Chaos and Reliability:

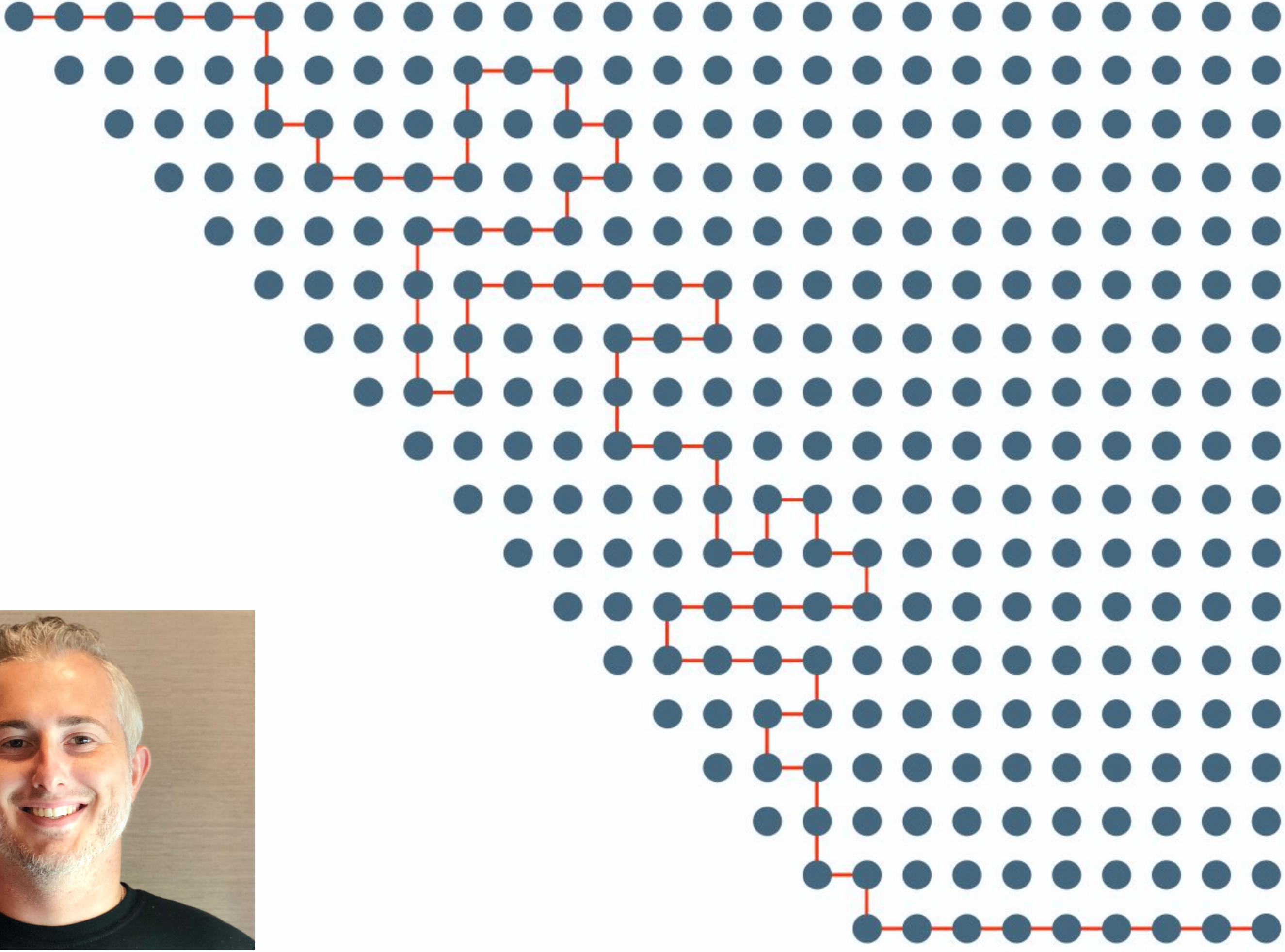
A Surprising Friendship in the Enterprise

Courtney Nash (Verica)

