Chaos Architecture

A Cloud Native Availability Model

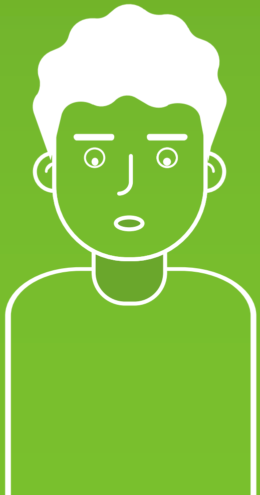
Adrian Cockcroft

@adrianco

AWS VP Cloud Architecture Strategy

As an architect my role wasn’t to tell other people what the architecture should be.

It was to ask awkward questions…



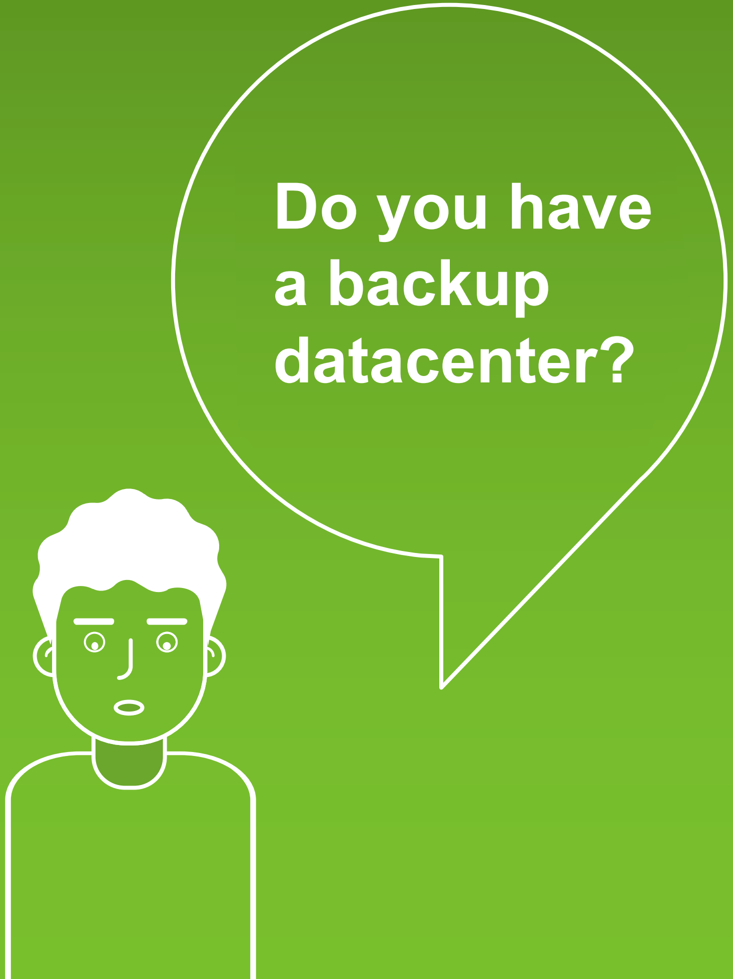
What should your system do when something fails?

- ⬢ Stop?
- ● Carry on with reduced functionality?

If a permissions look up fails, should you stop or continue?

Permissive failure, what’s the real cost of continuing?

See ***Memories, Guesses, and Apologies*** by Pat Helland



How often do you failover apps to it?

How often do you failover the **whole datacenter** at once?

“Availability Theater”

A fairy tale…

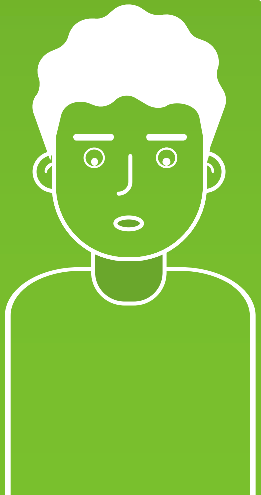
Once upon a time, in theory, if everything works perfectly, we have a plan to survive the disasters we thought of in advance.

How did that work out?

Forgot to renew domain name…	**SaaS vendor**
Didn’t update security certificate and it expired…	**Entertainment site**
Datacenter flooded in hurricane Sandy…	**Finance company, Jersey City**
Whoops!	**YOU, tomorrow**

“You can’t legislate against failure, focus on fast detection and response.”

—Chris Pinkham



How do you know that your system works at all?

How is it supposed to recover after the failure goes away?

