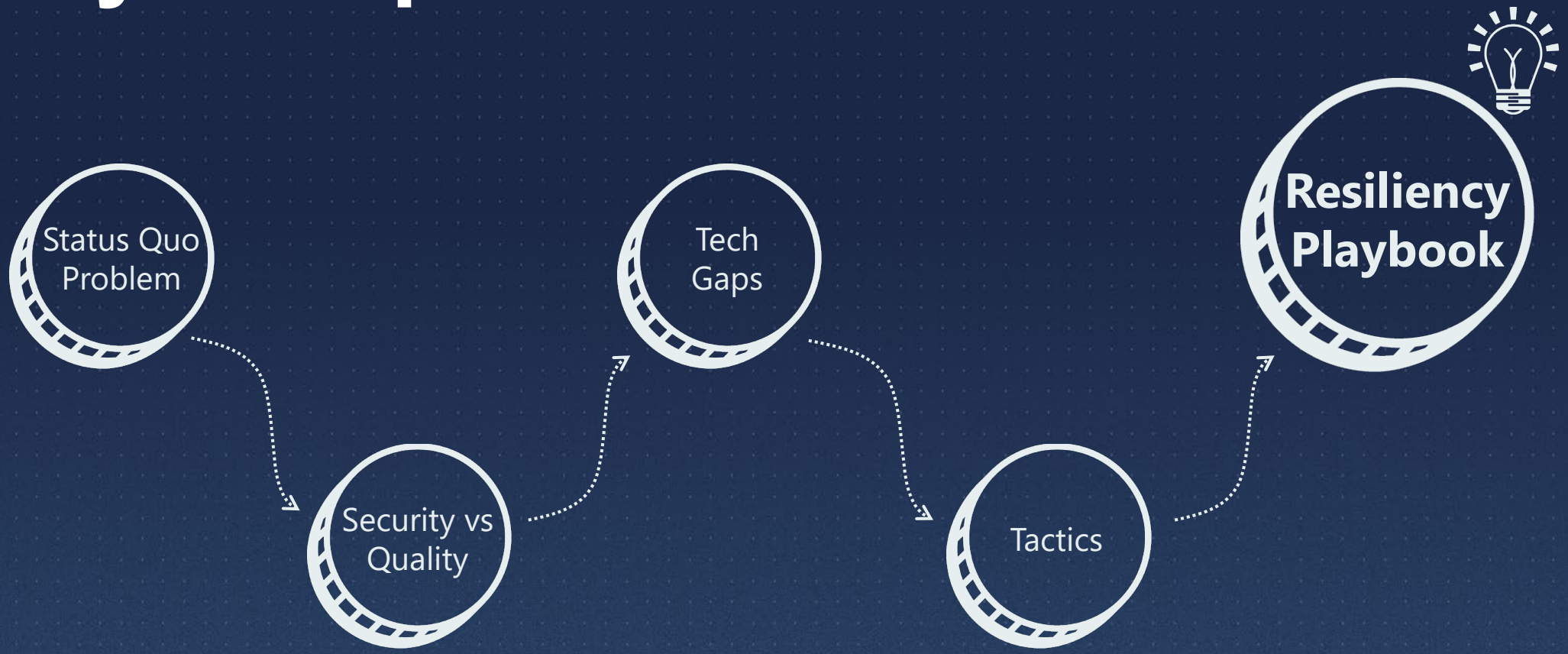
FATHOM5
BRILLIANT MACHINES

To Resiliency and Beyond!