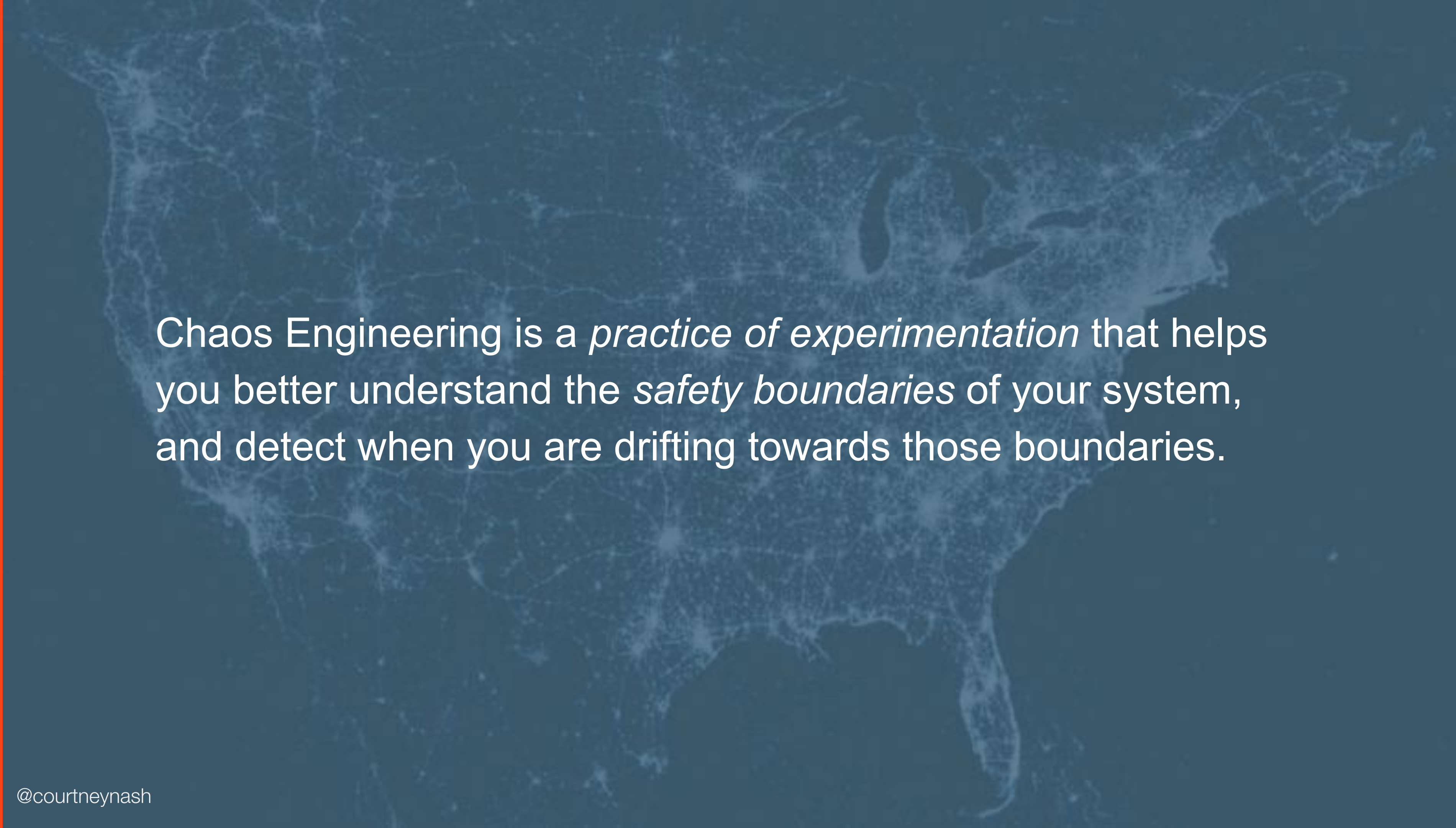
Troy Koss (Capital One)

Devops Enterprise Summit 2021

Courtney Nash
Internet Incident Librarian
Verica



Troy Koss
Director of SRE
Capital One



Chaos Engineering is a *practice of experimentation* that helps you better understand the *safety boundaries* of your system, and detect when you are drifting towards those boundaries.

Chaos Engineering borrows from the same approach to experimentation as clinical trials.

Myth: Chaos Engineering is an "advanced capability."

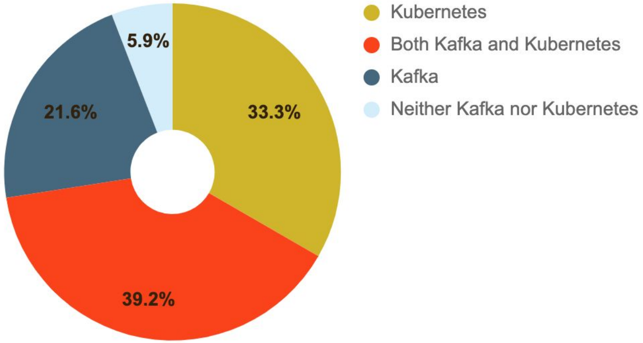
You have to be at a certain level of sophistication with your systems in order to consider undertaking CE.

How It Started

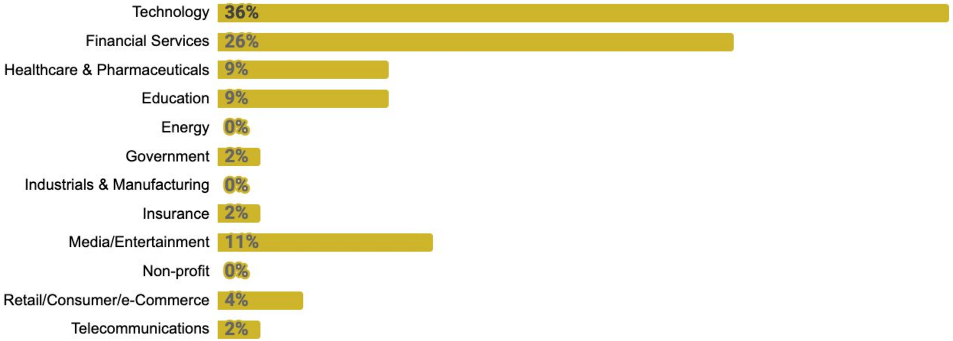
How It's Going

Who Is Ready for/Interested in CE?

