Security is Dead.
Long Live Rugged DevOps:
IT at Ludicrous Speed…

Joshua Corman & Gene Kim
AppSecDC
April 4, 2012

Rugged? DevOps Cookbook

About Joshua Corman

- Director of Security Intelligence for Akamai Technologies
  - Former Research Director, Enterprise Security [The 451 Group]
  - Former Principal Security Strategist [IBM ISS]

- Industry:
  - Expert Faculty: The Institute for Applied Network Security (IANS)
  - 2009 NetworkWorld Top 10 Tech People to Know
  - Co-Founder of "Rugged Software" www.ruggedsoftware.org
  - BLOG: www.cognitivedissidents.com

- Things I've been researching:
  - Compliance vs Security
  - Disruptive Security for Disruptive Innovations
  - Chaotic Actors
  - Espionage
  - Security Metrics

About Gene Kim

- Researcher, Author
- Industry:
  - Invented and founded Tripwire, CTO (1997-2010)
  - Co-author: “Visible Ops Handbook”(2006), “Visible Ops Security” (2008)
  - Co-author: “When IT Fails: The Novel,” “The DevOps Cookbook” (Coming May 2012)
- Things I’ve been researching:
  - Benchmarked 1300+ IT organizations to test effectiveness of IT controls vs. IT performance
  - DevOps, Rugged DevOps
  - Scoping PCI Cardholder Data Environment

Rugged? DevOps Cookbook

Consequences: Value & Replaceability

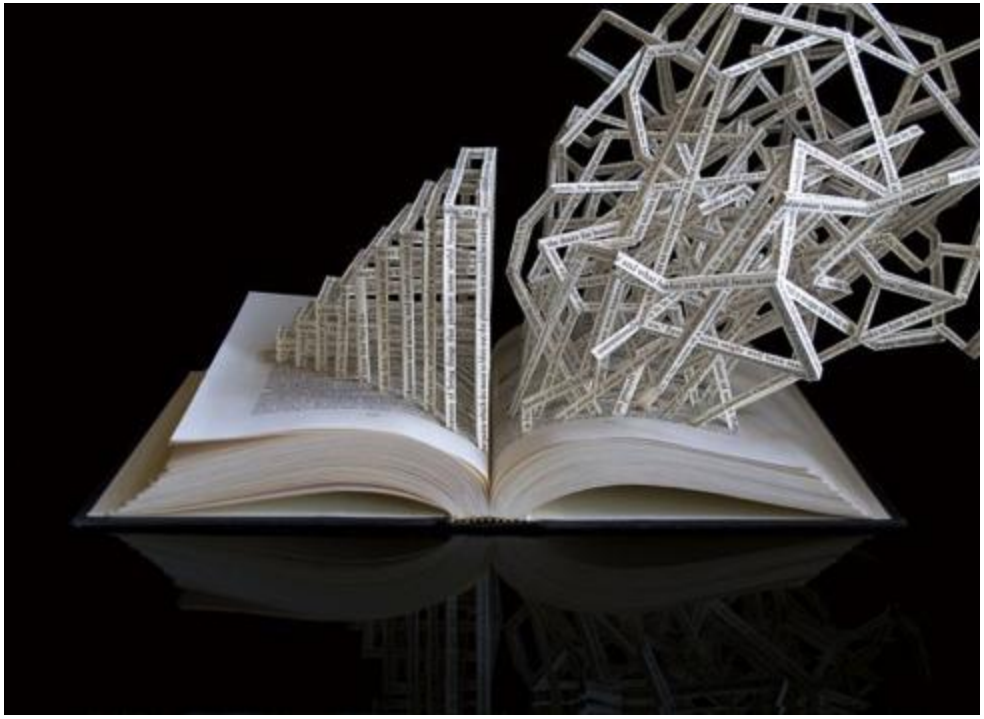
http://blog.cognitivedissidents.com/2011/10/24/a-replaceability-continuum/

Dogma: You Don't Need To Be Faster Than the Bear…

How will we rise?

http://www.vanityfair.com/culture/2012/05/internet-regulation-war-sopa-pipa-defcon-hacking#pluck-comments

Vanity Fair: World War 3.0
The battle for the Net b/w Chaos & Control



http://www.vanityfair.com/culture/2012/05/internet-regulation-war-sopa-pipa-defcon-hacking

Rugged? DevOps COOKBOOK

The Downward Spiral

Operations Sees…

- Fragile applications are prone to failure
- Long time required to figure out “which bit got flipped”
- Detective control is a salesperson
- Too much time required to restore service
- Too much firefighting and unplanned work
- Urgent security rework and remediation
- Planned project work cannot complete
- Frustrated customers leave
- Market share goes down
- Business misses Wall Street commitments
- Business makes even larger promises to Wall Street

Dev Sees…

- More urgent, date-driven projects put into the queue
- Even more fragile code (less secure) put into production
- More releases have increasingly “turbulent installs”
- Release cycles lengthen to amortize “cost of deployments”
- Failing bigger deployments more difficult to diagnose
- Most senior and constrained IT ops resources have less time to fix underlying process problems
- Ever increasing backlog of work that cold help the business win
- Ever increasing amount of tension between IT Ops, Development, Design…

These aren’t IT or Infosec problems…
These are business problems!

Good News: It Can Be Done

Bad News: You Can’t Do It Alone

Ops

QA And Test

Source: Rick Vandyll

Development

Infosec

Rugged? DevOps Cookbook

Product Management And Design

Source: Flickr: birdsandanchors

Agenda

- Problem statement
- What is DevOps?
- What is Rugged?
- What is Rugged DevOps?
- Things you can do right away

Potentially Unfamiliar Words You Will See

- Kanban
- Andon cord
- Sprints
- Rugged
- DevOps
- Bottleneck
- Systems thinking
- Controls reliance