How to engineer survivable systems

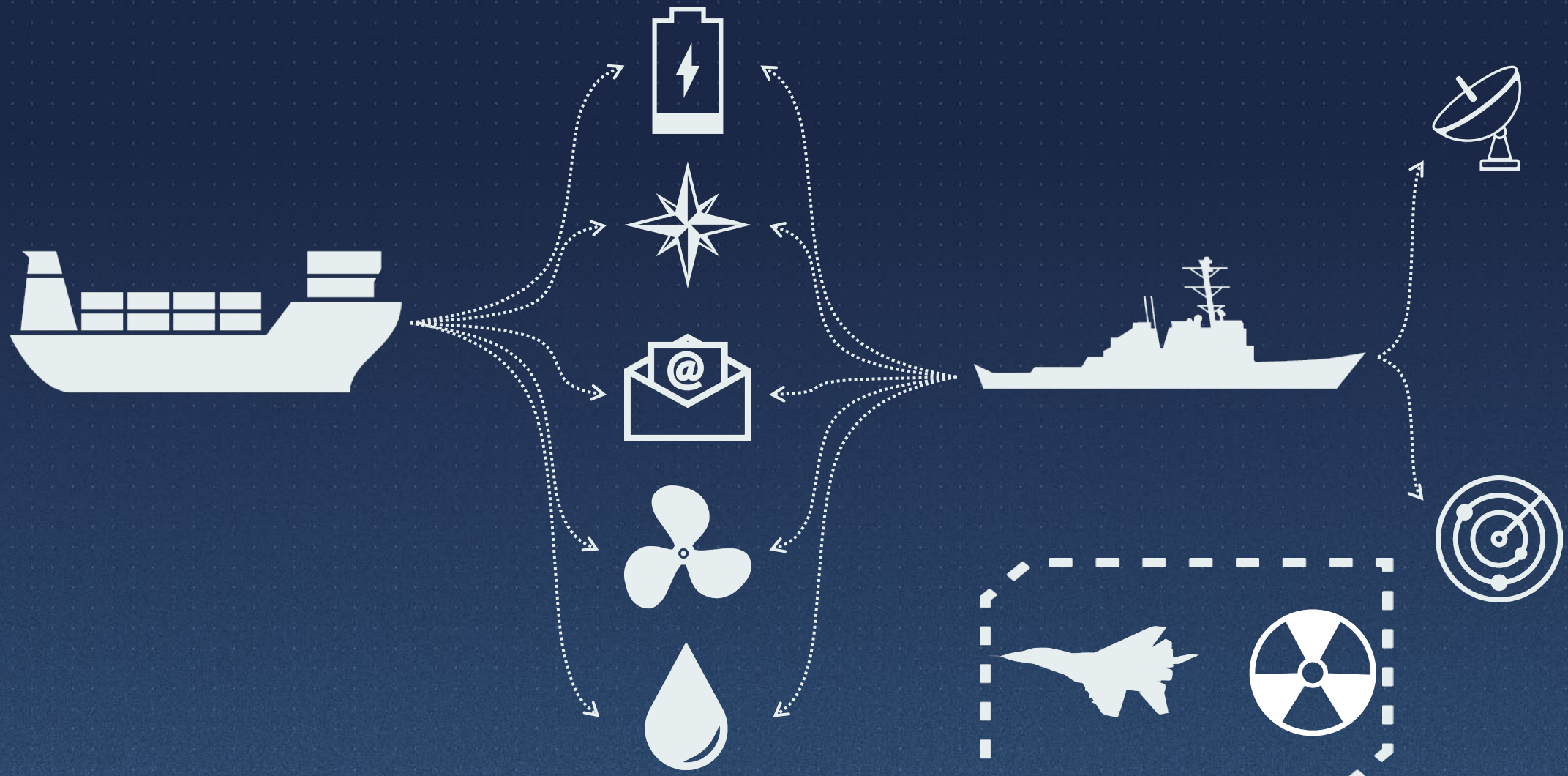
Today's Map


Status Quo Problem
Security vs Quality
Tech Gaps
Tactics
Resiliency Playbook

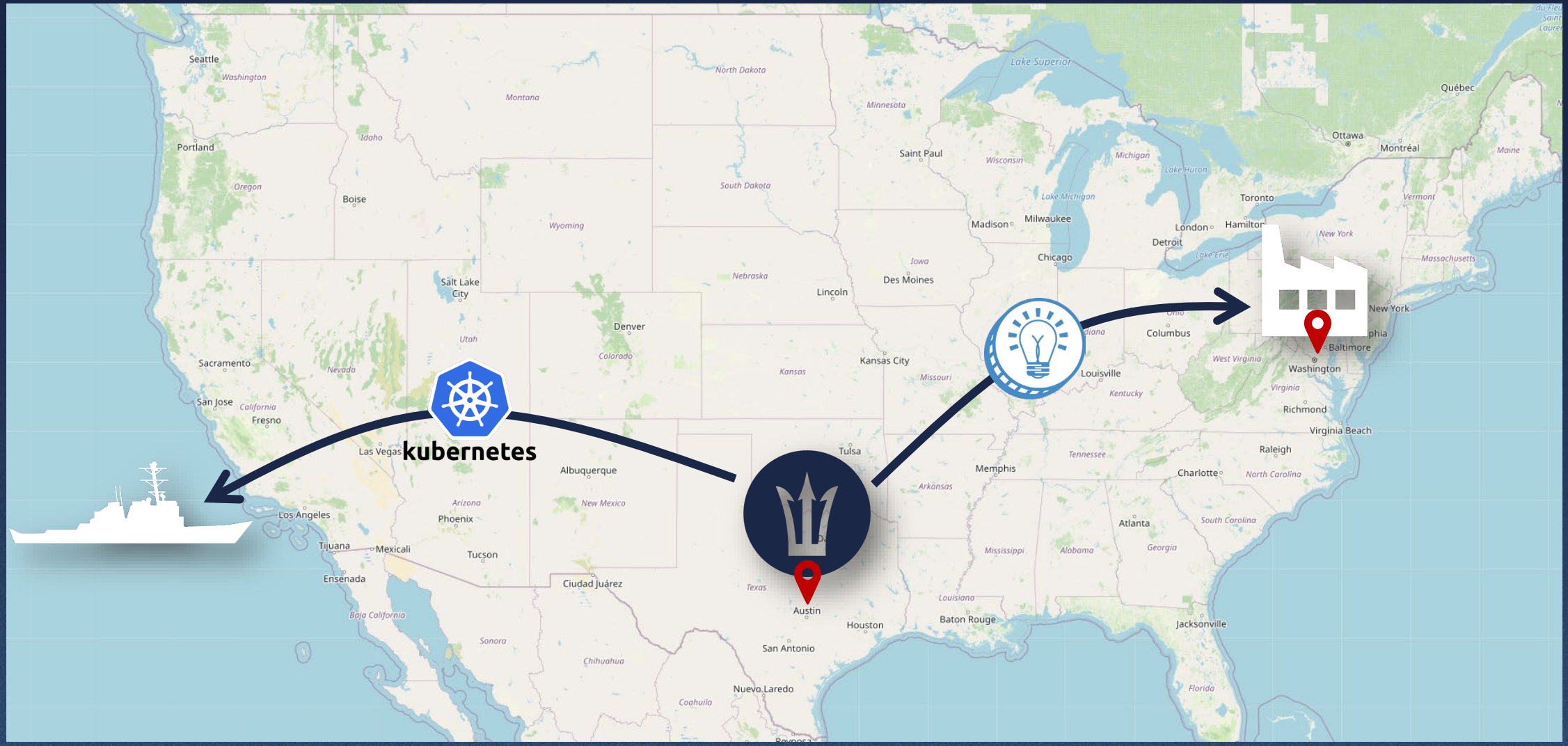
About Me

Matt Wiseman
Managing Principal
Cyber Resilient Engineering
at Fathom5

About the Maritime Domain



About Fathom5




Status Quo Problem
Tech Gaps
Resiliency Playbook
Security vs Quality
Tactics

Why has the status quo failed to protect critical systems?

