DevOps and Audit

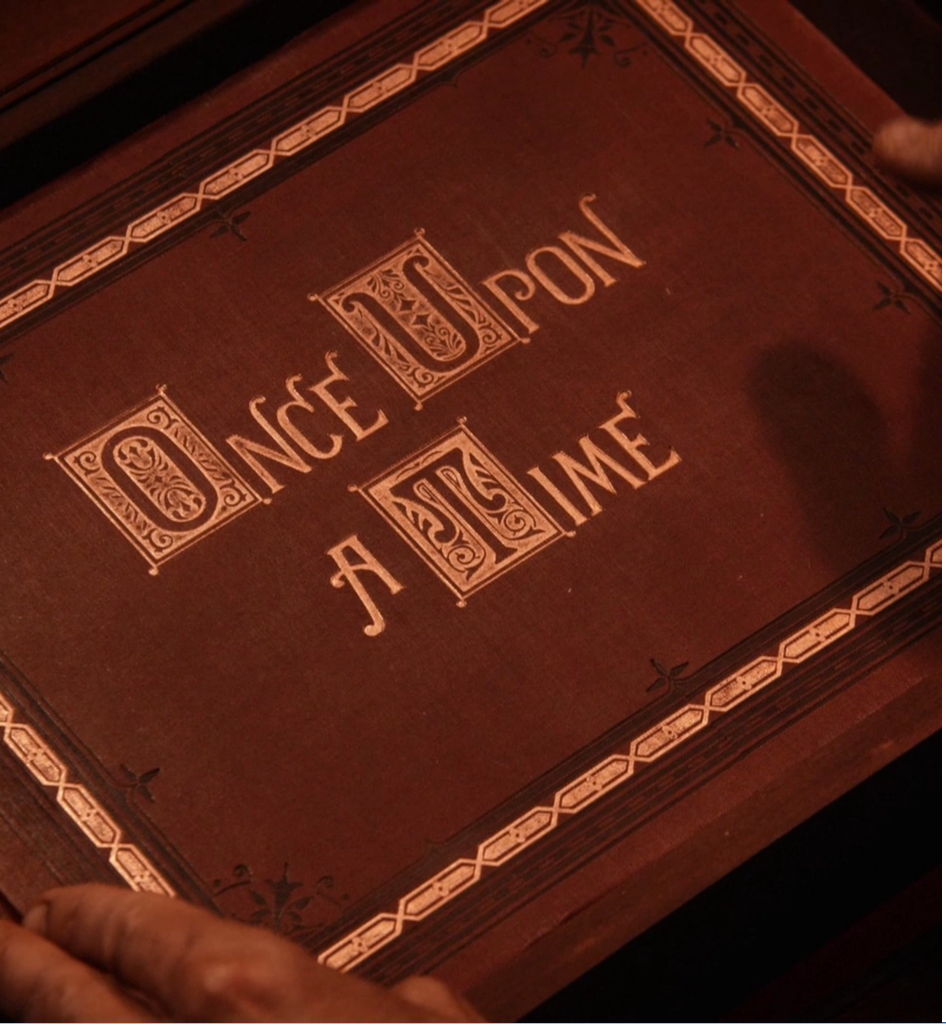
…and Security and Compliance

Jeff Gallimore

excella CONSULTING

excella.com

@excellaco



Let me tell you a story.

AnyOnlineBusiness, Inc.

Sells products and services.

Stores information.

Takes payments.

The audit begins.

Developers are deploying their own code to production without any formal change approval processes.

No separation of duties.

Business risks.

BR1: Service goes down, and revenue can’t be generated. *(availability)*

BR2: User accounts and data are compromised or disclosed. *(confidentiality, security, privacy)*

BR3: Transactions are incorrect or compromised. *(integrity)*

Specific risks.

R1: An internal actor abuses provided or developed privileges to commit fraud to the organization and/or its customers. *(BR2, BR3)*

R2: Code is deployed into production that causes an outage, service impairment, or errors in data. *(BR1)*

R3: An external actor gains unauthorized access to production or pre-production environments (e.g., database, OS, networking) and installs malicious code, or changes or steals data. *(BR2, BR3)*

Control strategy.