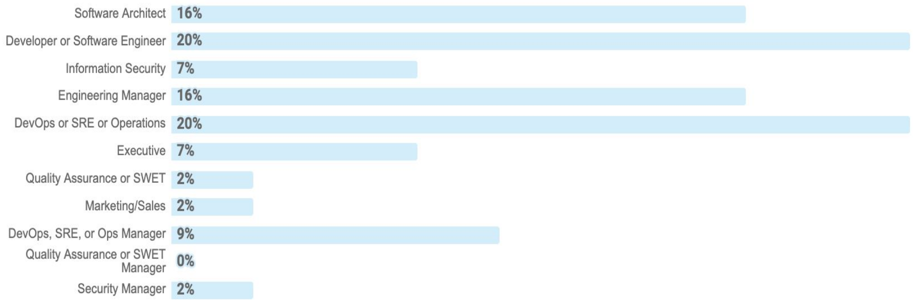
What Are They Using?



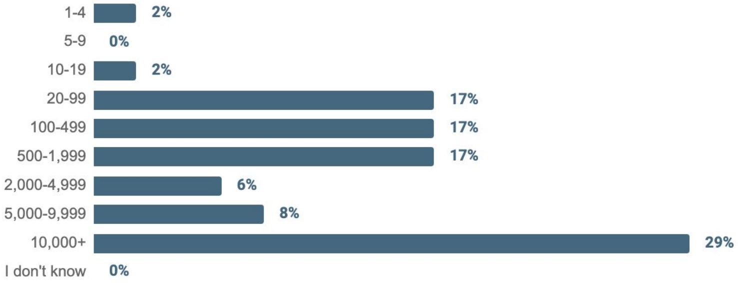
Industry



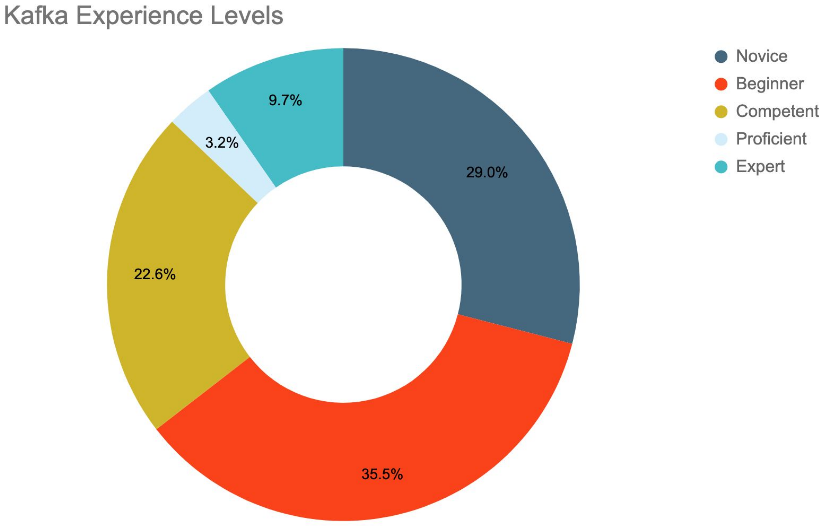
Role



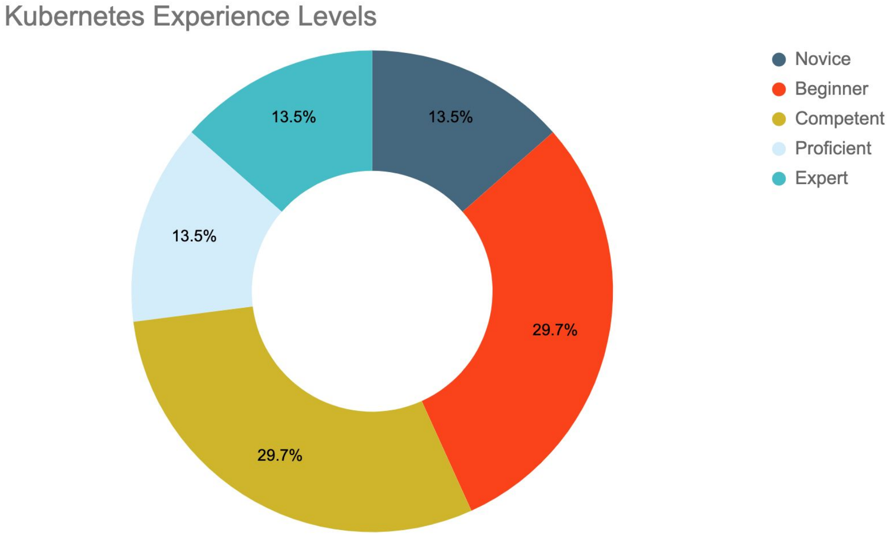
Organization Size



Operator Experience with Kubernetes and Kafka



64% Novice/Beginner



43% Novice/Beginner

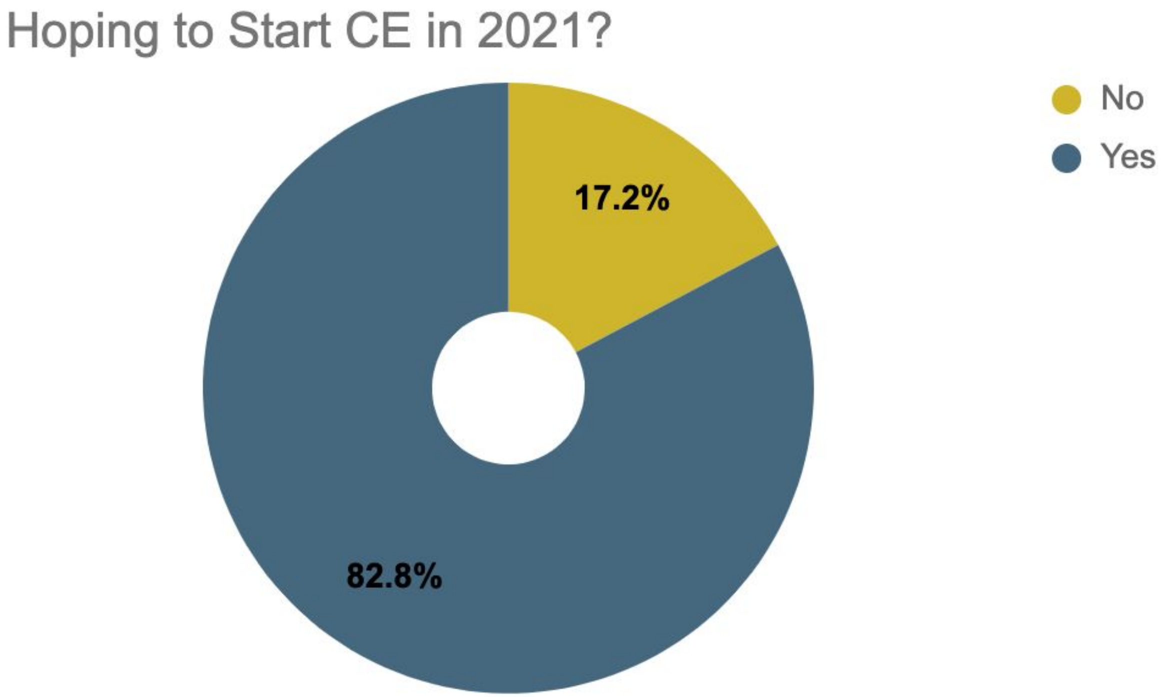
TODO: Enterprise Complexity

- Had a challenge of application modernization at ever increasing scale. At my previous enterprise company, my team embraced a Site Reliability Engineering approach to modernize and elevate our technology stack while retaining the reliability we are known for.
-

Enterprise Clarity

- Contrary to popular belief Chaos engineering and experimentation was a key starting practice to grapple with the sheer scale and complexity of our systems.
- Use case - Kubernetes - run experiments on kubernetes to develop comprehension of dependencies. Artifactory pull time inconsistencies. Findings about configuration and networking complexities in across VPCs

Gaining Confidence with Chaos Engineering



Myth: Chaos Engineering *introduces* (more) chaos into your system

"Our systems are chaotic enough as it is, why make it worse?"