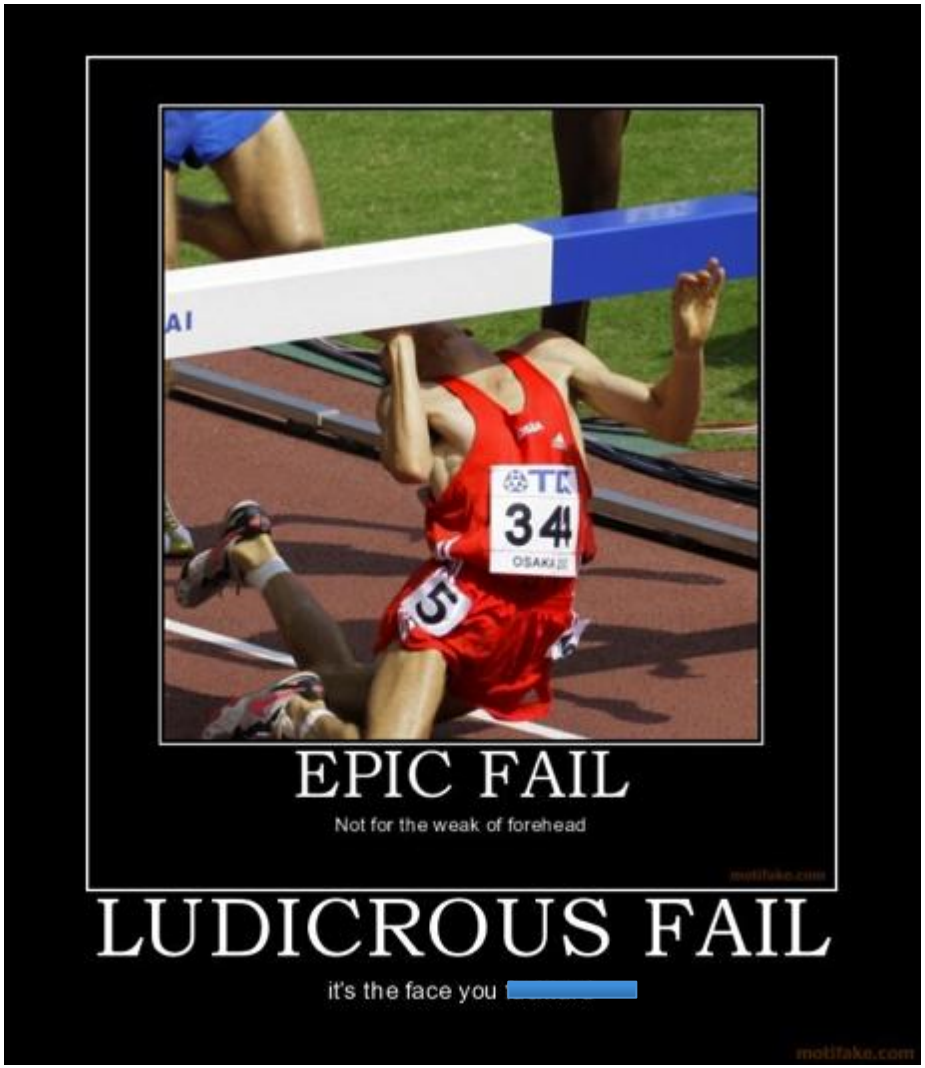
Problem Statement

Ludicrous Speed?

Ludicrous Speed

Ludicrous Speed!

Ludicrous Fail?!



What Is DevOps?

Source: John Allspaw

Little bit weird
Sits closer to the boss
Thinks too hard

Pulls levers & turns knobs
Easily excited
Yells a lot in emergencies

Source: John Allspaw

Rugged? DevOps Cookbook

Ops who think like devs
Devs who think like ops

Source: John Allspaw

Rugged? DevOps Cookbook

Source: John Allspaw

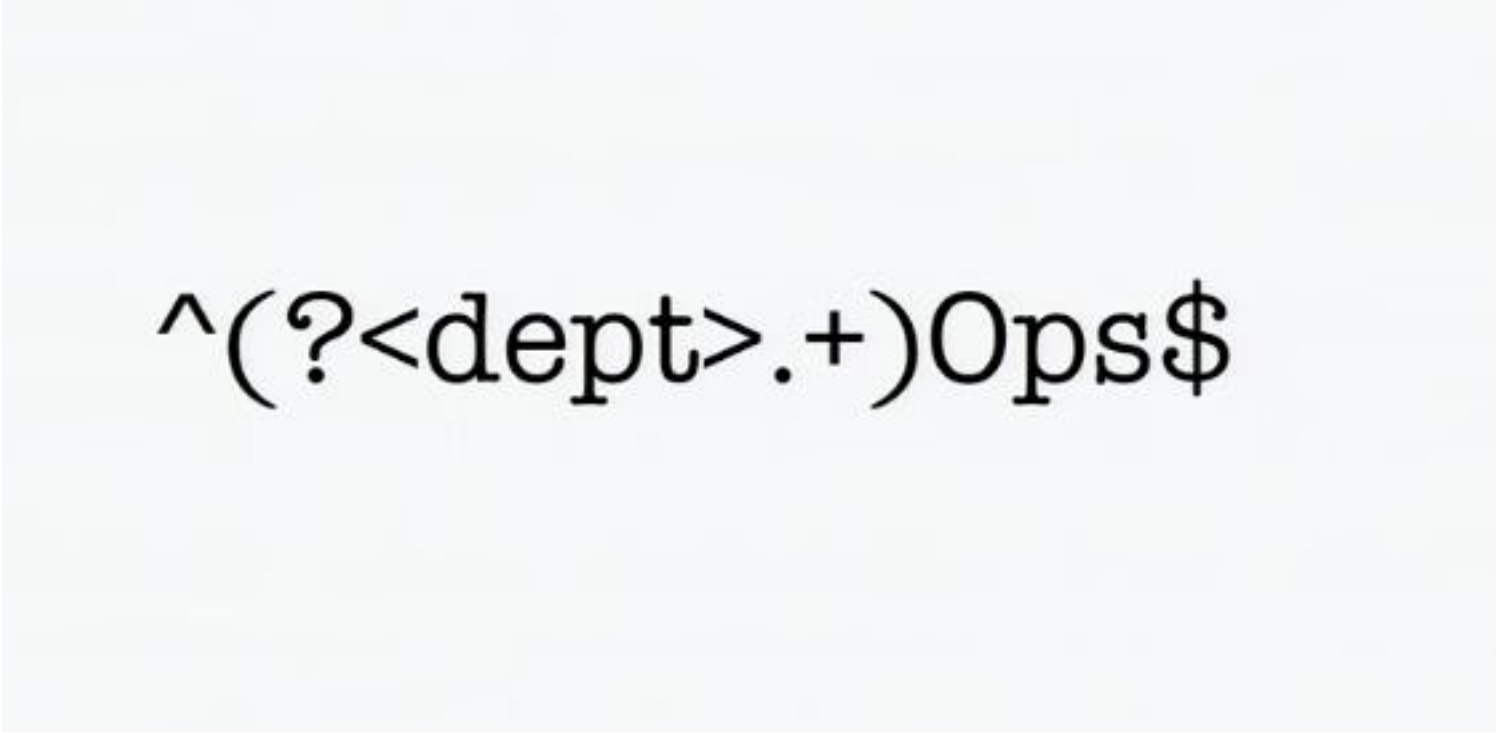
DevOps

is incomplete,
is interpreted wrong,
and is too isolated

Source: Theo Schlossnagle

Rugged? DevOps Cookbook

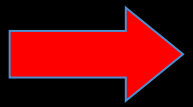
Source: Theo Schlossnagle



Source: Theo Schlossnagle

Amazon May Deployment Stats

(production hosts & environments only)



11.6 seconds

Mean time between deployments (weekday)

1,079

Max # of deployments in a single hour

10,000

Mean # of hosts simultaneously receiving a deployment

30,000

Max # of hosts simultaneously receiving a deployment

What Is Rugged?