To mitigate the risk of an internal actor abusing provided or developed privileges to commit fraud to the organization and/or its customers *(R1)*, we use the following control strategy…

CS1: All code is validated through defined controls prior to production deployment to prevent developers from inserting “back doors” or vulnerabilities into production.

Control environment.

To implement **CS1** so all code is validated through defined controls prior to production deployment to prevent developers from inserting “back doors” or vulnerabilities into production, we have the following controls…

Controls.

C1: All changes are reviewed by the Change Control Board (CCB) prior to release.

a) The changes are submitted for review at least two weeks prior to the next CCB meeting.
b) The submitter must complete the Change Control Form (CCF), documenting the changes to be made, which environments the change should be applied to, what risks are associated with the change, and rollback procedures.
c) If the CCB approves the change, the change will be scheduled for the next release window with the IT Operations team.

Audit testing and evidence.

CS1 evidence:

a) Documentation of CCB procedures.
b) CCB meeting agendas for the last year.
c) CCFs for each CCB meeting for the last year.
d) Record of approval for each CCF.
e) Record of changes applied for each production release window, along with CCF for each of those changes.
f) Record of which changes were applied successfully and which failed.
g) For change failures, record of rollback procedures applied and outcome of the rollback.

This is a horror story.

Findings.

1. Material differences between documented CCB procedures and actual CCB meetings.
2. 86% changes applied to production were approved by CCB.
3. 17% of changes approved by CCB were not applied to production.
4. 8% of changes did not have accompanying CCF.
5. 31% of production changes failed in some manner. 5% caused significant service disruption.
6. Rollback procedures failed for 53% of failed changes.

The End.

No, wait. It’s a fairytale!

Control environment.

To implement **CS1** so all code is validated through defined controls prior to production deployment to prevent developers from inserting “back doors” or vulnerabilities into production, we have the following controls…

Controls.

C1: Jenkins runs static code analysis on each code checkin to validate the code conforms to the established lexical, syntactic, and semantic ruleset.

C2: All code is peer-reviewed for correctness and adherence to the organizational coding standards. To ensure the peer review happened and the reviewers are "qualified" to do the review, Development and IT Operations management has instituted a documented peer review process.

More controls.

C3: For changes defined as “high risk” (e.g., new areas for the developer, security-sensitive modules such as authentication), changes must also be reviewed by the designated subject matter expert before production deployment.

C4: Automated security testing of the code and environment is performed as part of the deployment pipeline.

And more controls.

C5: All production deployments must have a JIRA ticket number. Deployers must input the JIRA ticket number into the Jenkins build pipeline system for code to be deployed into production.

C6: All production deployments are logged and published through information radiators. Jenkins records all deployments, as well as all corresponding JIRA tickets and the results of all automated and manual tests, release notes, service incidents, peer reviews, and signoffs.

Audit testing and evidence.

CS1 evidence:

a) Static code analysis tool ruleset.
b) Change history for the last five changes made to the static code analysis tool ruleset and review against CCB membership.
c) Report of build statistics for the last six months showing the number of broken builds caused by static code analysis rule violations.
d) Report of the first ten code reviews in Stash and matching JIRA promotion tickets each month for the last six months.
e) Report of every fifth production deployment logged in Jenkins over the last six months.
f) Documented coding standards.

Findings.

1. You’re all good.

The End.

No, wait. It’s based on a true story!

Etsy

Resembles AnyOnlineBusiness, Inc.

Profiled in The DevOps Handbook by Gene Kim, et al.

U.S. Citizenship and Immigration Services
U.S. DEPARTMENT OF HOMELAND SECURITY

Federal agency (i.e., a high-compliance environment).

Achieved continuous deployment (i.e., multiple production deployments *per day*).

Why did the story change?

It had to change.