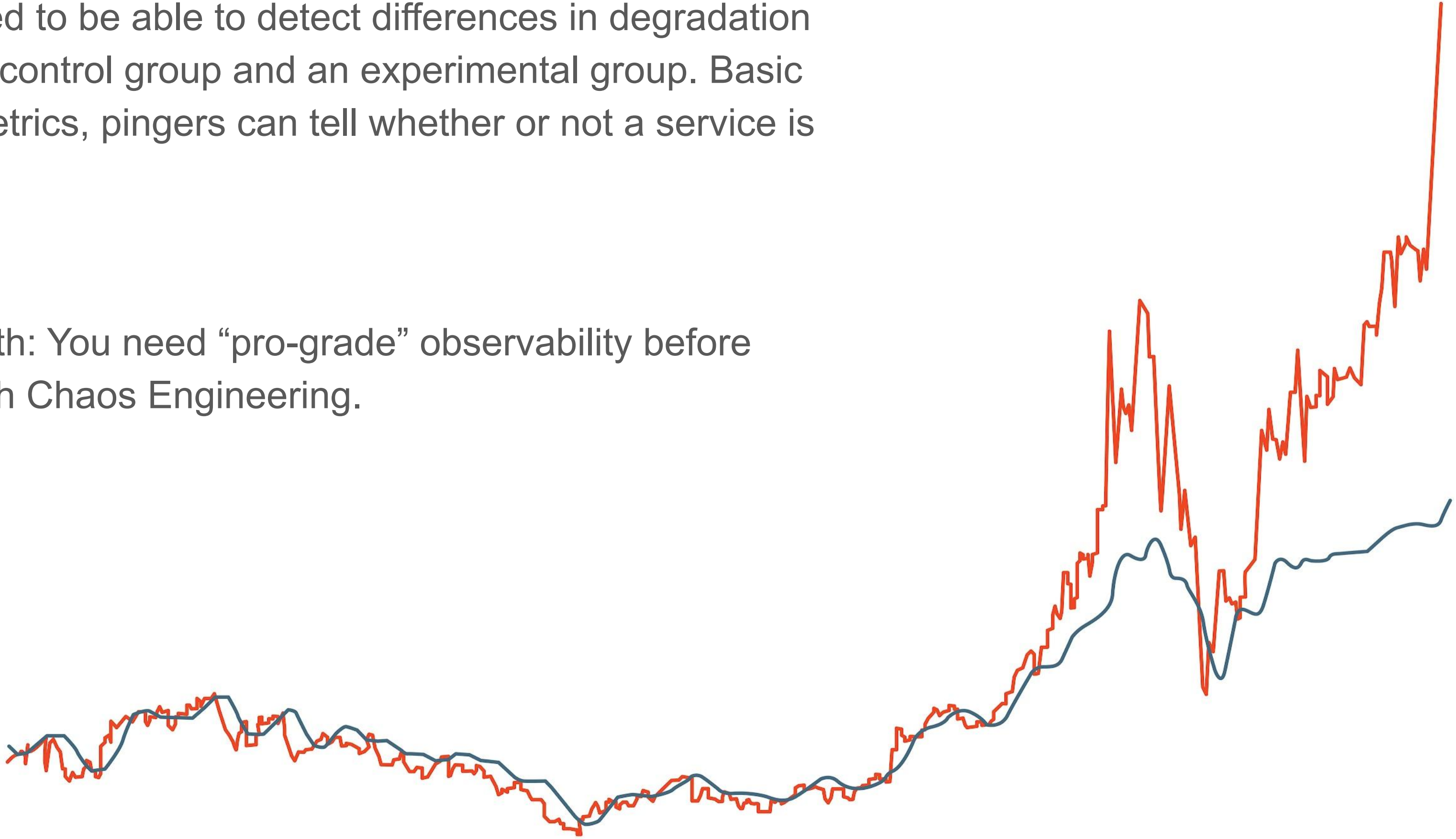
TODO: SLOs to the rescue

- At Verizon teams didn’t have SLOs or a consistent measure to observe. Used chaos eng / cv to adopt and discover SLOs. (pod deployment time?)
- At cap one we’re helping teams define and Leverage SLO/EB to ensure chaos engineering isn’t introducing more chaos. Allows us to experiment safely.
- Looking to use chaos eng / cv to help refine out what our SLOs need to be more accurately (185ms vs 200, 250, 300 etc).
- SLO tool at Capital One