Rugged Software Development

Joshua Corman, David Rice, Jeff Williams

2010

Rugged? DevOps Cookbook

RSA CONFERENCE | Where The World Talks Security

USA 2009 20-24 April | Moscone Center | San Francisco

GREYLOCK PARTNERS

Rugged? DevOps COOKBOOK

Rugged? DevOps COOKBOOK

RUGGED SOFTWARE

Rugged? DevOps COOKBOOK

…so software not only needs to be…

FAST

Rugged? DevOps COOKBOOK

AGILE

Are You Rugged?

Rugged? DevOps Cookbook

Rugged? DevOps Cookbook

THE MANIFESTO

Rugged? DevOps Cookbook

The Rugged Manifesto

I am rugged... and more importantly, my code is rugged.

I recognize that software has become a foundation of our modern world.

I recognize the awesome responsibility that comes with this foundational role.

I recognize that my code will be used in ways I cannot anticipate, in ways it was not designed, and for longer than it was ever intended.

I recognize that my code will be attacked by talented and persistent adversaries who threaten our physical, economic, and national security.

I recognize these things - and I choose to be rugged.

I am rugged because I refuse to be a source of vulnerability or weakness.

I am rugged because I assure my code will support its mission.

I am rugged because my code can face these challenges and persist in spite of them.

I am rugged, not because it is easy, but because it is necessary... and I am up for the challenge.

Rugged? DevOps COOKBOOK

I recognize that my code will be used in ways I cannot anticipate, in ways it was not designed, and for longer than it was ever intended.

Rugged?

www.ruggedsoftware.org

CrossTalk

http://www.crosstalkonline.org/issues/marchapril-2011.html

What Is Rugged DevOps?

The Rugged Way in the Cloud–Building Reliability and Security Into Software

James Wickett
james.wickett@owasp.org

Rugged Survival Guide

- Defensible Infrastructure
- Operational Discipline
- Situational Awareness
- Countermeasures

On YouTube: "PCI Zombies"

Rugged? DevOps Cookbook

Source: James Wickett

Rugged? DevOps Cookbook

Survival Guide/Pyramid

www.ruggedsoftware.org

Defensible Infrastructure

Survival Guide/Pyramid

Gene Kim

MULTIPLE AWARD-WINNING CTO, RESEARCHER, VISIBLE OPS CO-AUTHOR, ENTREPRENEUR & FOUNDER OF TRIPWIRE

Operational Discipline

Defensible Infrastructure

Survival Guide/Pyramid

Situational Awareness

Operational Discipline

Defensible Infrastructure

Survival Guide/Pyramid

- Countermeasures
- Situational Awareness
- Operational Discipline
- Defensible Infrastructure

Zombie Proof Housing

http://all-that-is-interesting.com/post/4956385434/the-first-zombie-proof-house

Zombie Proof Housing

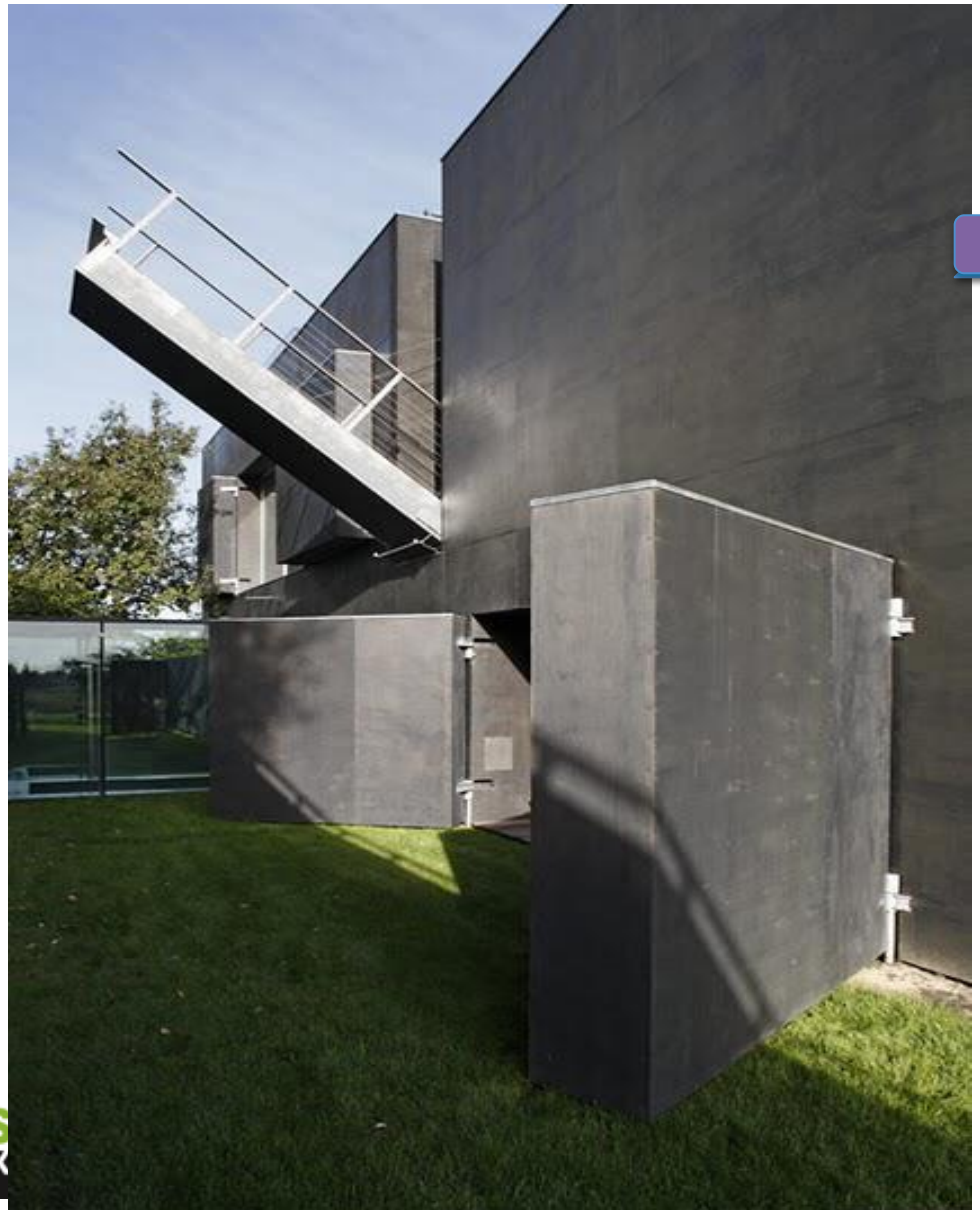
http://all-that-is-interesting.com/post/4956385434/the-first-zombie-proof-house