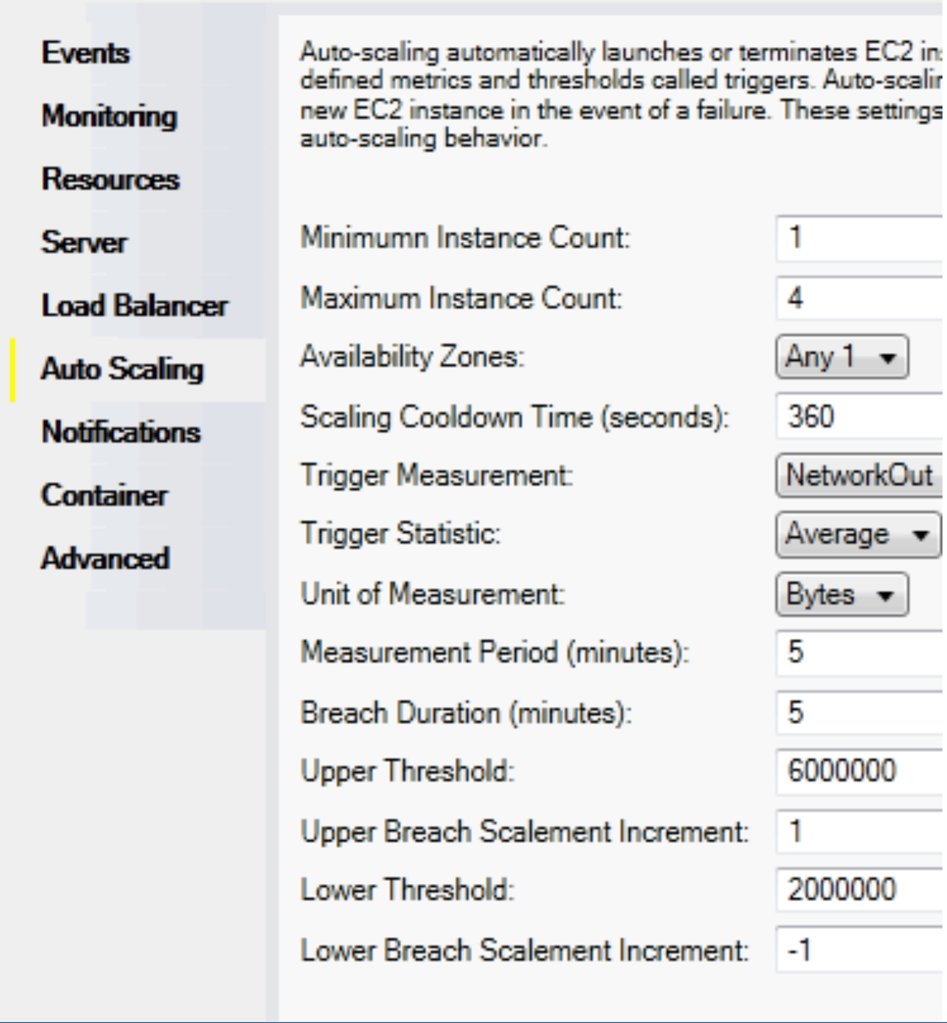
We’ve changed how we do our daily work.

How we view audit, security, and compliance has to change, too.

There was the data center.

“Show me the logs for this machine.”

Then there was the cloud.



Events
Monitoring
Resources
Server
Load Balancer
Auto Scaling
Notifications
Container
Advanced

Auto-scaling automatically launches or terminates EC2 in
defined metrics and thresholds called triggers. Auto-scalir
new EC2 instance in the event of a failure. These settings
auto-scaling behavior.

Minimumn Instance Count:	1
Maximum Instance Count:	4
Availability Zones:	Any 1
Scaling Cooldown Time (seconds):	360
Trigger Measurement:	NetworkOut
Trigger Statistic:	Average
Unit of Measurement:	Bytes
Measurement Period (minutes):	5
Breach Duration (minutes):	5
Upper Threshold:	6000000
Upper Breach Scalement Increment:	1
Lower Threshold:	2000000
Lower Breach Scalement Increment:	-1

Auto-scaling.

