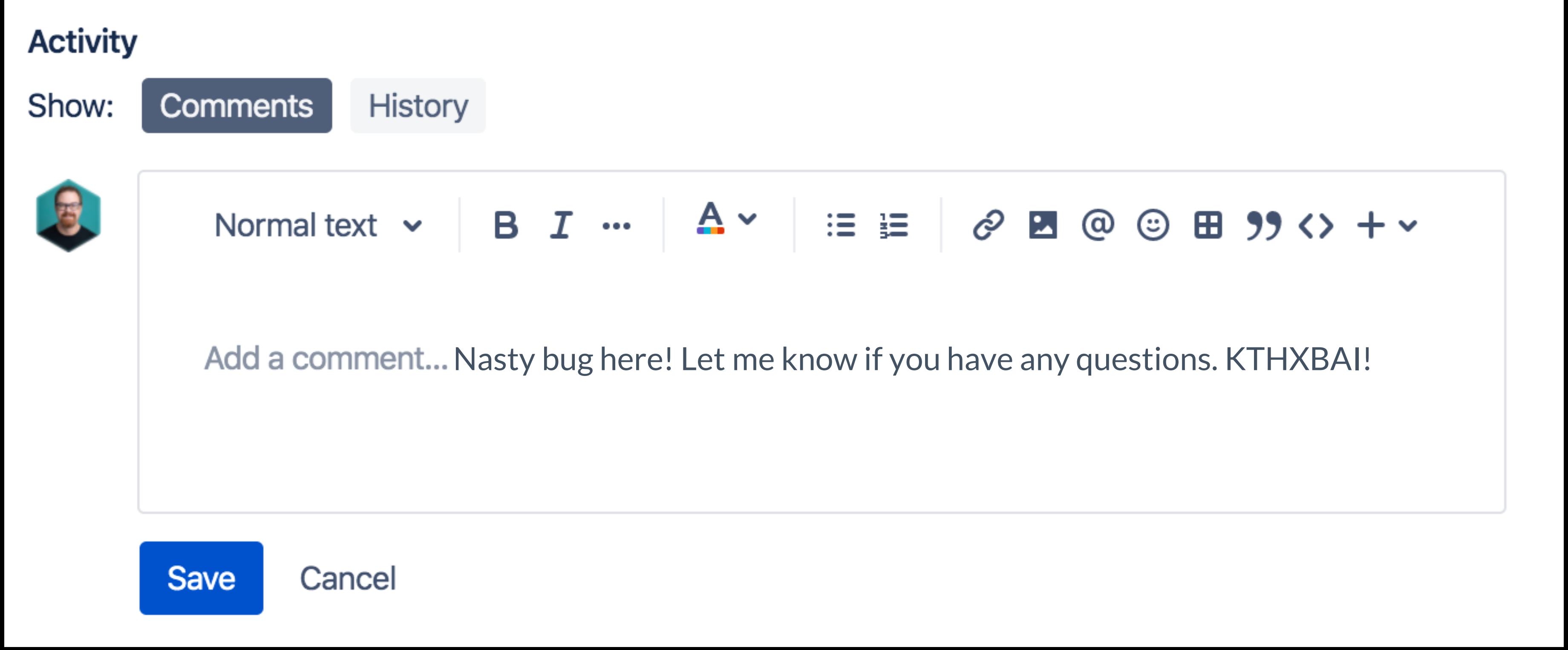
Where are we spending our effort and to what outcome?

- What is the context of the thing we are testing? Is it high value or is it largely a nice to have?

“Should I even have been testing this thing?”

Ante Over - With Engineering

- Ticket hucking and hoping somehow the story beats out a feature or other thing that brings customer/money.



Add a comment... Nasty bug here! Let me know if you have any questions. KTHXBAI!

Problem Three: Just Say Yes

We are misaligned

FIX ALL THE THINGS!

FIX ALL THE THINGS!

Fix The Right Thing
Bed-Stuy
No one told me
SAL

The Chase to Perfection

- Find 11ty Billion issues -> We have to fix all of these!
- Why? What is the actual risk in the context of the business?
- What if your QA filled 1,000 tickets for bugs that are unlikely to degrade user experience?

"I've never had a satisfying conversation on why a security issue is ever more important than a feature. Ever." - Product VP

Be a Good Improv Partner

- YES! And...
- The business is there to take risks
- What risks should we be taking and how can everyone know

Closing Thoughts

- Democratize Security through the Org, with easy to use tools and information.
- “Security is Everyone's responsibility” is a cop out
- Security == Quality | Engineers Care about Code They Write
- Just say no! (to your inner if “anything is wrong it’s all wrong”)
- Support Dev teams. Buy them some tacos.

Useful Links

- Secure Code Example - https://github.com/kelseyhightower/nocode
- Library analysis tools
  - https://docs.npmjs.com/cli/audit
  - https://snyk.io
  - https://fossa.com

- Nifty Startup - https://stackhawk.com

Thanks!