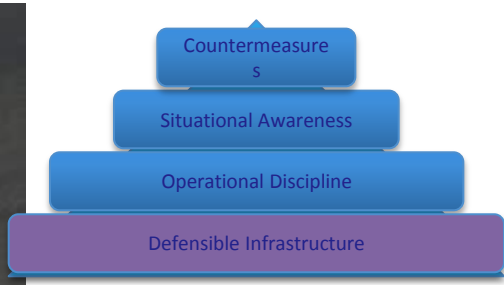
Zombie Proof Housing

http://all-that-is-interesting.com/post/4956385434/the-first-zombie-proof-house

Zombie Proof Housing

http://all-that-is-interesting.com/post/4956385434/the-first-zombie-proof-house





Rugged? DevOps Cookbook

Zombie Proof Housing

http://all-that-is-interesting.com/post/4956385434/the-first-zombie-proof-house

Zombie Proof Housing

http://all-that-is-interesting.com/post/4956385434/the-first-zombie-proof-house

Zombie Proof Housing

http://all-that-is-interesting.com/post/4956385434/the-first-zombie-proof-house

Rugged-ities

- Availability
- Survivability
- Defensibility
- Security
- Longevity
- Portability

Source: James Wickett

Home > News

Security

In Security:

News

Reviews

Features

How-tos

Slideshows

Dropbox admits it suffered serious password failure

Drops authentication for four hours

By John E Dunn | Techworld | Published: 10:26, 21 June 2011

+1 0 Like Tweet 9

Cloud file synchronisation company Dropbox has admitted that it suffered a serious security lapse that allowed an unknown number of users to log into any account using any password.

In a blog post, Dropbox said that for four hours on the afternoon of 20 June (US Pacific Time) a bug in its authentication system would have allowed some users to log in "without the correct password."

"A very small number of users (much less than 1 percent) logged in during that period [...]. As a precaution, we ended all logged in sessions," the blog said.

How Do You Do Rugged DevOps?

DEVOPS COOKBOOK

*a prescriptive guide to starting and finishing successful DevOps and *Ops-style transformations




- "DevOps Cookbook" Authors
  - Patrick DeBois, Mike Orzen, John Willis
- Goals
  - Codify how to start and finish DevOps transformations
  - How does Development, IT Operations and Infosec become dependable partners
  - Describe in detail how to replicate the transformations describe in "When IT Fails: The Novel"

The First Way:
Systems Thinking

The First Way:
Systems Thinking (Left To Right)

- Never pass defects to downstream work centers
- Never allow local optimization to create global degradation
- Increase flow: elevate bottlenecks, reduce WIP, throttle release of work, reduce batch sizes

Definition: Agile Sprints

- The basic unit of development in Agile Scrums, typically between one week and one month
- At the end of each sprint, team should have potentially deliverable product

Aha Moment: shipping product implies not just code – it's the environment, too!

Help Dev And Ops Build Code And Environments

- Dev and Ops work together in Sprint 0 and 1 to create code and environments
  - Create environment that Dev deploys into
  - Create downstream environments: QA, Staging, Production
  - Create testable migration procedures from Dev all the way to production
- Integrate Infosec and QA into daily sprint activities

The First Way:
Systems Thinking: Infosec

- Get a seat at the table
  - DevOps programs are typically led by Dev, QA, IT Operations and Product Management
- Add value at every step in the flow of work
  - See the end-to-end value flow
  - Shorten and amplify feedback loops
  - Help break silos (e.g., server, networking, database)

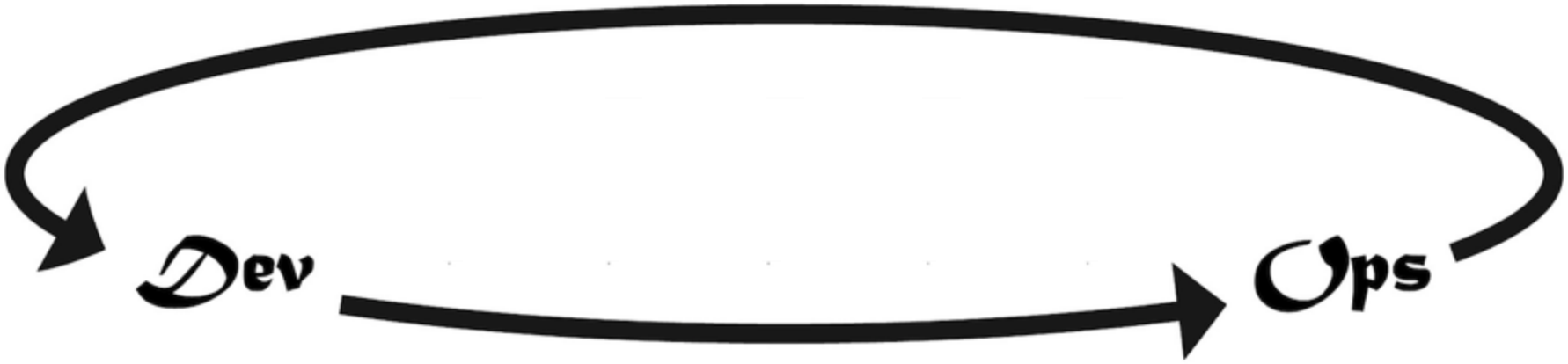
The First Way:
Systems Thinking: Infosec Insurgency

- Have infosec attend the daily Agile standups
  - Gain awareness of what the team is working on
- Find the automated infrastructure project team (e.g., puppet, chef)
  - Provide hardening guidance
  - Integrate and extend their production configuration monitoring
- Find where code packaging is performed
  - Integrate security testing pre- and post-deployment
- Integrate into continuous integration and release process
  - Add security test scripts to automated test library

The First Way: Outcomes

- Determinism in the release process
- Continuation of the Agile and CI/CR processes
- Creating single repository for code and environments
- Packaging responsibility moves to development
- Consistent Dev, QA, Int, and Staging environments, all properly built before deployment begins
- Decrease cycle time
  - Reduce deployment times from 6 hours to 45 minutes
  - Refactor deployment process that had 1300+ steps spanning 4 weeks
- Faster release cadence