Virtual machines start and stop based on configured triggers.

Infrastructure as code.

Software-defined infrastructure.

Automate the creation and tear-down of infrastructure using code.

Infrastructure is ephemeral – not long-lived.

We used to manage infrastructure like this.

Now we manage infrastructure like this.

How do you implement your controls?

#choices

Paper-based processes.

Creates “risk management theater”.

Counter-productive.

Lengthens cycle times and feedback loops.

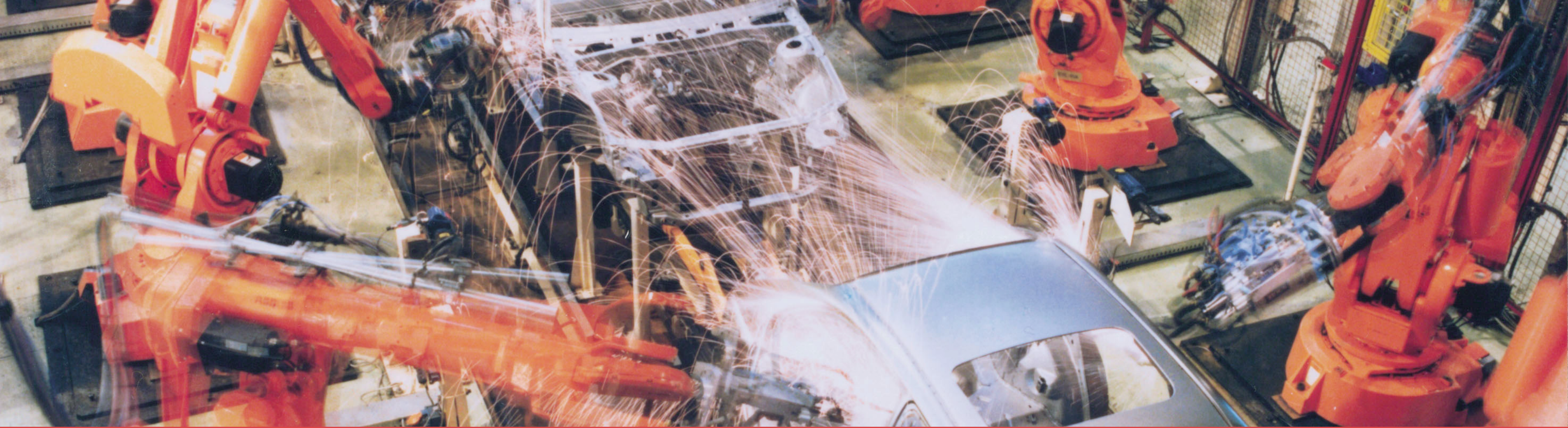
Creates waste and inefficiency.

Involves decision-makers distant from the actual work.

Can be circumvented.