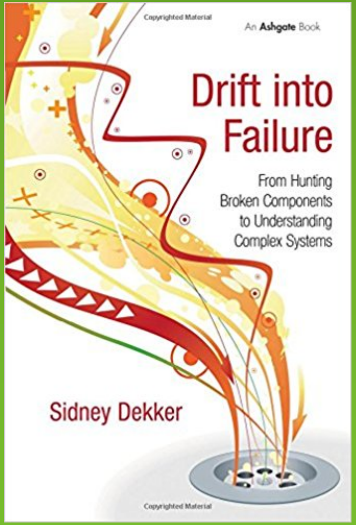
acmqueue

The Network Is Reliable

ACM Queue 2014

Bailiss & Kingsbury

@pbailiss @aphyr



Drift Into Failure

Sydney Dekker

The Pragmatic Programmers

Release It!
Second Edition

Design and Deploy
Production-Ready Software

Michael T. Nygard
Edited by Katharine Dvorak

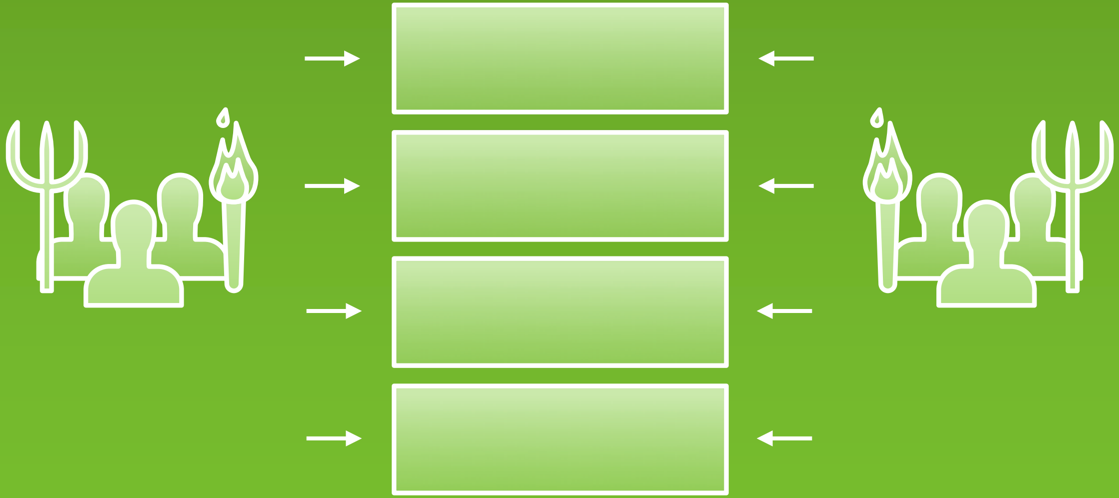
Release It!