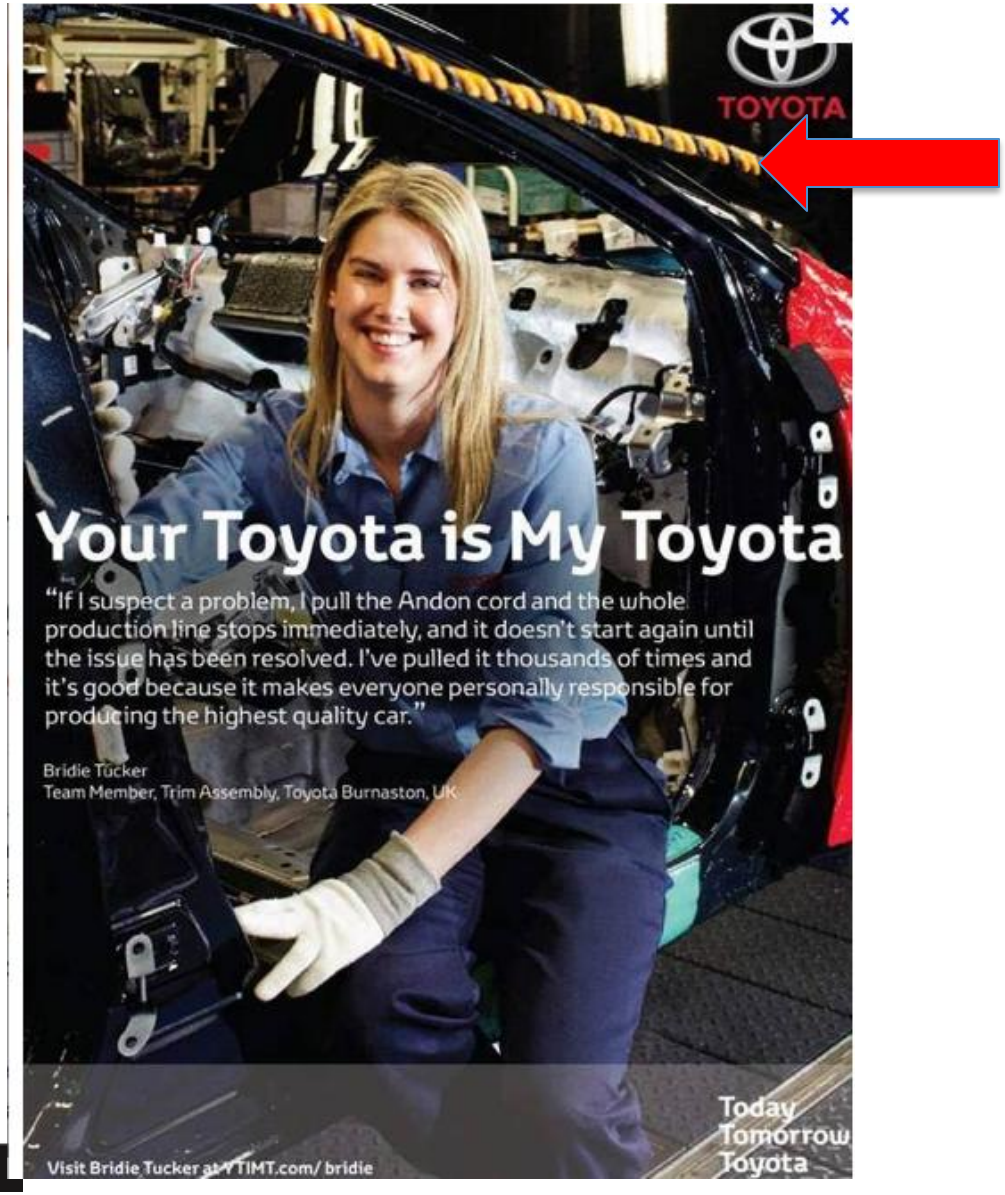
The Second Way:
Amplify Feedback Loops



The Second Way:
Amplify Feedback Loops (Right to Left)

- Protect the integrity of the entire system of work, versus completion of tasks
- Expose visual data so everyone can see how their decisions affect the entire system

Definition: Andon Cord



Rugged? DevOps COOKBOOK

Integrate Ops Into Dev

- Embed Ops person into Dev structure
  - Describes non-functional requirements, use cases and stories from Ops
  - Responsible for improving “quality at the source” (e.g., reducing technical debt, fix known problems, etc.)
  - Has special responsibility for pulling the Andon cord

Integrate Dev Into Ops

- MobBrowser case study: “Waking up developers at 3am is a great feedback loop: defects get fixed very quickly”

- Goal is to get Dev closer to the customer
  - Infosec can help determine when it’s too close (and when SOD is a requirement)

Keep Shrinking Batch Sizes

- Waterfall projects often have cycle time of one year
- Sprints have cycle time of 1 or 2 weeks
- When IT Operations work is sufficiently fast and cheap, we may decide to decouple deployments from sprint boundaries (e.g., Kanbans)

Definition: Kanban Board

- Signaling tool to reduce WIP and increase flow

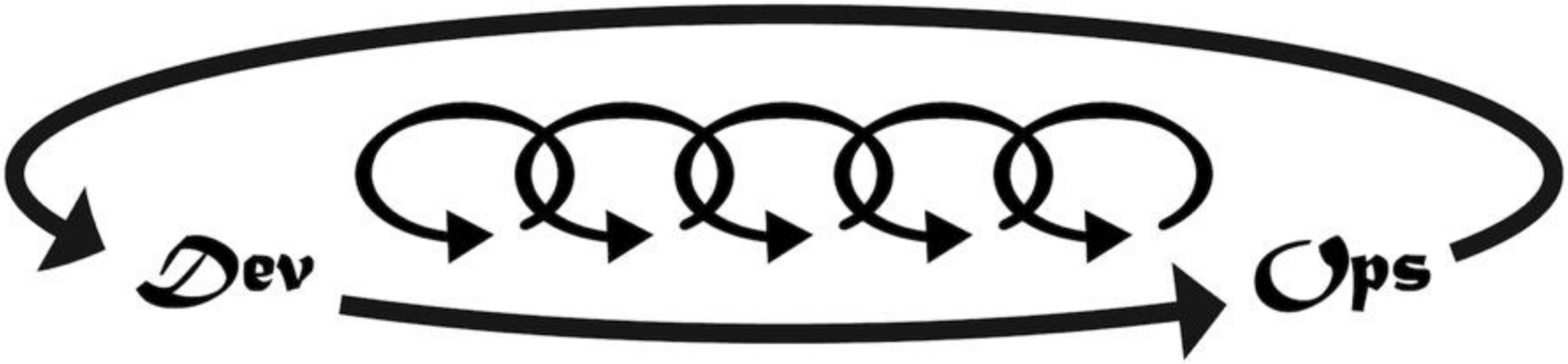
The Second Way:
Amplify Feedback Loops: Infosec Insurgency

- Extend criteria of what changes/deploys cannot be made without triggering full retest
- Create reusable Infosec use and abuse stories that can be added to every project
  - “Handle peak traffic of 4MM users and constant 4-6 Gb/sec Anonymous DDoS attacks”
- Integrate Infosec and IR into the Ops/Dev escalation processes (e.g., RACI)
- Pre-enable, shield / streamline successful audits
  - Document separation of duty and compensating controls
  - Don’t let them disrupt the work