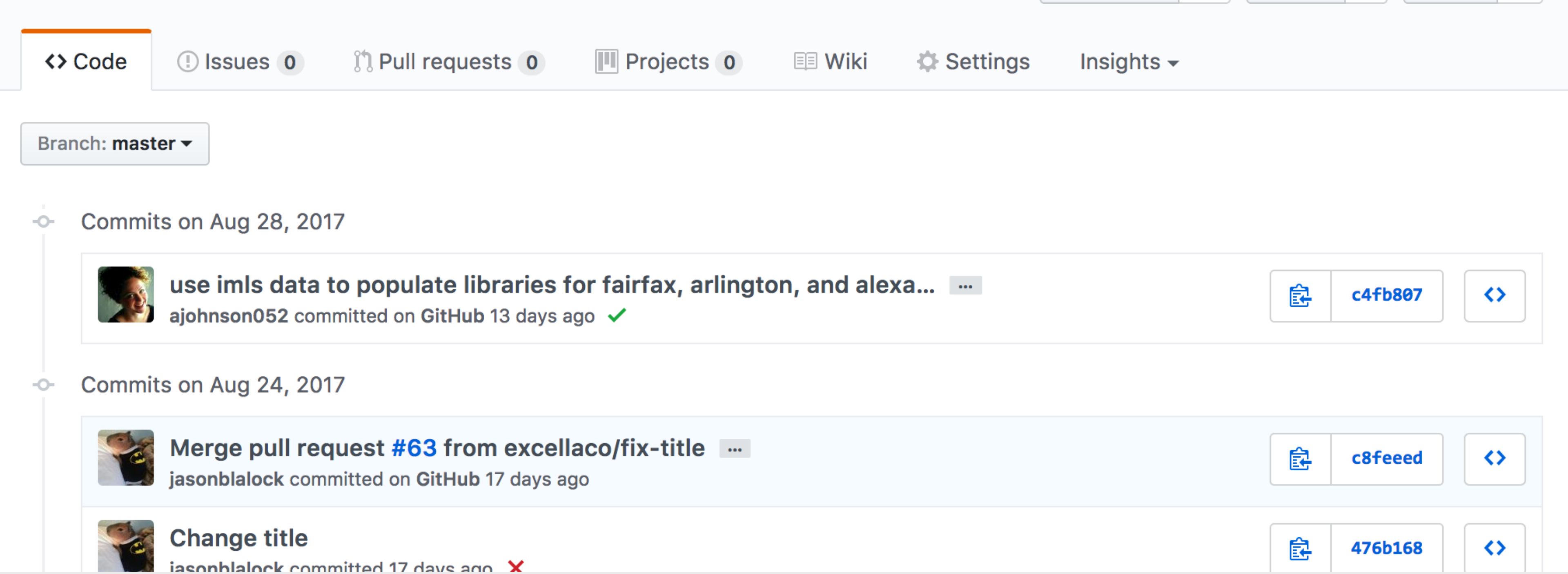
Hurts situational awareness.

Change the story…



Apply automation liberally.

Use the exhaust of normal day-to-day work.



Commits on Aug 28, 2017

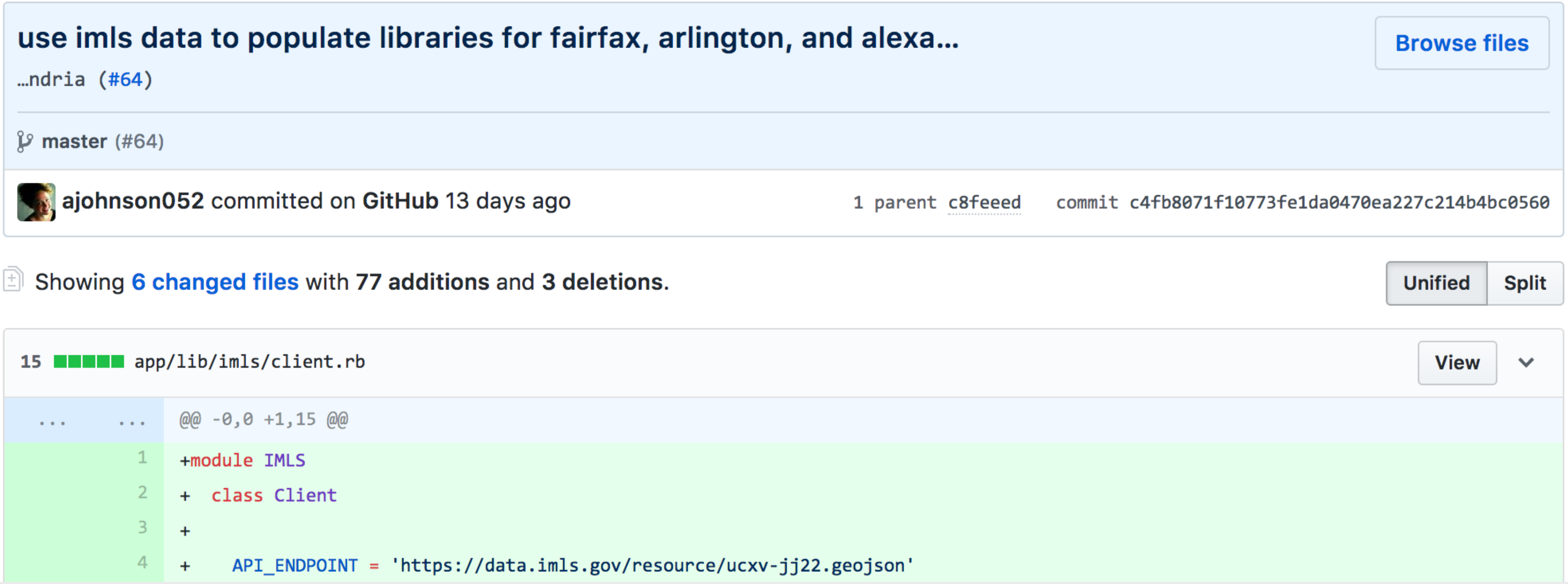
use imls data to populate libraries for fairfax, arlington, and alexa… …
ajohnson052 committed on GitHub 13 days ago ✓ c4fb807