Second Edition 2017

Michael Nygard

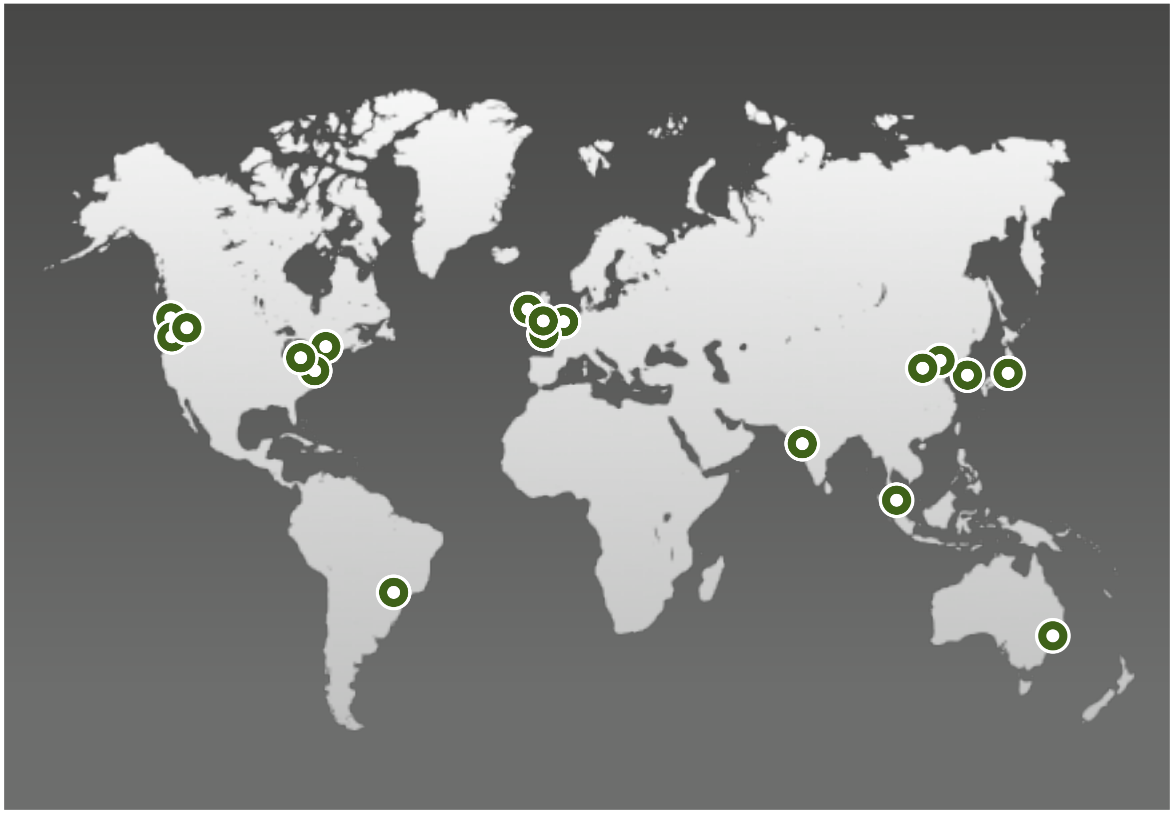
Chaos Architecture

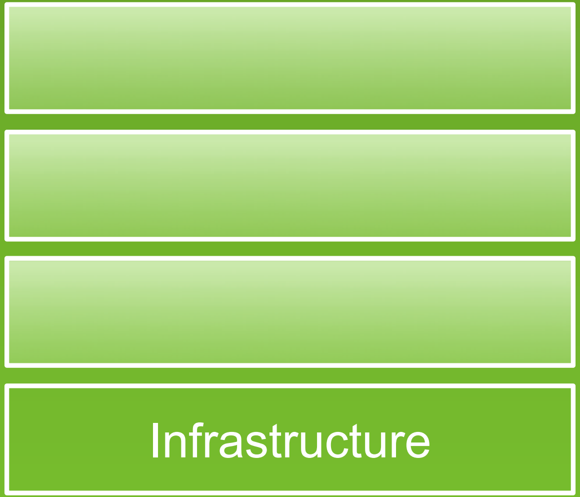
Four layers
Two teams
An attitude



Infrastructure and Services

No single point
of failure

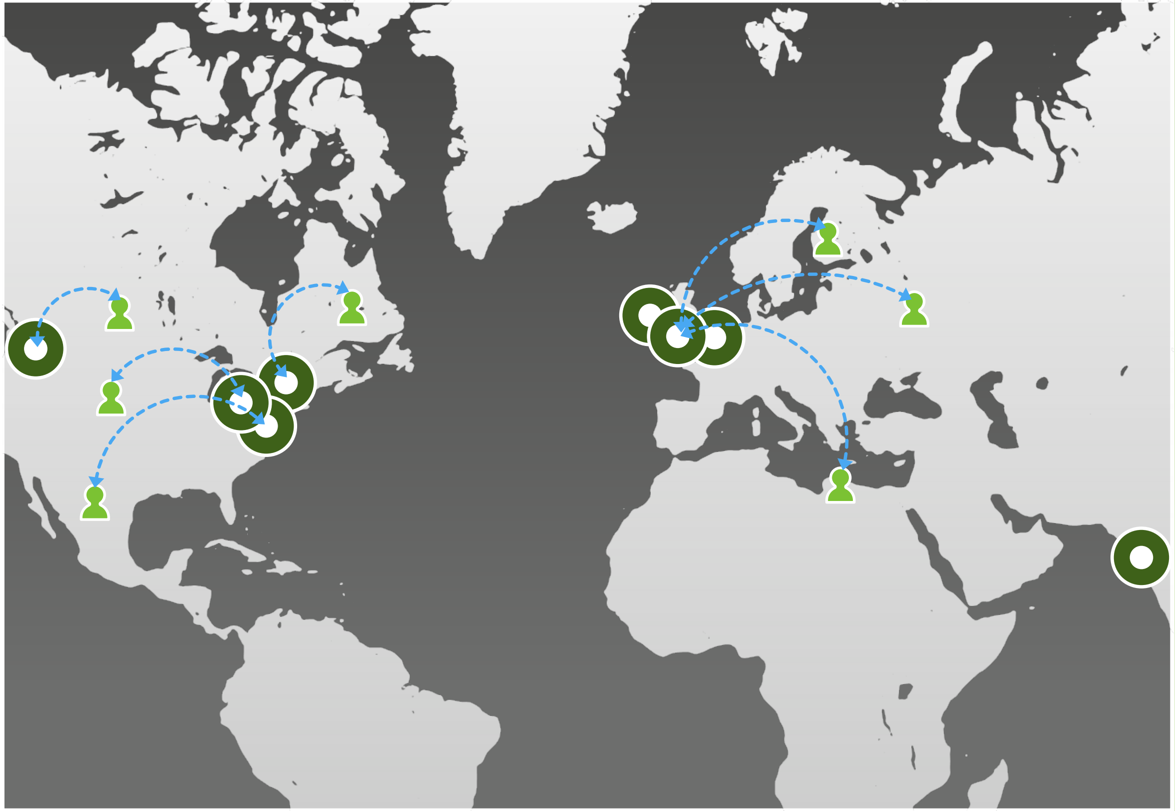




Infrastructure