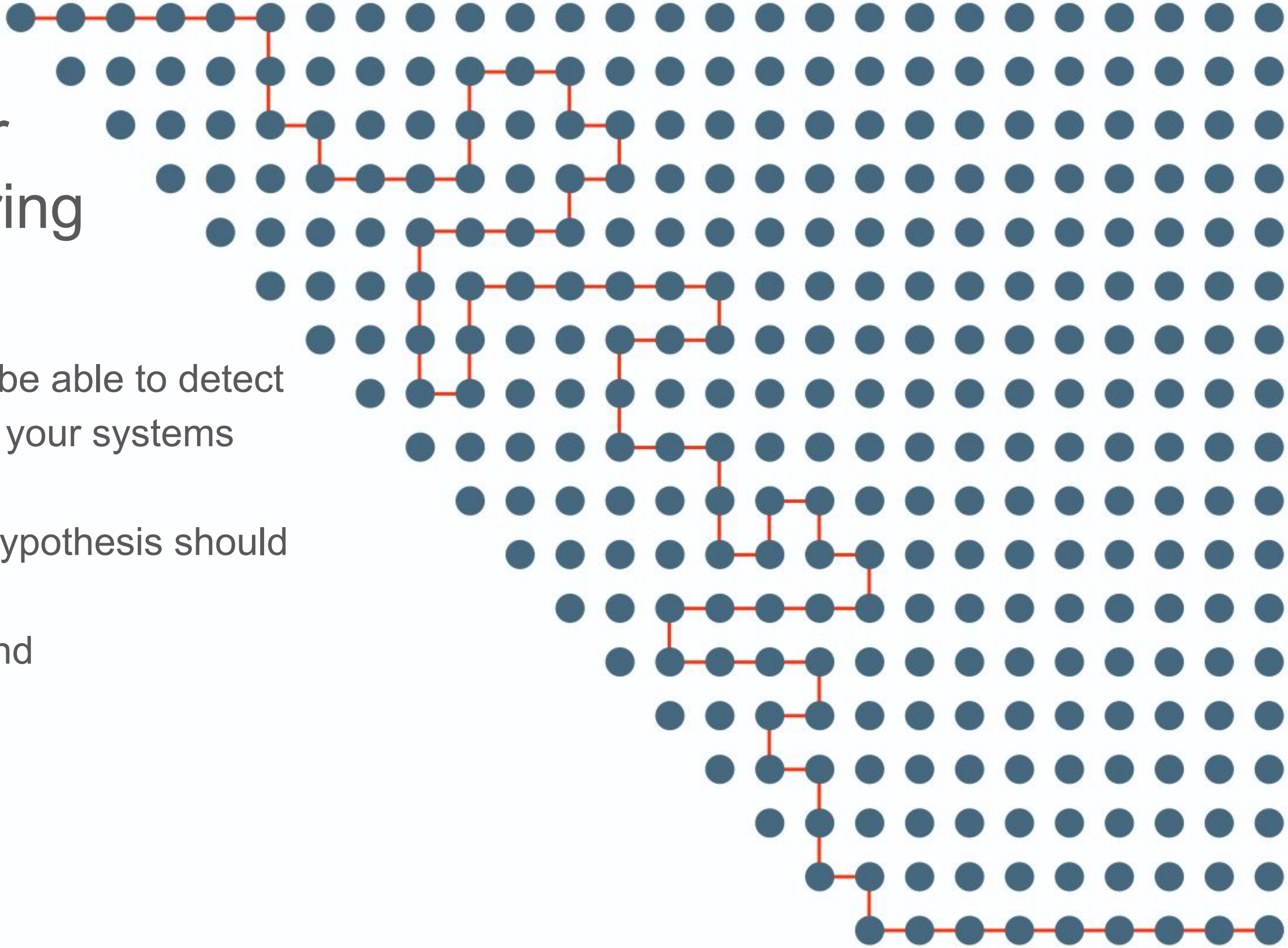
Instrumentation

You do need to be able to detect differences in degradation between a control group and an experimental group. Basic logging, metrics, pingers can tell whether or not a service is up.

Hidden Myth: You need “pro-grade” observability before starting with Chaos Engineering.



Prerequisites for Chaos Engineering



1. **Instrumentation** to be able to detect degraded state(s) in your systems