NIST Interagency Report 8011 (2017)

- *Automation Support for Security Control Assessments*

"predefined control sets have been applied to provide detailed technical requirements **without documenting traceability** of control items to more general requirements"

"many security programs have focused on the individual controls as a **compliance checklist**, with **little consideration given to how the controls work together**"

https://nvlpubs.nist.gov/nistpubs/ir/2017/NIST.IR.8011-1.pdf

Rugged Software (2010)

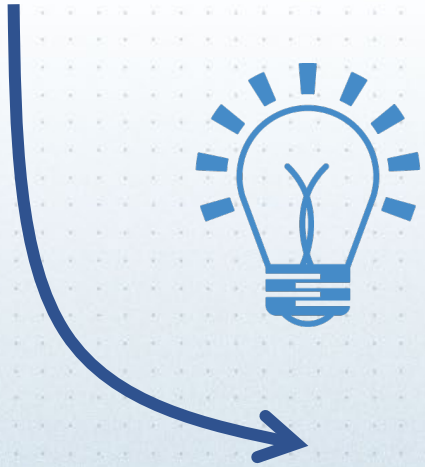
- *Manifesto, Handbook, and Implementation Guide*

"The best projects today perform activities like threat modeling, security architecture, secure coding training, and security testing. However, **it's generally unclear how these activities connect back to the business goals**"

"Frequently these activities simply report vulnerabilities or risks that **do not become part of any sort of coherent security strategy**. In fact, most of these efforts **create no lasting value**, and are simply repeated from scratch after some period of time."

https://ruggedsoftware.org/

Why has the status quo failed to protect critical systems?



Requirements need to be linked to business/ mission value

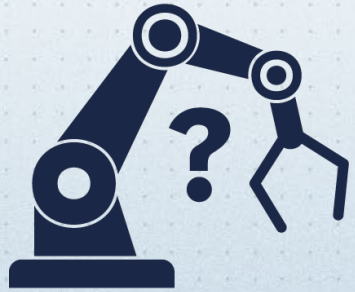

Status Quo Problem
Security vs Quality
Tech Gaps
Tactics
Resiliency Playbook

Is security a “functional” or “non-functional” requirement?

First, Some Definitions

Functional

- Unit of work
- What a thing does



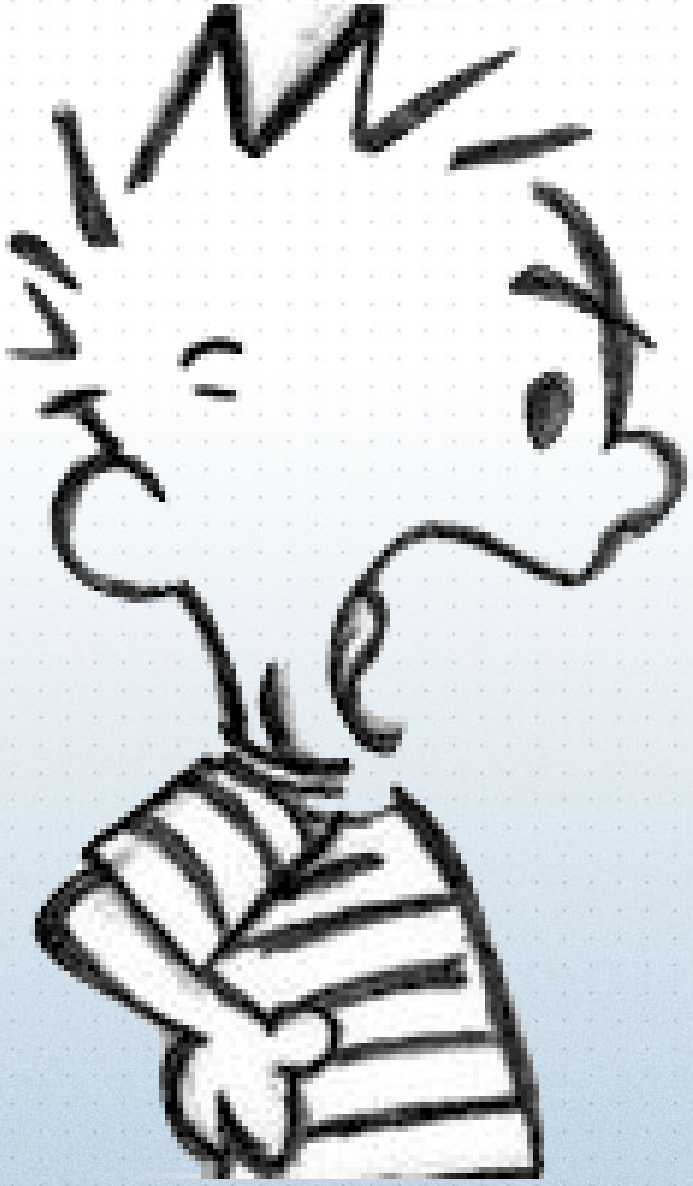
Non-functional

- Measure of performance
- How well a thing works

Quality is Non-Functional

- Reliability
- Maintainability
- Usability
- Availability
- Portability
- “other –ilities…”

What about Security?