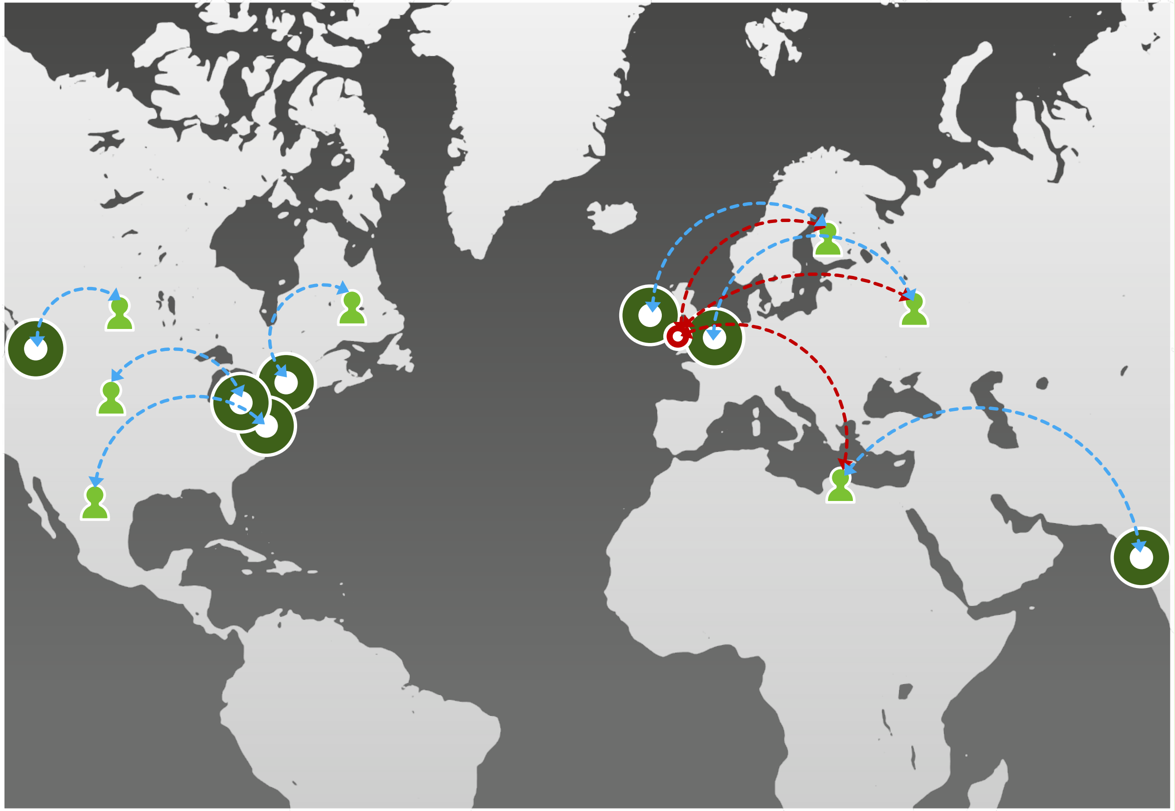
Switching and Interconnecting

Data replication
Traffic routing
Avoiding issues
Anti-entropy recovery



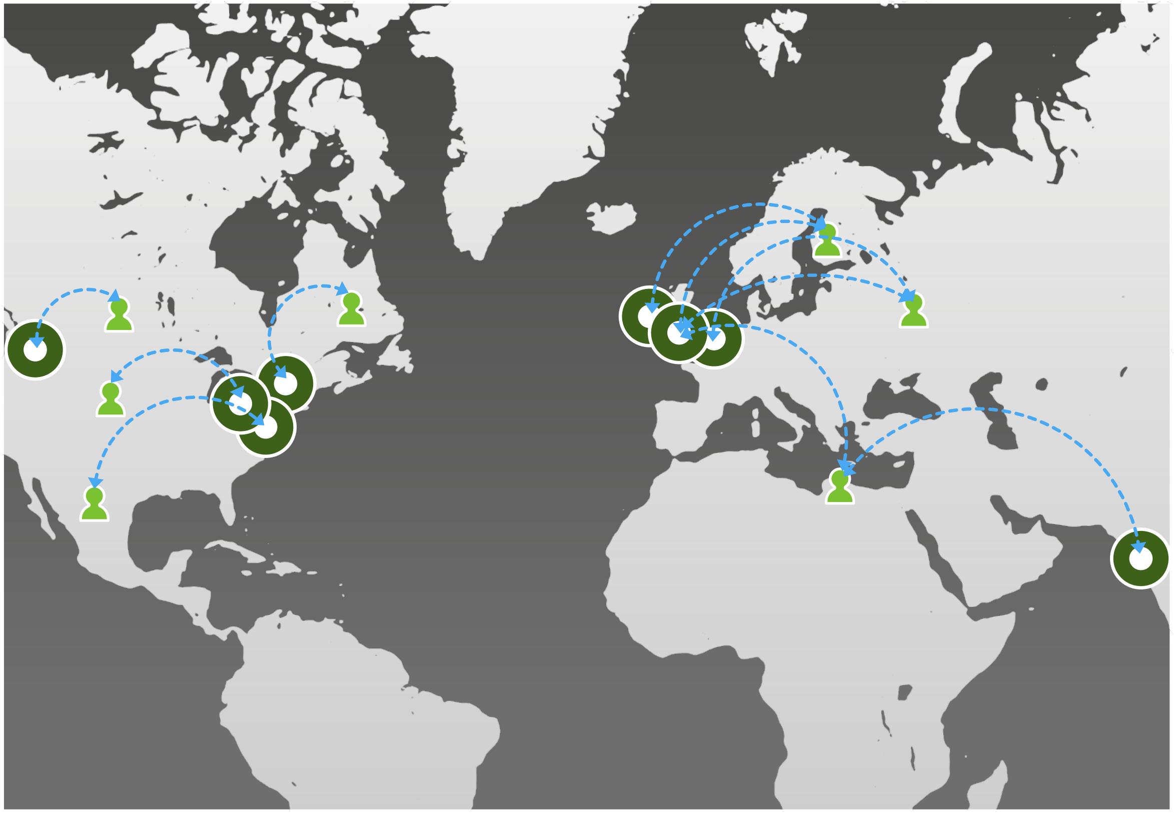
Switching and Interconnecting

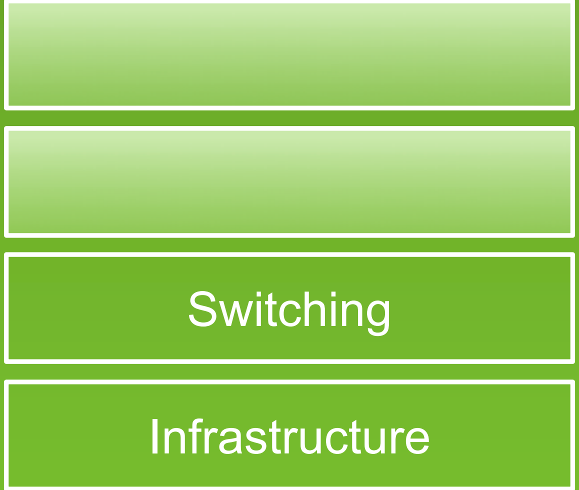
Data replication
Traffic routing
Avoiding issues
Anti-entropy recovery