2. Social **awareness**
3. **Expectations** that hypothesis should be upheld
4. **Alignment** to respond

TODO: Instrumentation

Metrics, Logs, & Alerts to get started. Alerting policy should work as expected. CE also helps with verification of your observability

Social Awareness

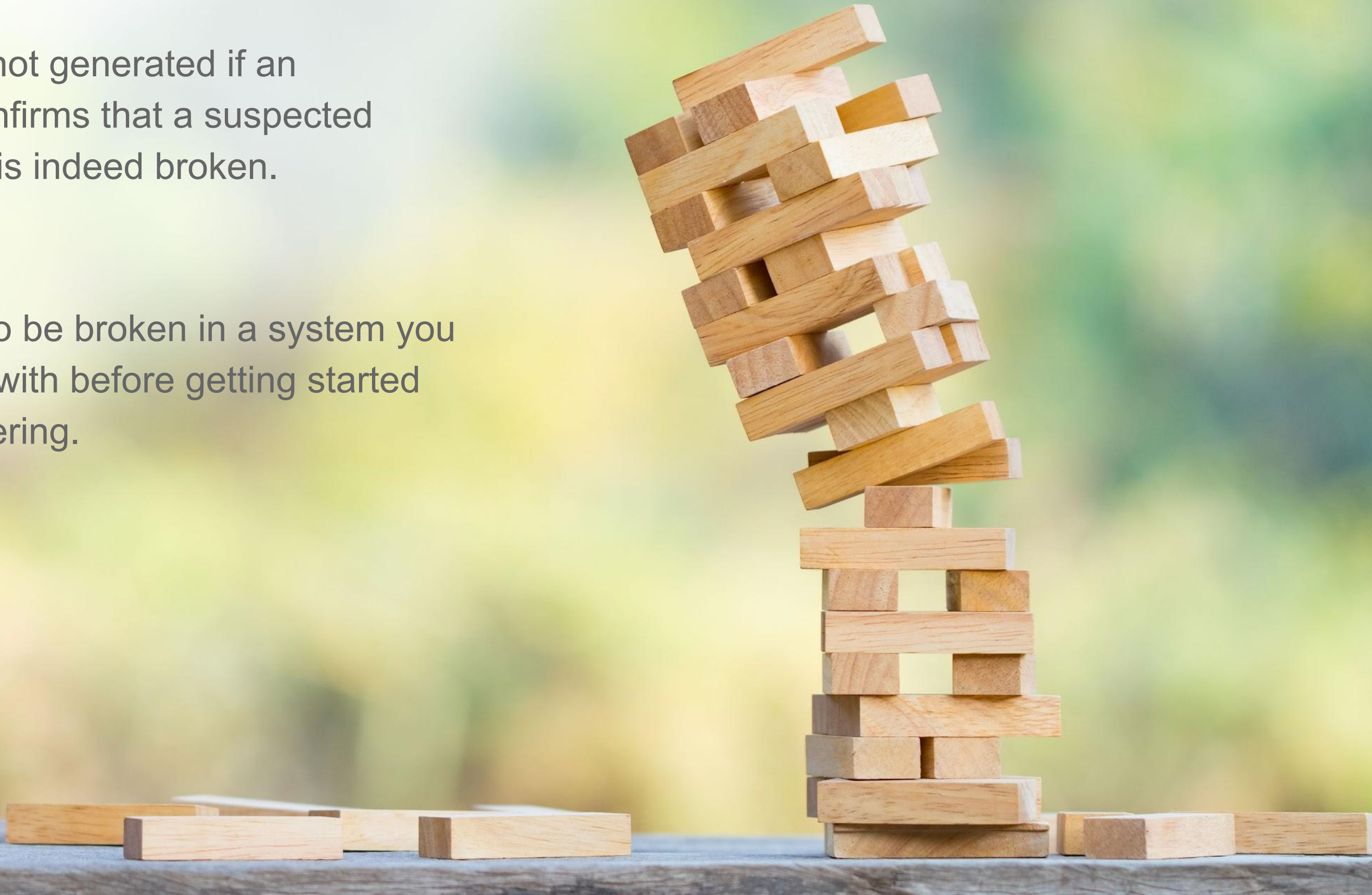
You'll want buy in from anyone who touches or relies on any system you'd include in an experiment.