Security is Non-Functional

- Confidentiality
- Integrity
- Availability
- Authenticity
- Non-Repudiation

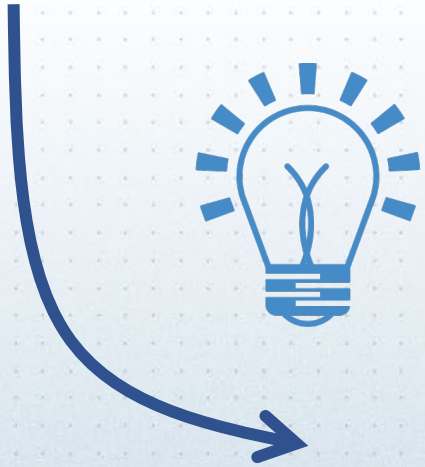
Security is Functional

- Cryptography
- Secrets Management
- Mutual Authentication
- Logging
- Auditing
- Intrusion Prevention

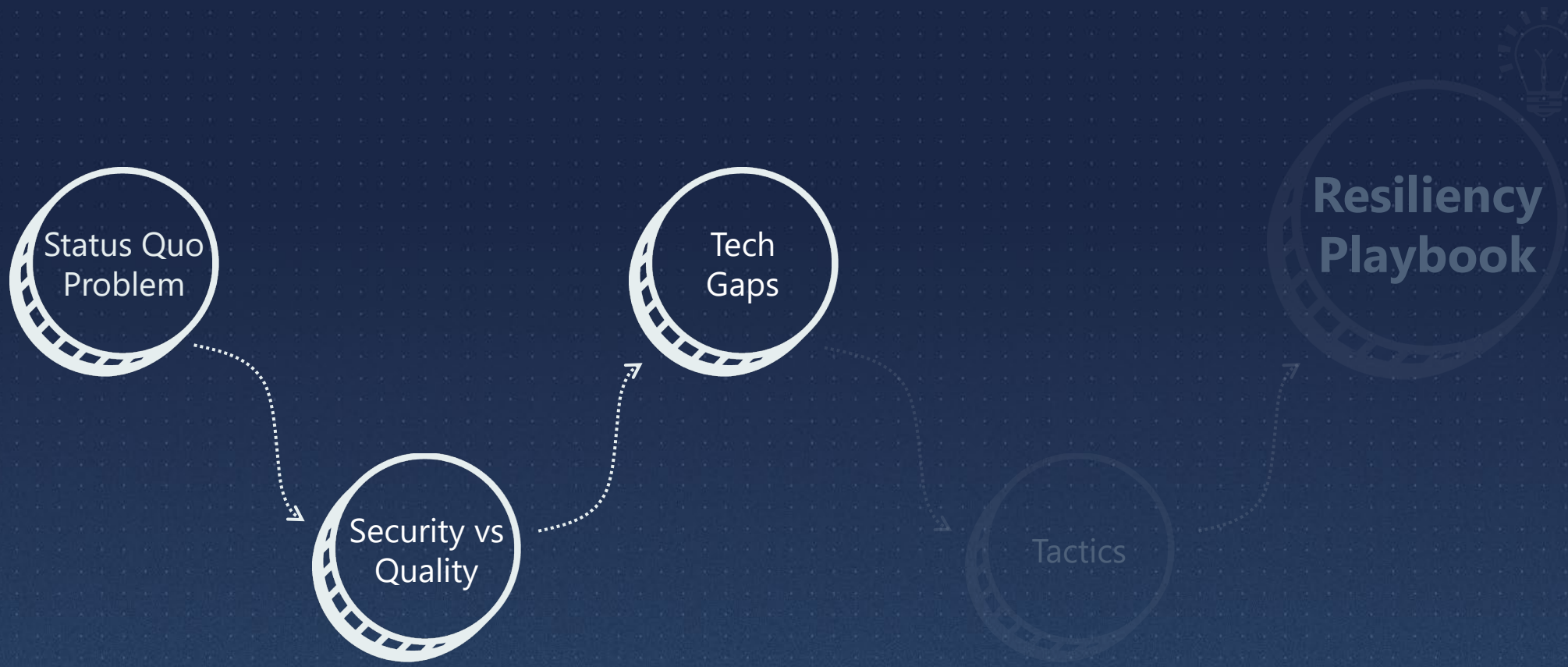
…

https://www.flickr.com/photos/thisisrorydean/8000230363/

Is security a “functional” or “non-functional” requirement?



Both! There are capabilities to build (for unique users) and metrics to assess


Status Quo Problem
Security vs Quality
Tech Gaps
Tactics
Resiliency Playbook

Is the industry balancing investments in the right tech?

Today's Most Popular Tools

- SAST
- DAST
- SCA
- SBOM Generators
- Image Scanners
- Network Pen Test Tools

These tools reinforce a non-functional security stereotype when applied in isolation!
(i.e. nothing to engineer, just findings to track and patch)