Switching and Interconnecting

Data replication
Traffic routing
Avoiding issues
Anti-entropy recovery





Switching

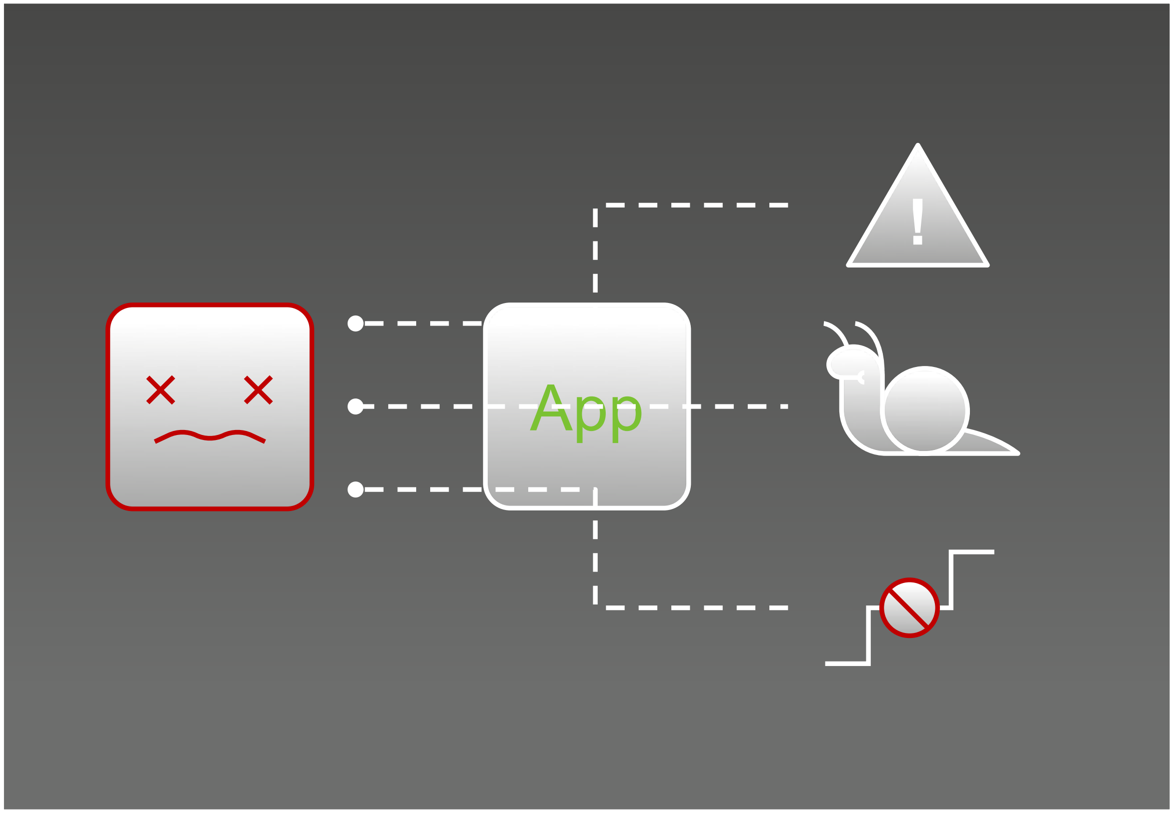
Infrastructure

Application Failures

Error returns

Slow response

Network partition



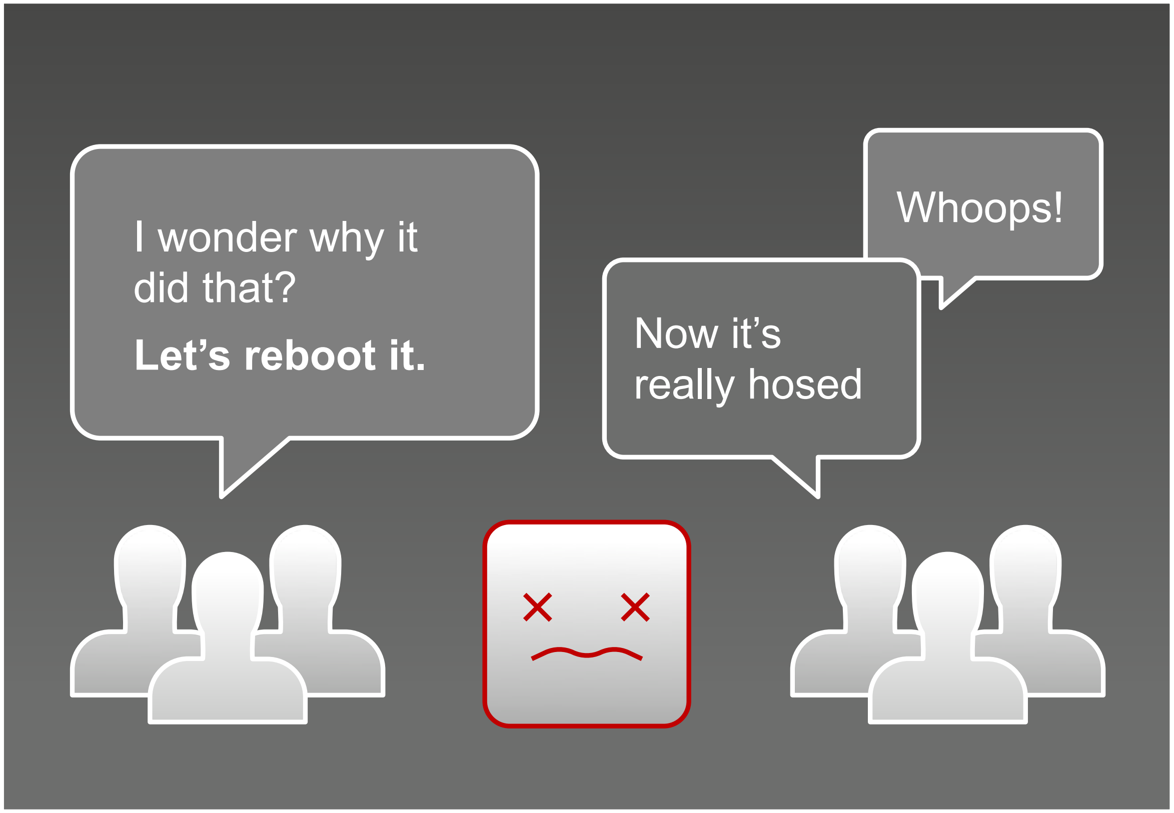
Application

Switching

Infrastructure

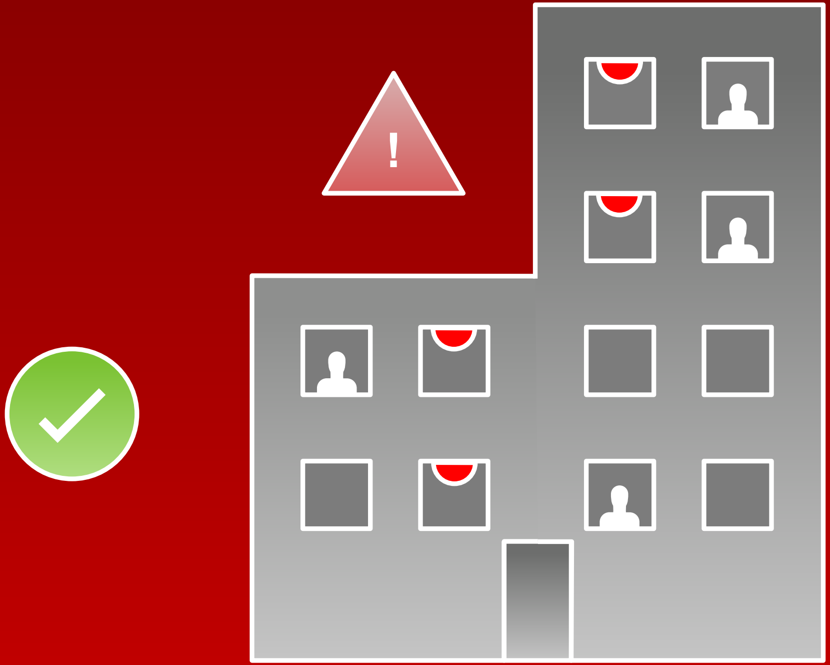
People

Unexpected application behavior often causes people to intervene and make the situation worse



People Training

A fire drill is a boring routine where we make everyone take the stairs and assemble in the parking lot



People Training

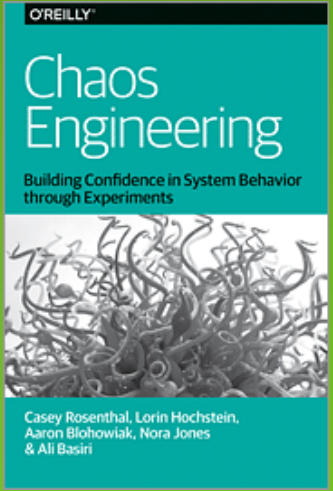
Fire drills save lives in the event of a real fire, because people are trained how to react

Who runs the “fire drill” for I.T.?