Commits on Aug 24, 2017

Merge pull request #63 from excellaco/fix-title …
jasonblalock committed on GitHub 17 days ago c8feeed

Change title
jasonblalock committed 17 days ago ✗ 476b168

Like change history from source control.


use imls data to populate libraries for fairfax, arlington, and alexa…

…ndria (#64)

ajohnson052 committed on GitHub 13 days ago

1 parent c8feeed   commit c4fb8071f10773fe1da0470ea227c214b4bc0560

Showing 6 changed files with 77 additions and 3 deletions.

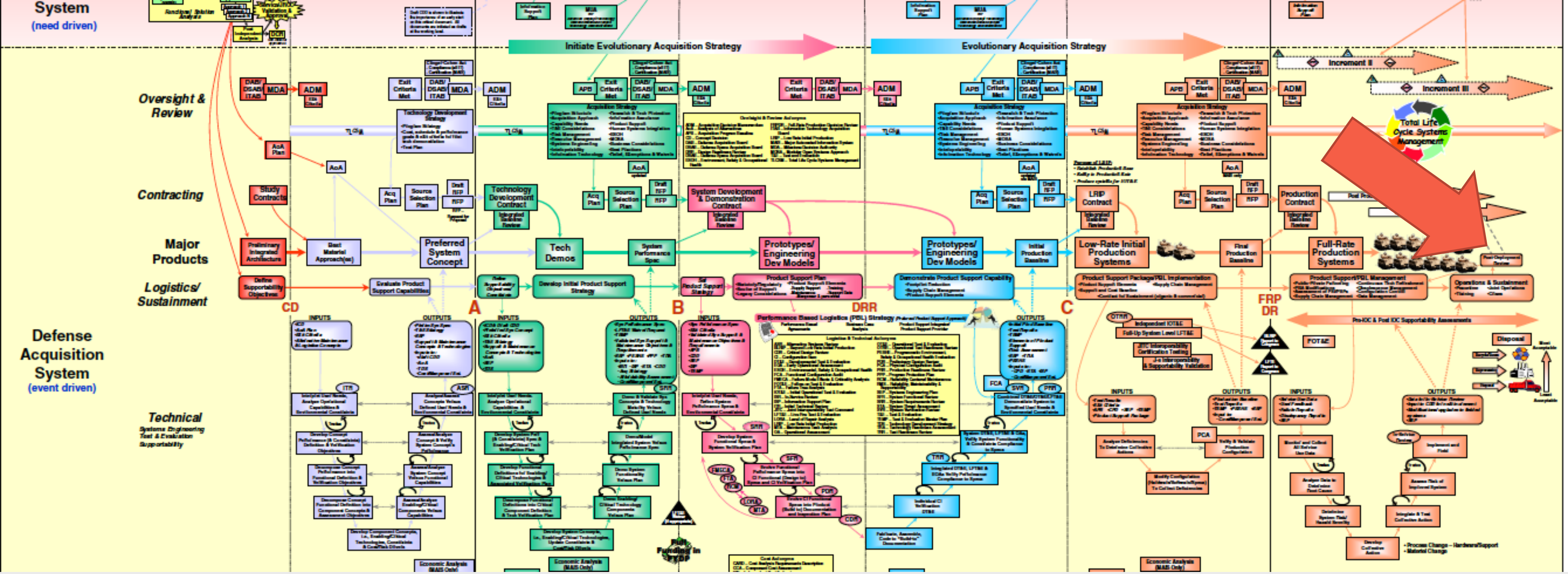
15 ▮▮▮▮▮ app/lib/imls/client.rb

```
@@ -0,0 +1,15 @@
+module IMLS
+  class Client
+
+    API_ENDPOINT = 'https://data.imls.gov/resource/ucxv-jj22.geojson'
```