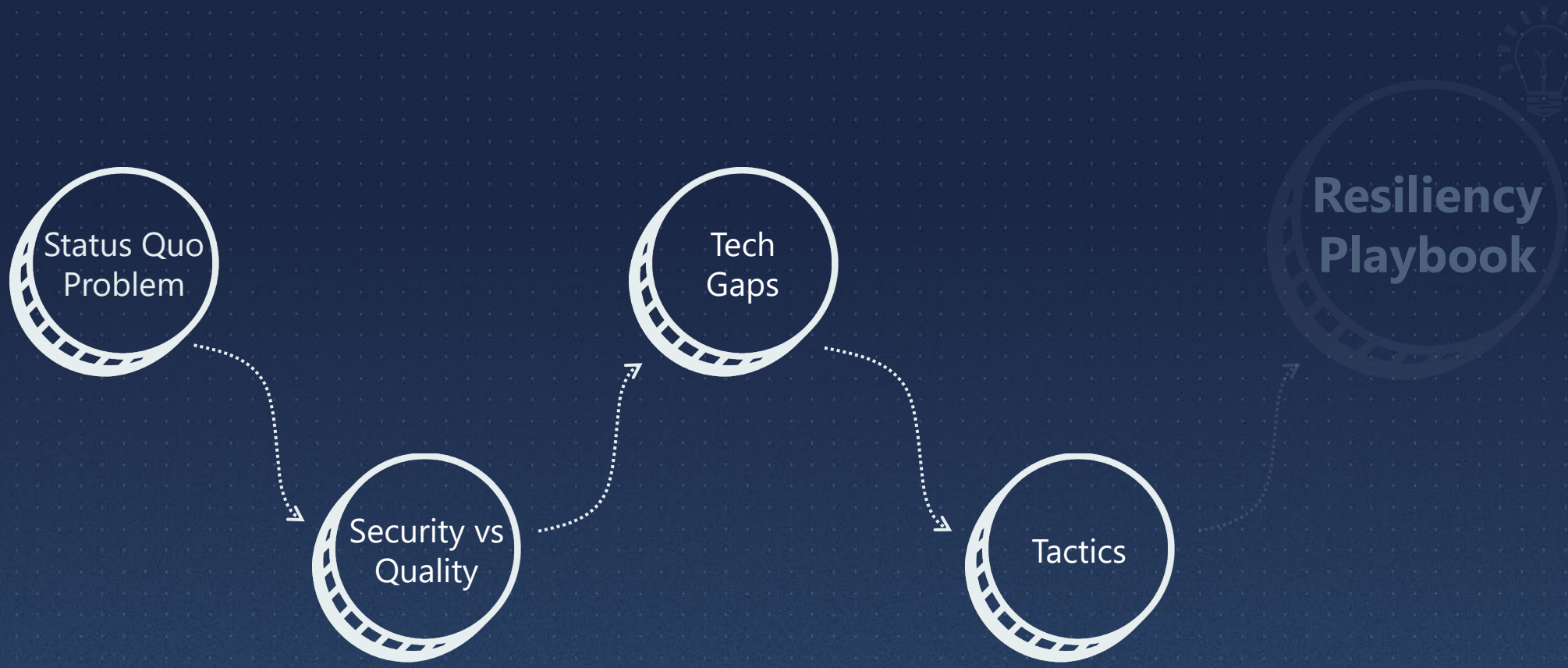
Tomorrow's Most Popular Tools (a bet)

- Hardened Libs/Archs
- Extensible Fuzzing Frameworks
- Debloaters
- Declarative Security Engines/Runbooks
- Resiliency Models
- AI-Driven Pen Test Agents

Today's tools are still necessary, but Security Engineering would benefit from more that lend themselves to functional development!

Is the industry balancing investments in the right tech?

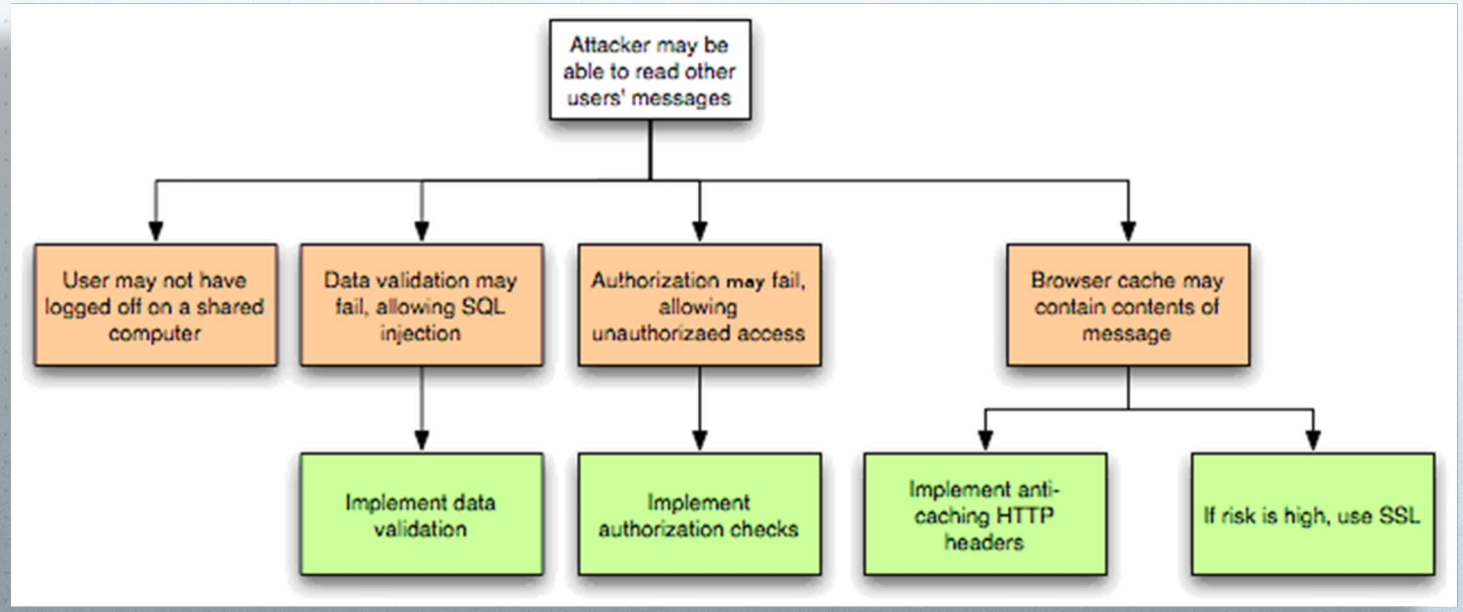
Need to shift focus to system resiliency by design over managing CVEs/CWEs


Status Quo Problem
Security vs Quality
Tech Gaps
Tactics
Resiliency Playbook

What can we do to start engineering for resilience?