Running experiments without telling people who'd be involved/impacted will only create animosity and resistance to future CE efforts.

Hypotheses to Uphold

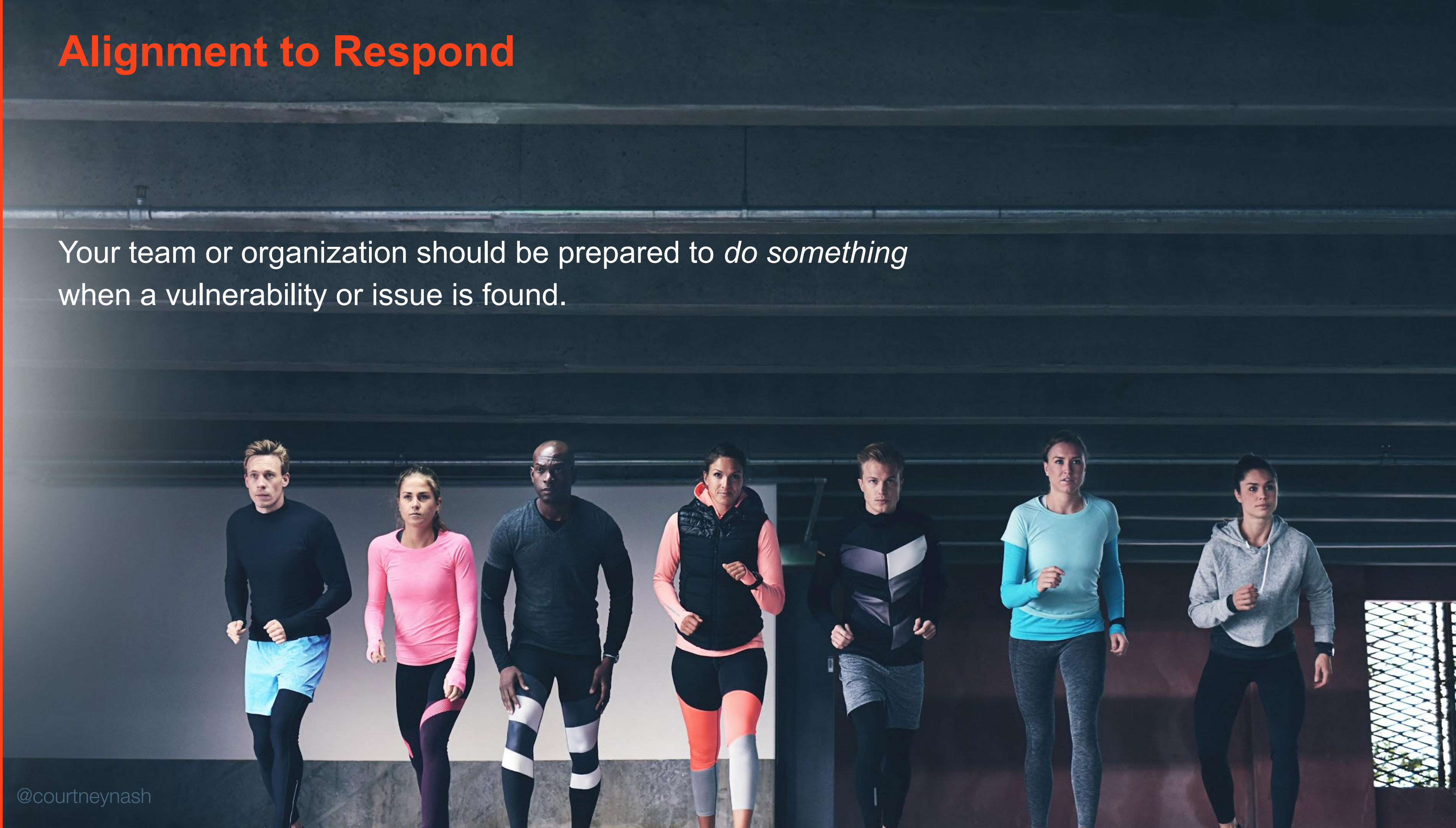
New knowledge is not generated if an experiment only confirms that a suspected broken component is indeed broken.

Fix what is *known* to be broken in a system you plan to experiment with before getting started with Chaos Engineering.