With the details.

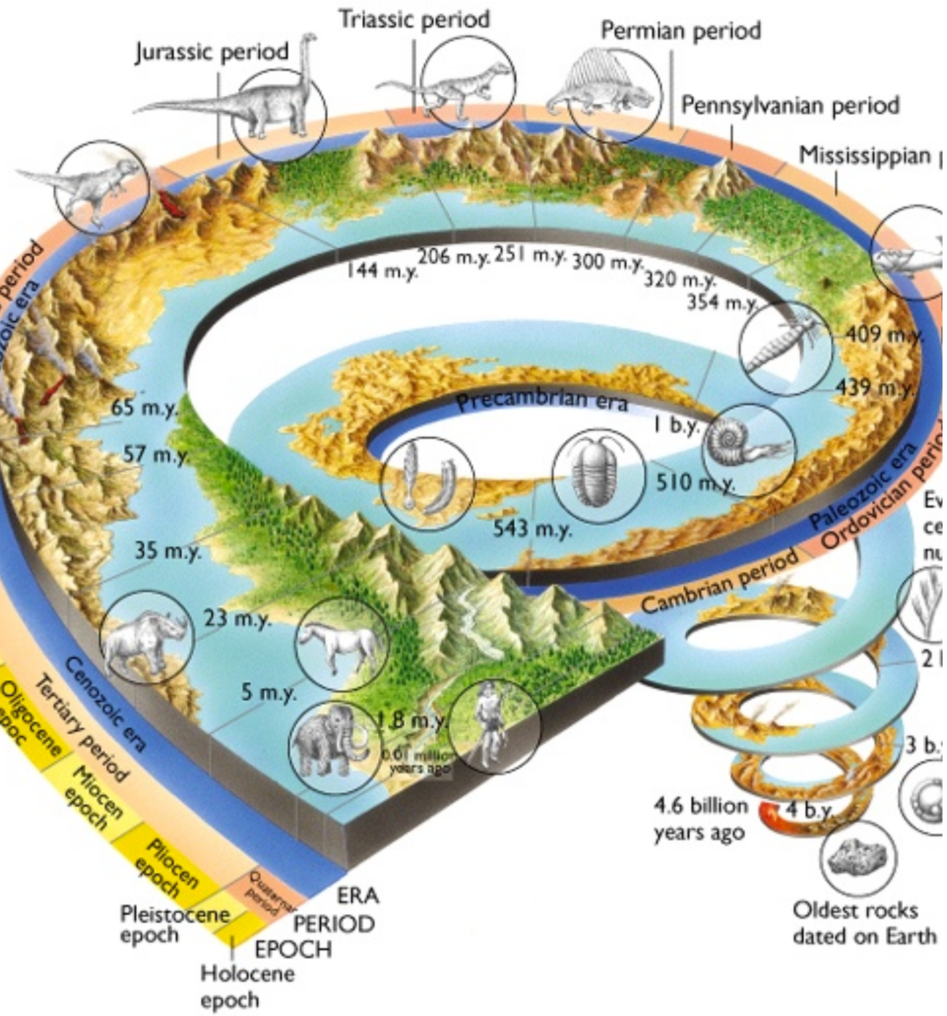
When do you involve Audit, Security, and Compliance?