Engineering for Resilience

- "What functions do I need to engineer into my system to protect, detect, respond, and recover from cyber events?"
  - Threat Modeling starts providing answers…

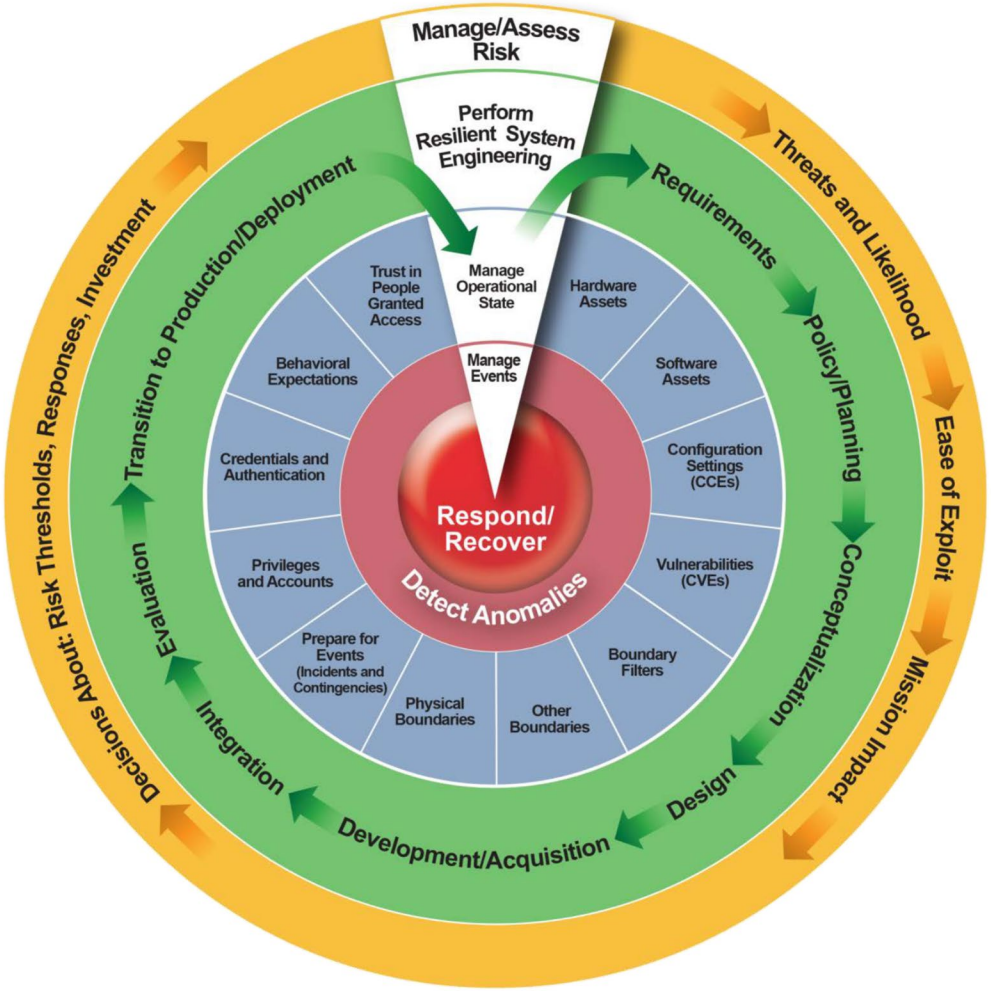


https://owasp.org/www-community/Threat_Modeling_Process

NISTIR 8011

"The four abstraction layers support integrated systems engineering by making the desired results of a security program **clear and measurable** at a concrete level. This, in turn, makes the results more understandable to non-security experts and thereby easier to **link to desired business/mission results**."