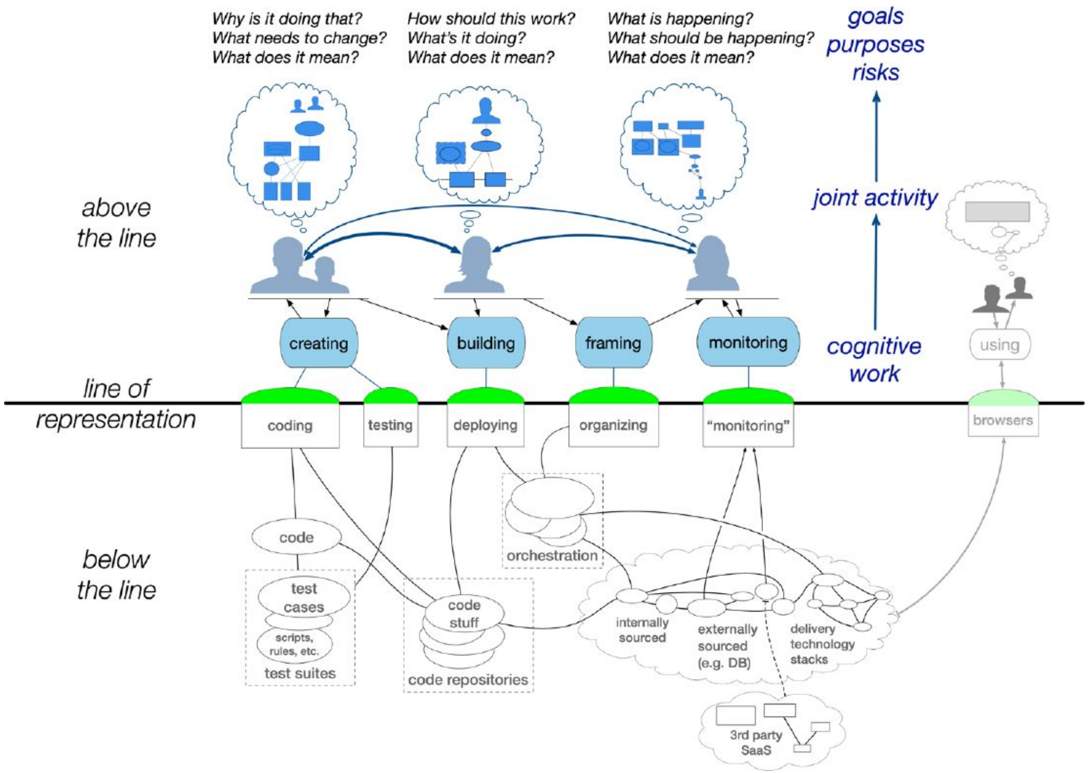
Alignment to Respond

Your team or organization should be prepared to *do something* when a vulnerability or issue is found.



Above the Line

FIGURE 1: ABOVE AND BELOW THE LINE OF REPRESENTATION



Richard I. Cook. 2020. Above the Line, Below the Line: The resilience of Internet-facing systems relies on what is below the line of representation. ACM Queue 17, 6

To properly support Chaos Engineering, your *cultural infrastructure* is as important as your technical infrastructure.

TODO: Chaos Eng is critical to SRE!

These are essentially just SRE practices and the spirit of SRE culture! You have to understand your systems. Observability, Continuous verification and resilient architecture are intents we focus on at Capital one.

Getting Started with Chaos Engineering: Game Days

From a technical point of view, they are easy to set up and don't have to be sophisticated in terms of implementation. Most companies and engineering teams are familiar with the idea of game days as well.

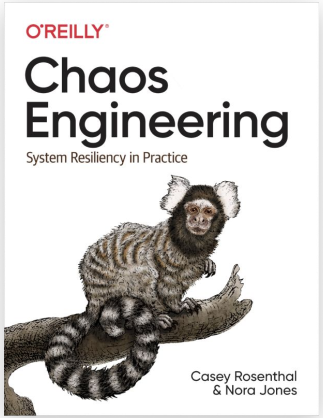
1. Get a bunch of people* in a room who are responsible for a system or set of systems.
2. Shut off a component that the system should be robust enough to tolerate without it.
3. Record the learning outcomes and bring the component back online.

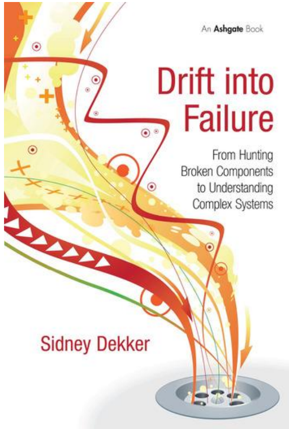
Note: You don't need to run your Game Day in production! Insights can come from conducting experiments first in a staging or test environment.

*Only invite cats if you want to put up with this the entire time.

Chaos Engineering Resources

1. Get the free Chaos Engineering book
2. Subscribe to the Chaos Community Broadcast
3. Read How Complex Systems Fail and Drift Into Failure
4. Stay in touch! Our research is getting underway this year, and we'd love for you to be involved. greetings@verica.io
5. Find Courtney: @courtneynash or courtney@verica.io
6. Find Troy: https://www.linkedin.com/in/troykoss/ troy.koss@capitalone.com




An Ashgate Book
Drift into Failure
From Hunting Broken Components to Understanding Complex Systems
Sidney Dekker

VERICA

Capital One

@courtneynash

Backup