The Second Way: Outcomes

- Andon cords that stop the production line
- Kanban to control work
- Project freeze to reduce work in process
- Eradicating “quick fixes” that circumvent the process
- Ops user stories are part of the Agile planning process
- Better build and deployment systems
- More stable environment
- Happier and more productive staff

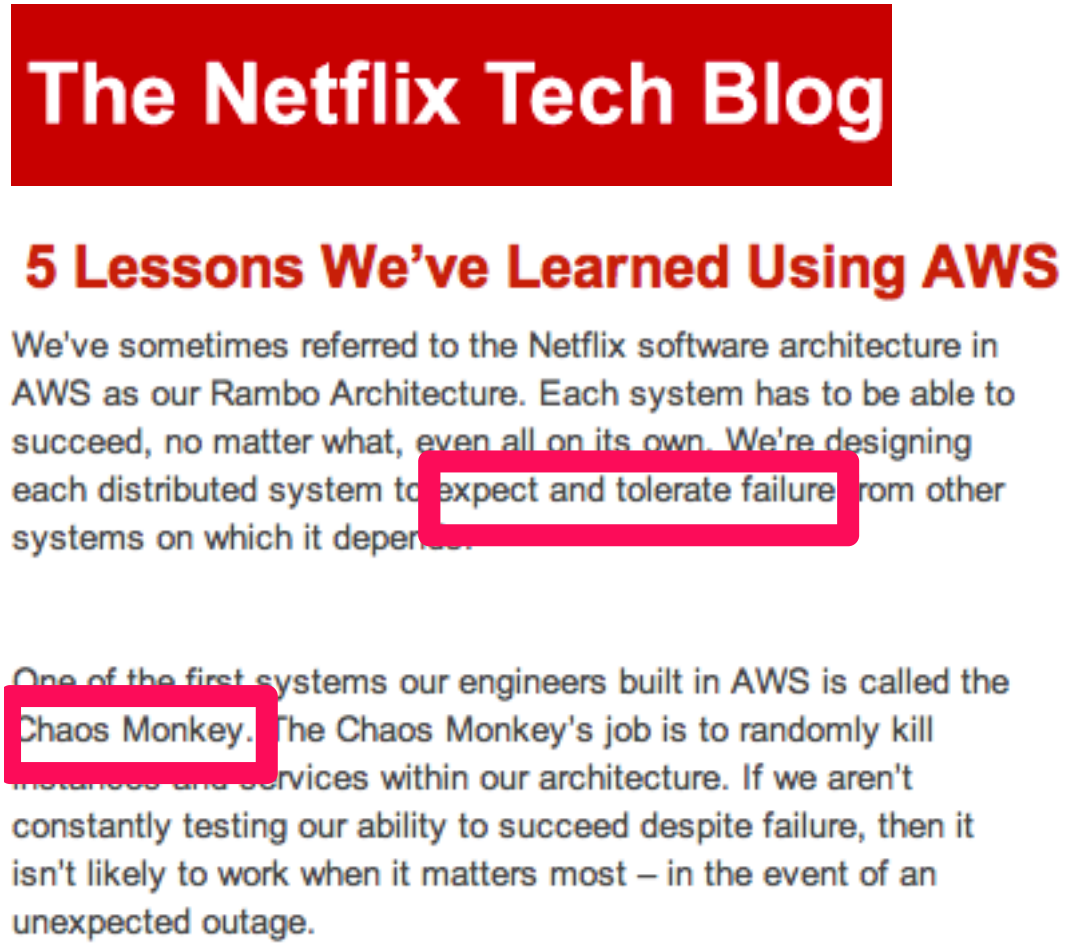
The Third Way:
Culture Of Continual Experimentation And Learning



The Third Way: Culture Of Continual Experimentation And Learning

- Foster a culture that rewards:
  - Experimentation (taking risks) and learning from failure
  - Repetition is the prerequisite to mastery
- Why?
  - You need a culture that keeps pushing into the danger zone
  - And have the habits that enable you to survive in the danger zone

Help IT Operations…


The Netflix Tech Blog

5 Lessons We've Learned Using AWS

We've sometimes referred to the Netflix software architecture in AWS as our Rambo Architecture. Each system has to be able to succeed, no matter what, even all on its own. We're designing each distributed system to expect and tolerate failure from other systems on which it depends.

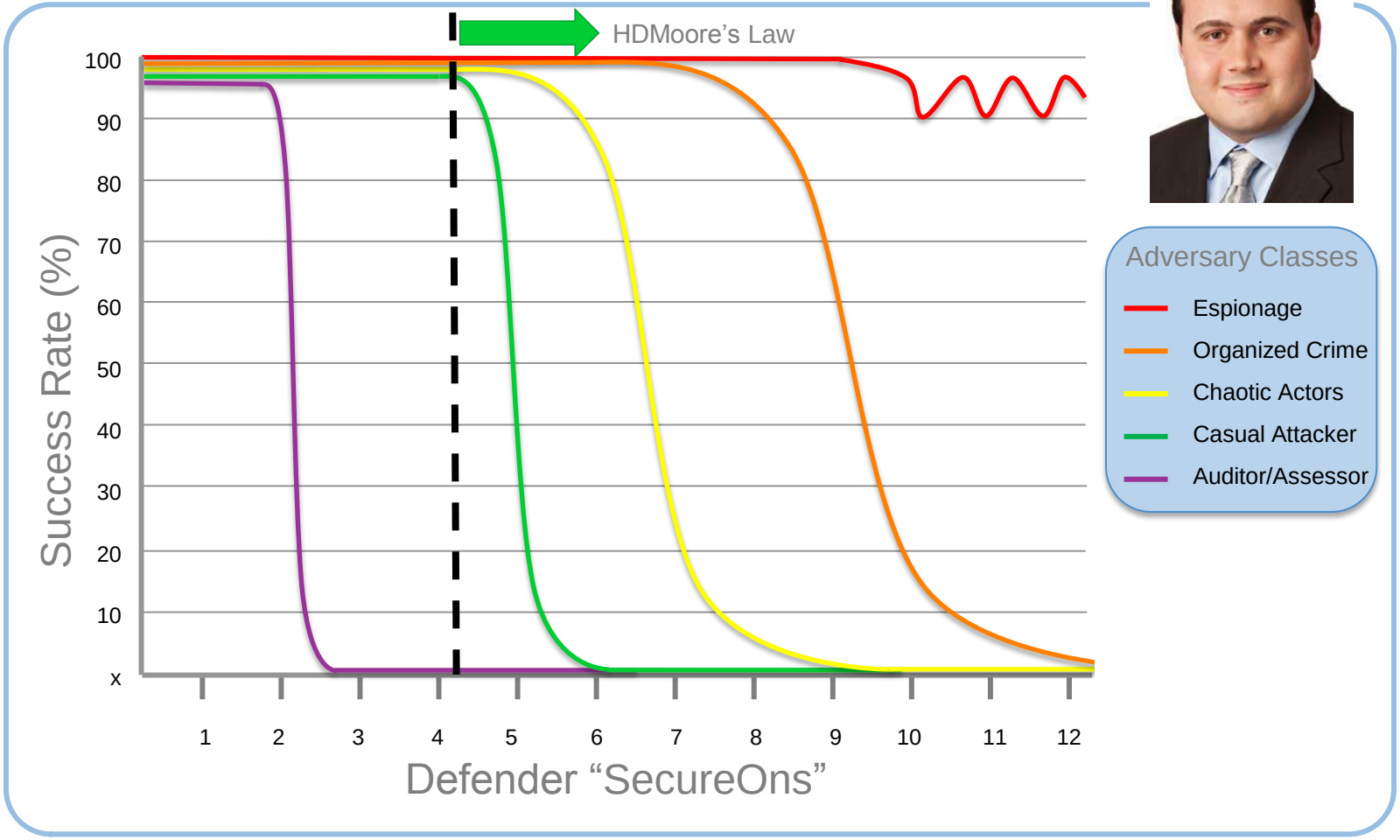
One of the first systems our engineers built in AWS is called the Chaos Monkey. The Chaos Monkey's job is to randomly kill instances and services within our architecture. If we aren't constantly testing our ability to succeed despite failure, then it isn't likely to work when it matters most – in the event of an unexpected outage.