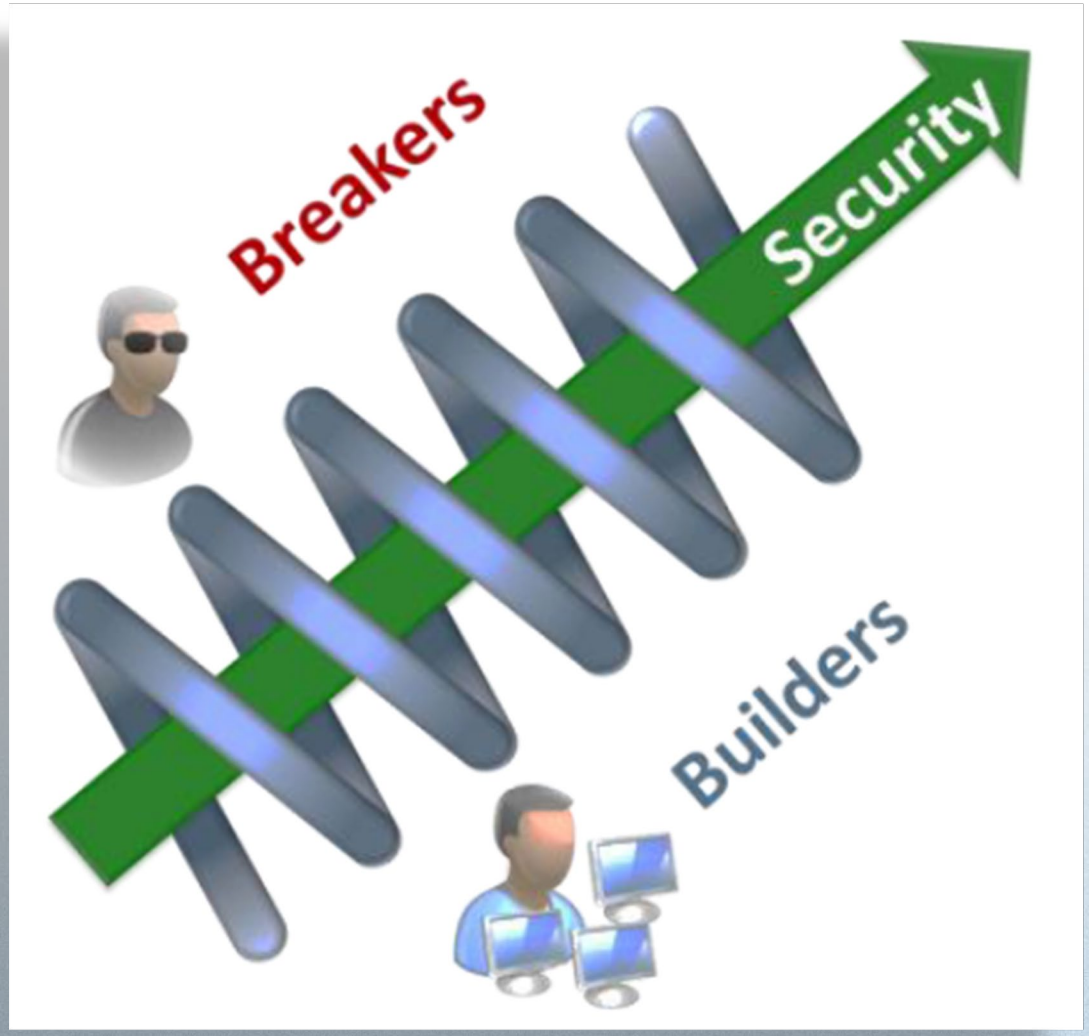
1) Attack Step Layer
2) Functional Capability Layer
3) Sub-Capability Layer
4) Control Item Layer



https://nvlpubs.nist.gov/nistpubs/ir/2017/NIST.IR.8011-1.pdf

Rugged Software

"rugged describes staying ahead of the threat over time. Rugged organizations create secure code as a byproduct of their culture. You are rugged because you run the gauntlet, instrument your organization and your code, constantly experiment to see if anything breaks, and survive the process of **hardening yourself through real-world experience**. Rugged organizations produce rugged code designed to **withstand not just today's threat, but future challenges as well**."



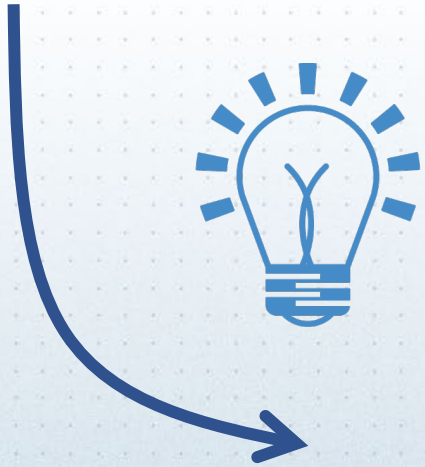
https://raw.githubusercontent.com/rugged-software/rugged-software.github.io/master/documents/Rugged-Handbook-v7.pdf

Nat'l Cyber Strat

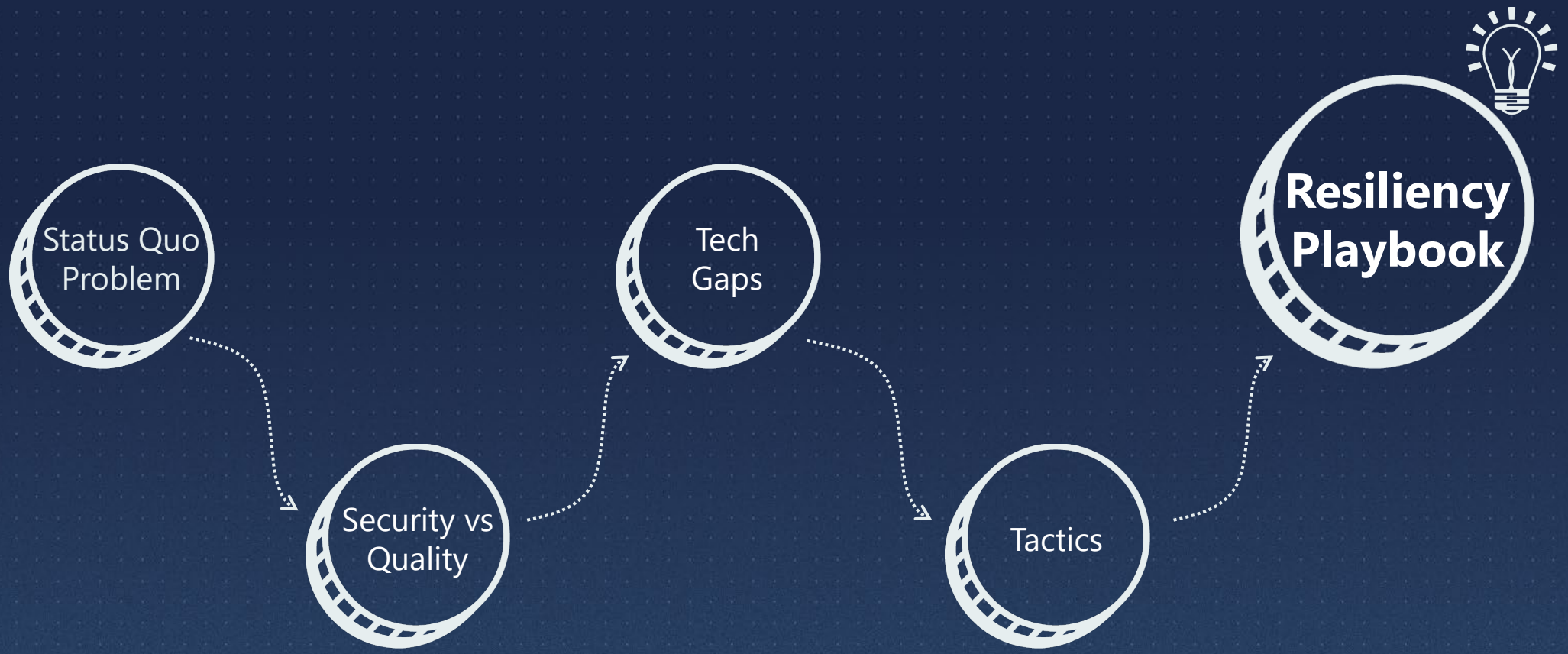
"We will complement our efforts to out-innovate other countries with focused, coordinated action to optimize critical and emerging technologies for cybersecurity **as they are developed and deployed**. We will ensure that **resilience is not a discretionary element of new technical capabilities but a commercially viable element** of the innovation and deployment process."