- People
- Application
- Switching
- Infrastructure

Chaos
Engineering
Team

- People
- Application
- Switching
- Infrastructure



O'REILLY

Chaos Engineering

Building Confidence in System Behavior through Experiments

Casey Rosenthal, Lorin Hochstein,
Aaron Blohowiak, Nora Jones
& Ali Basiri

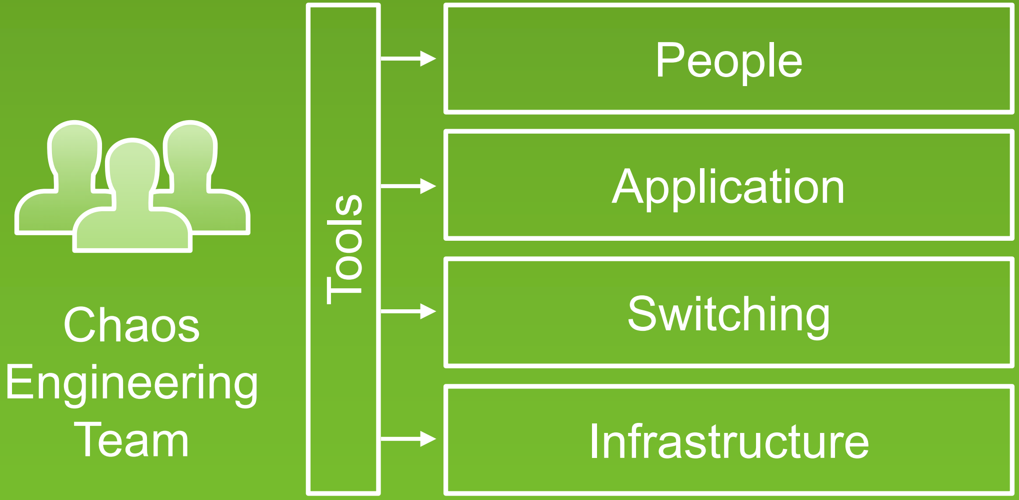
Chaos
Engineering
Team

People

Application

Switching

Infrastructure



Chaos Engineering Team

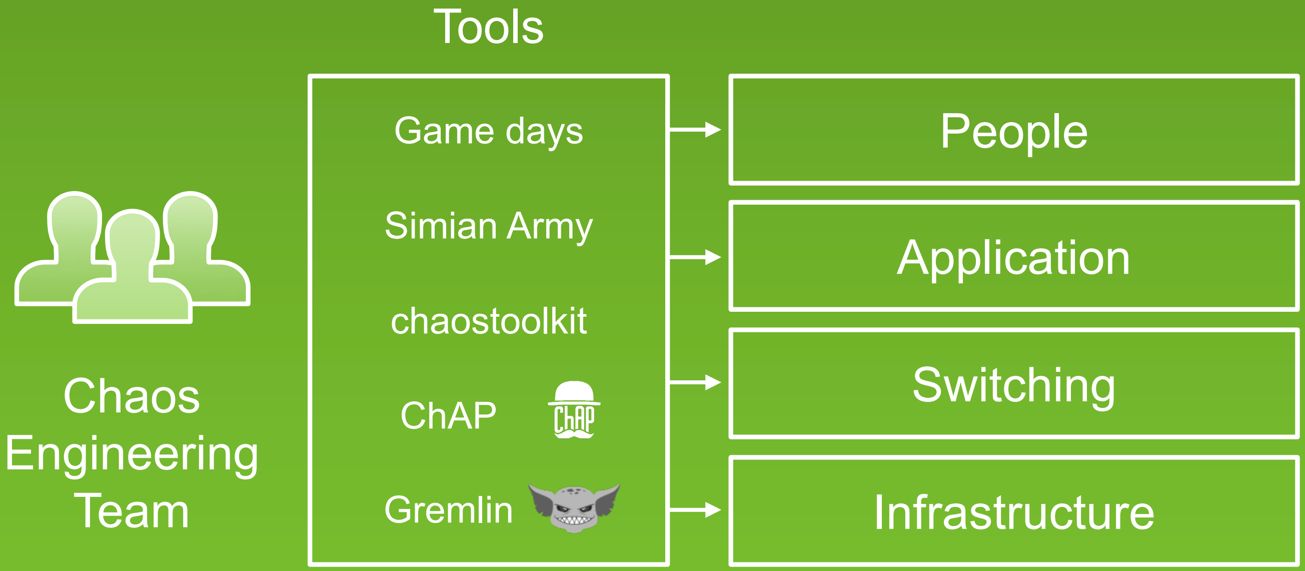
Tools

People

Application

Switching

Infrastructure

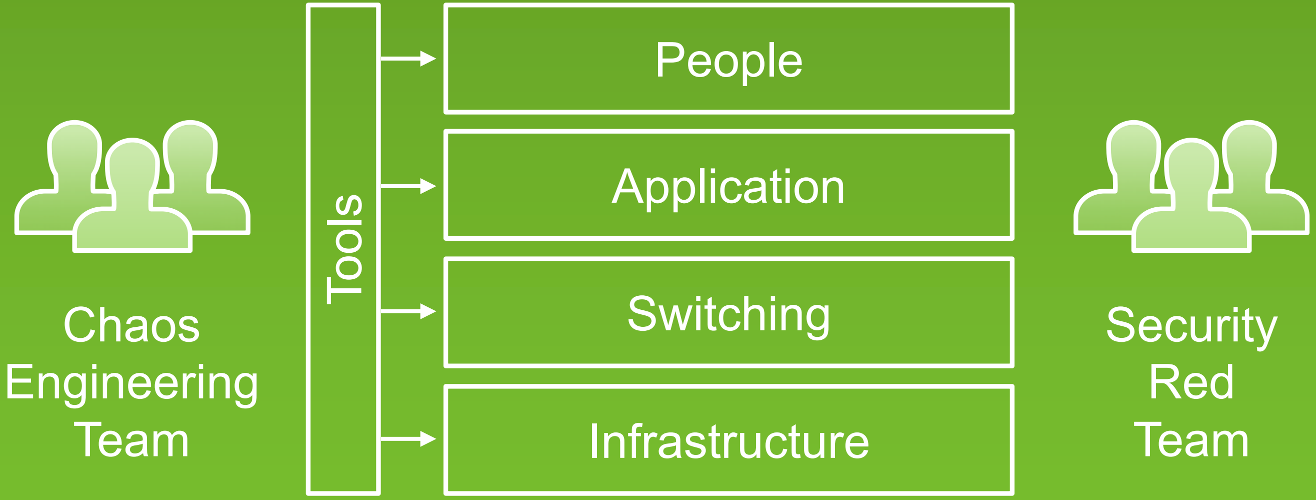


Chaos Engineering Team

Tools

- Game days
- Simian Army
- chaostoolkit
- ChAP
- Gremlin

- People
- Application
- Switching
- Infrastructure



Chaos
Engineering
Team

Tools

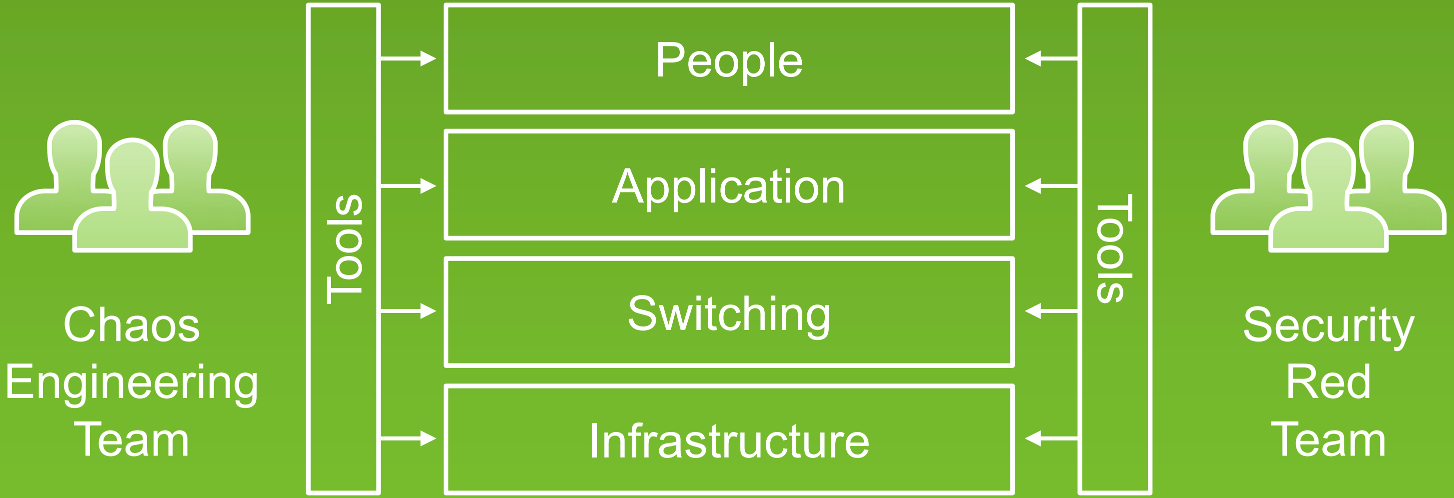
People

Application

Switching

Infrastructure

Security
Red
Team



Chaos Engineering Team