- "The best way to avoid failure is to fail constantly"
- Harden the production environment
- Have scheduled drills to "crash the data center"
- Create your "chaos monkeys" to introduce faults into the system (e.g., randomly kill processes, take out servers, etc.)
- Rehearse and improve responding to unplanned work
  - NetFlix: Hardened AWS service
  - StackOverflow
  - Amazon firedrills (Jesse Allspaw)
  - The Monkey (Mac)

Rugged! DEVOPS COOKBOOK

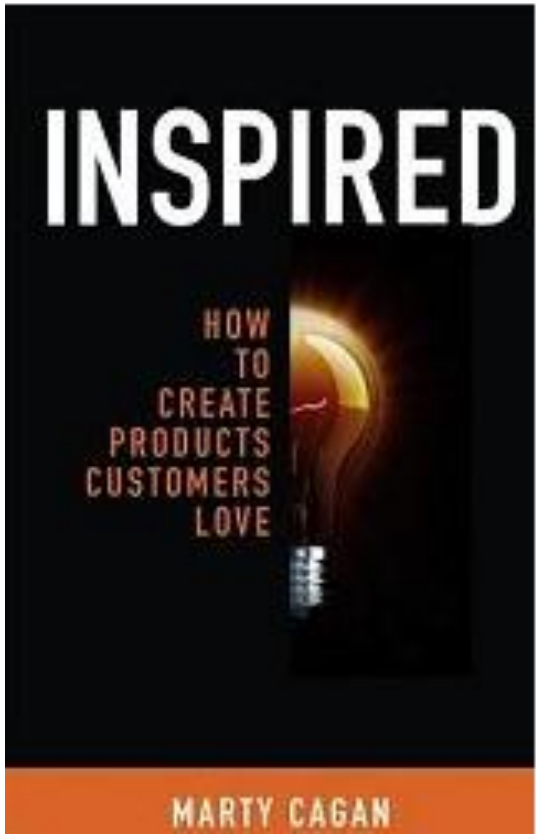
You Don’t Choose Chaos Monkey…
Chaos Monkey Chooses You

HDMoore's Law



http://blog.cognitivedissidents.com/2011/11/01/intro-to-hdmoores-law/

Help Product Management…



CNET › News › E-Bus

August 6, 1999 3:50 PM PDT

eBay online again after 14-hour outage

By Tim Clark
Staff Writer, CNET News

Lesson: Allocate 20% of Dev cycles to paying down technical debt

Rug

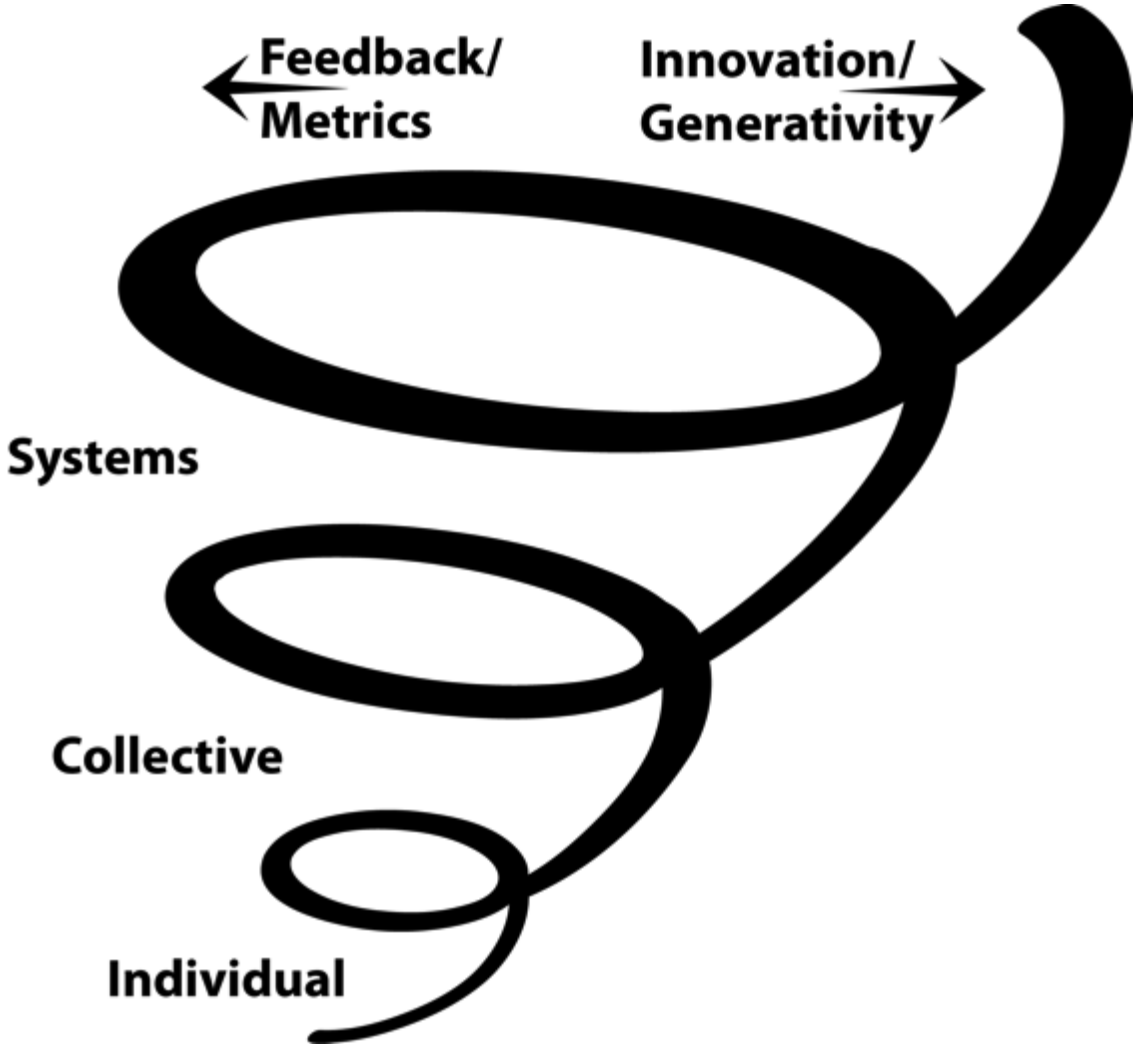
The Third Way: Culture Of Continual Experimentation And Learning: Infosec