https://www.whitehouse.gov/wp-content/uploads/2023/03/National-Cybersecurity-Strategy-2023.pdf

What can we do to start engineering for resilience?



Assume compromise. Put yourself in the hacker's shoes. Keep it outcome focused.


Status Quo Problem
Security vs Quality
Tech Gaps
Tactics
Resiliency Playbook

How do we operationalize this?

Our Cyber Resiliency Playbook

- ❑Phase 1: Release the CVEs!

- ❑Phase 2: Speed-Run MVS!

- ❑Phase 3+: Malicious BDD ‘til Infinity!

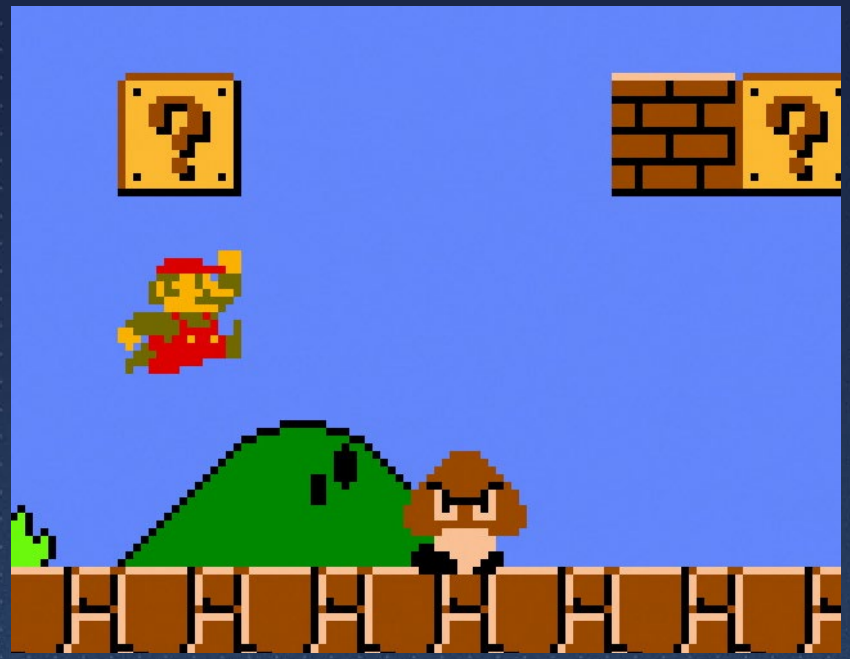
(1) Release the CVEs!

https://www.amazon.com/Clash-Titans-Sam-Worthington/dp/B002ZG977Y

We start by surfacing the known vulnerabilities and weaknesses, giving us the opportunity to kill risk at the source

- ✓SAST
- ✓Image Scanning
- ✓SBOM Generation + Scanning
- ✓CVSS x EPSS x SSVC for a Risk-Based Approach to Vulnerability Triage

(2) Speed-Run Min Viable Security!



https://www.polygon.com/2014/6/5/5784190/mario-maker-nintendo-e3-2014-rumor

We prioritize the top 20% of functional specifications that buy down 80% of the attack surface, and deploy to prod

- ✓Secure-by-Design/Default (per CISA)
- ✓Pass Compliance Muster (can't deploy an MVP otherwise)

https://www.cisa.gov/sites/default/files/2023-04/principles_approaches_for_security-by-design-default_508_0.pdf

(3) Malicious BDD ‘til Infinity!

https://upload.wikimedia.org/wikipedia/en/e/e1/93tilinfinity.jpg

We put on our white hats to stress test system performance under malicious activity, and continuously engineer improvements

- ✓System Modeling and Criticality Analysis
- ✓Threat Modeling with Malicious Behavior Statements
- ✓White Box Software Penetration Testing
- ✓Engineer Protection, Detection, Response, and Recovery Capabilities
- ✓Re-Test & Repeat

Summary

- Security needs to connect back to key mission outcomes to yield lasting value
- Security Engineering is both functional and non-functional
- DevSecOps culture is key to implementing it
- More investments are needed in functional security technology
- The more we break and fix, the higher our system resiliency can become with each new release
- We can demonstrate **clear measurable improvement in mission performance** from cybersecurity investments

Questions? Interested? Contact Me

Matt Wiseman
cyber@fathom5.co