Tools

People

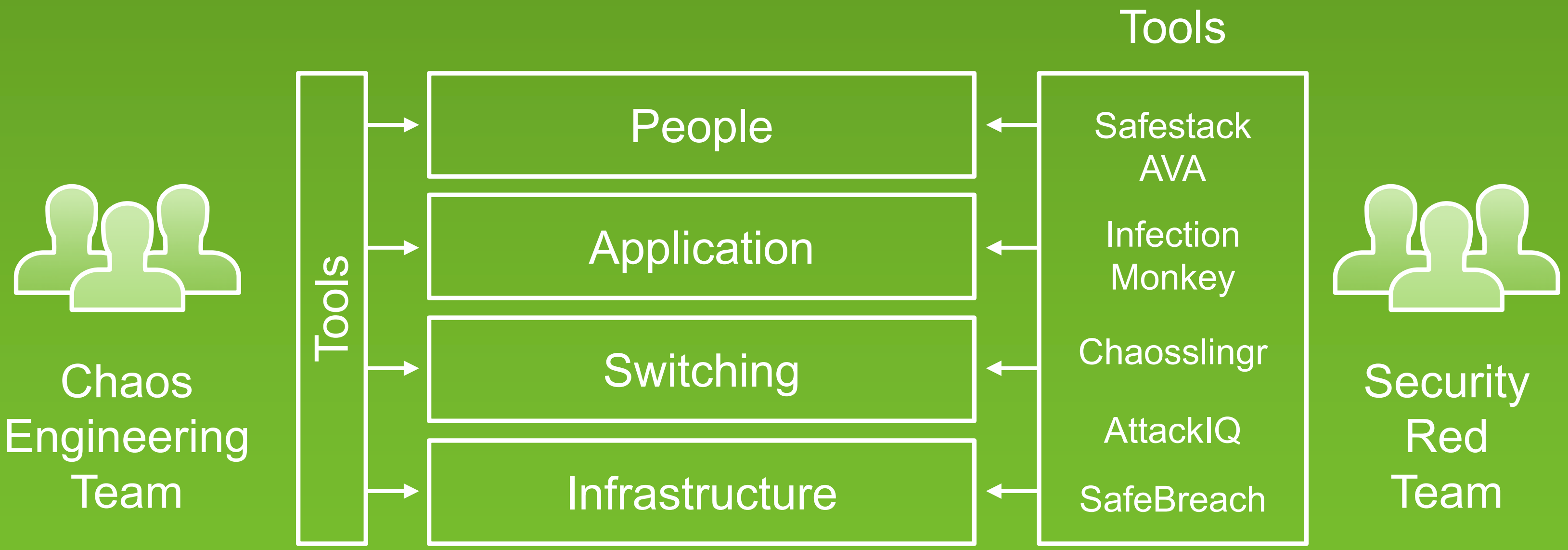
Application

Switching

Infrastructure

Tools

Security Red Team



Chaos Engineering Team

Tools

People

Application

Switching

Infrastructure

Tools

Safestack AVA

Infection Monkey

Chaosslingr

AttackIQ

SafeBreach

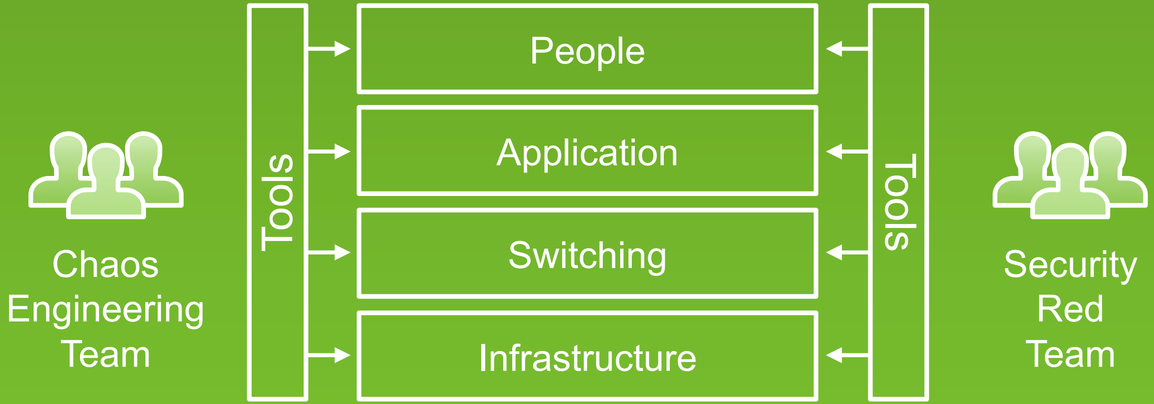
Security Red Team

Chaos Architecture

Four layers

Two teams

An attitude—
Break it to make it better



Chaos Engineering Team

Tools

People

Application

Switching

Infrastructure

Tools

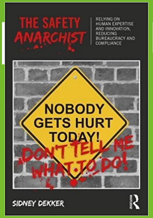
Security Red Team

Break it to make it safer

For more on the "New View" of Safety see:
Todd Conklin's Pre-accident podcast
John Allspaw's stella.report

Synoptic Illegibility

You can’t write down exactly what **really** happens, so you can’t write a synopsis or run-book. System safety is an emergent p rty


THE SAFETY ANARCHIST
RELYING ON HUMAN EXPERTISE AND INNOVATION, REDUCING BUREAUCRACY AND COMPLIANCE
NOBODY GETS HURT TODAY!
DON'T TELL ME WHAT TO DO!
SIDNEY DEKKER

The Safety Anarchist

Sydney Decker

Failures are a system problem—lack of safety margin

Not something with a root cause of component or human error

Blindfolded on a cliff edge, what would you do?

Hypothesis testing

- We think we have safety margin in this dimension, let’s carefully test to be sure
- In production
- Without causing an issue

Chaos testing ensures that you have:

Experienced Staff

Robust Applications

Dependable Switching Fabric

Redundant Service Foundation

Expensive and custom disaster recovery is being replaced by low cost, portable, automated chaos engineering.

Chaos Architecture

A Cloud Native Availability Model

Adrian Cockcroft

@adrianco

AWS VP Cloud Architecture Strategy