- Add Infosec fixes to the Agile backlog
  - Make technical debt visible
  - Help prioritize work against features and other non-functional requirements
- Weaponize the Security Monkey
  - Evil/Fuzzy/Chaotic Monkey
  - Eridicate SQLi and XSS defects in our lifetime
- Let loose the Security Monkies and the Simian Army
- Eliminate needless complexity
- Become the standard bearer: 20% of Dev cycles spent on non-functional requirements
- Take work out of the system
- Keep decreasing cycle time: it increases work that the system can achieve

The Third Way: Outcomes

- 15 minutes/daily spent on improving daily work
- Continual reduction of unplanned work
- More cycles for planned work
- Projects completed to pay down technical debt and increase flow
- Elimination of needless complexity
- More resilient code and environments
- Balancing nimbleness and practiced repetition
- Enabling wider range of risk/reward balance

The Upward Spiral



What Does Rugged DevOps Feel Like?

Rugged? DevOps COOKBOOK


HONEY BADGER
DON'T CARE


HONEY BADGER
CARES

Case Studies And Early Indicators

- Almost every major Internet online services company
- VERACODE Rapid SaaS Fix Blog on Lithium
  - http://www.veracode.com/blog/2012/01/vulnerability-response-done-right/
- Pervasive Monitoring
  - Analytics at LinkedIn viewed by CEO daily: LinkedIn Engineering: “The Birth Of inGraphs: Eric The Intern”

Applying RuggedDevOps

Things To Put Into Practice Tomorrow

- Identify your Dev/Ops/QA/PM counterparts
- Discuss your mutual interdependence and shared objectives
- Harden and instrument the production builds
- Integrate automated security testing into the build and deploy mechanisms
- Create your Evil/Hostile/Fuzzy Chaos Monkey
- Cover your untested branches
- Enforce the 20% allocation of Dev cycles to non-functional requirement

When IT Fails: The Novel and *The DevOps Cookbook*

Gene Kim, Tripwire founder,
Visible Ops co-author

- Coming in July 2012
- "In the tradition of the best MBA case studies, this book should be mandatory reading for business and IT graduates alike." -**Paul Muller, VP Software Marketing, Hewlett-Packard**
- "The greatest IT management book of our generation." **–Branden Williams, CTO Marketing, RSA**

When IT Fails: The Novel and *The DevOps Cookbook*

Gene Kim, Tripwire founder,
Visible Ops co-author

- Coming in July 2012

- If you would like the “Top 10 Things You Need To Know About DevOps,” sample chapters and updates on the book:

  Sign up at http://itrevolution.com
  Email genek@realgenekim.me
  Give me your business card

To Join The Movement

- If you would like the “Top 10 Things You Need To Know About DevOps,” sample chapters and updates on the book:

  Sign up at http://itrevolution.com
  Email genek@realgenekim.me
  Give me your business card

Thank You

@joshcorman
@RuggedSoftware
@RuggedDevOps

Appendix

Resources

- From the IT Process Institute www.itpi.org
  - Both Visible Ops Handbooks
  - ITPI IT Controls Performance Study
- Rugged Software by Corman, et al: http://ruggedsoftware.org
- “Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation” by Humble, Farley
- Follow us…
  - @JoshCorman, @RealGeneKim
  - mailto:genek@realgenekim.me
  - http://realgenekim.me/blog

Rugged? DevOps COOKBOOK

Common Traits of High Performers

Culture of…

Change management

- Integration of IT operations/security via problem/change management
- Processes that serve both organizational needs and business objectives
- Highest rate of effective change

Causality

- Highest service levels (MTTR, MTBF)
- Highest first fix rate (unneeded rework)

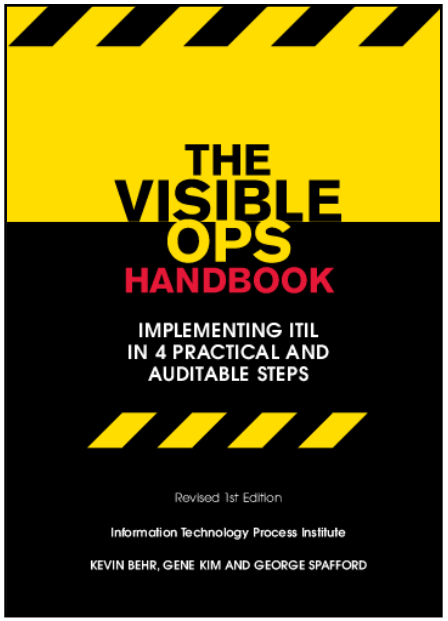
Compliance and continual reduction of operational variance

- Production configurations
- Highest level of pre-production staffing
- Effective pre-production controls
- Effective pairing of preventive and detective controls

Source: IT Process Institute

Visible Ops: Playbook of High Performers

- The IT Process Institute has been studying high-performing organizations since 1999
  - What is common to all the high performers?
  - What is different between them and average and low performers?
  - How did they become great?
- Answers have been codified in the Visible Ops Methodology
- The “Visible Ops Handbook” is available from the ITPI



www.ITPI.org

IT Operations Increases Process Rigor

- Standardize deployment
- Standardize unplanned work: make it repeatable
- Modify first response: ensure constrained resources have all data at hand to diagnose
- Elevate preventive activities to reduce incidents

Help Development…

- Help them see downstream effects
  - Unplanned work comes at the expense of planned work
  - Technical debt retards feature throughput
  - Environment matters as much as the code
- Allocate time for fault modeling, asking “what could go wrong?” and implementing countermeasures

Help QA…

- Ensure test plans cover not only code functionality, but also:
  - Suitability of the environment the code runs in
  - The end-to-end deployment process
- Help find variance…
  - Functionality, performance, configuration
  - Duration, wait time and handoff errors, rework, …