#choices


System
(need driven)
Initiate Evolutionary Acquisition Strategy
Evolutionary Acquisition Strategy
Increment II
Increment III
Oversight & Review
Contracting
Major Products
Logistics/ Sustainment
Defense Acquisition System
(event driven)
Technical
Systems Engineering
Test & Evaluation
Supportability
Preferred System Concept
Tech Demos
Prototypes/ Engineering Dev Models
Low-Rate Initial Production Systems
Full-Rate Production Systems
Total Life Cycle Systems Management
CD
A
B
DRR
C
FRP DR
ADM
MDA
AoA
Technology Development Contract
System Development & Demonstration Contract
LRIP Contract
Production Contract
Acquisition Strategy
Disposal

Typical entry into the process.

Typical role in the process.



Typical result of the process…

"You can't release until you get approval."

Typical result of the process…

“You can’t release until you fix these.”

Change the story…

Shift left.

Engage audit, security, and compliance early and often.

Integrate non-functional requirements from the beginning with automation.

KNOW the 10 SIGNS

EARLY DETECTION MATTERS

Would you rather know early or late?

"Do painful things more frequently so you can make it less painful."

-- Adrian Cockroft

Shift left.

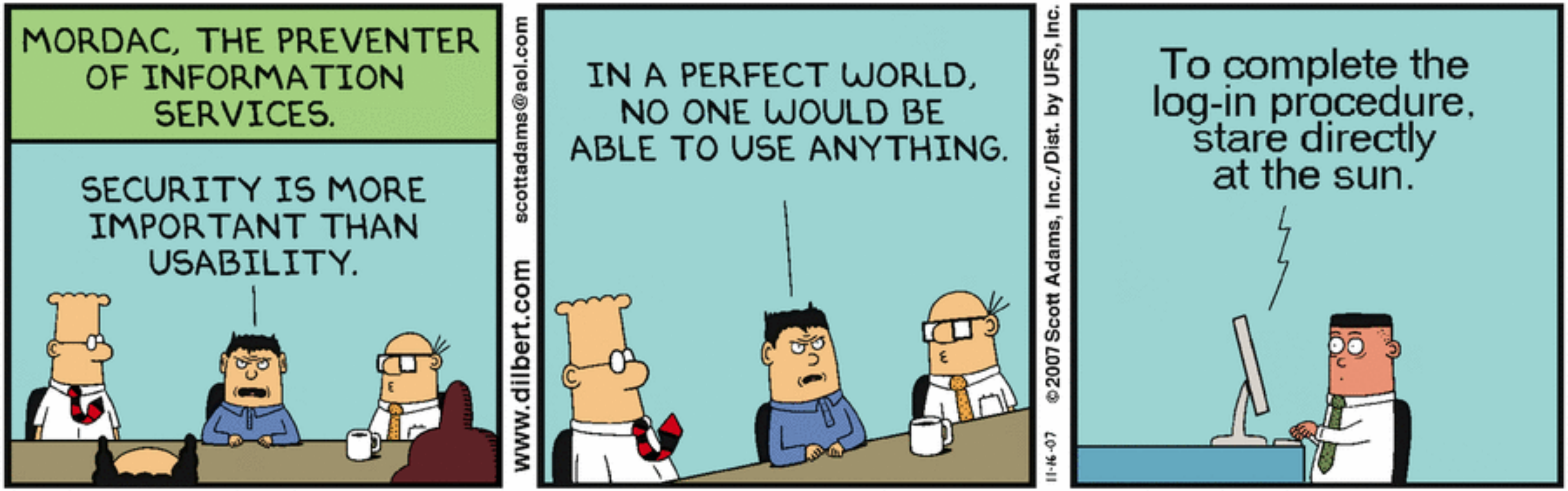
Engage audit, security, and compliance early and often.

Integrate non-functional requirements from the beginning with automation.

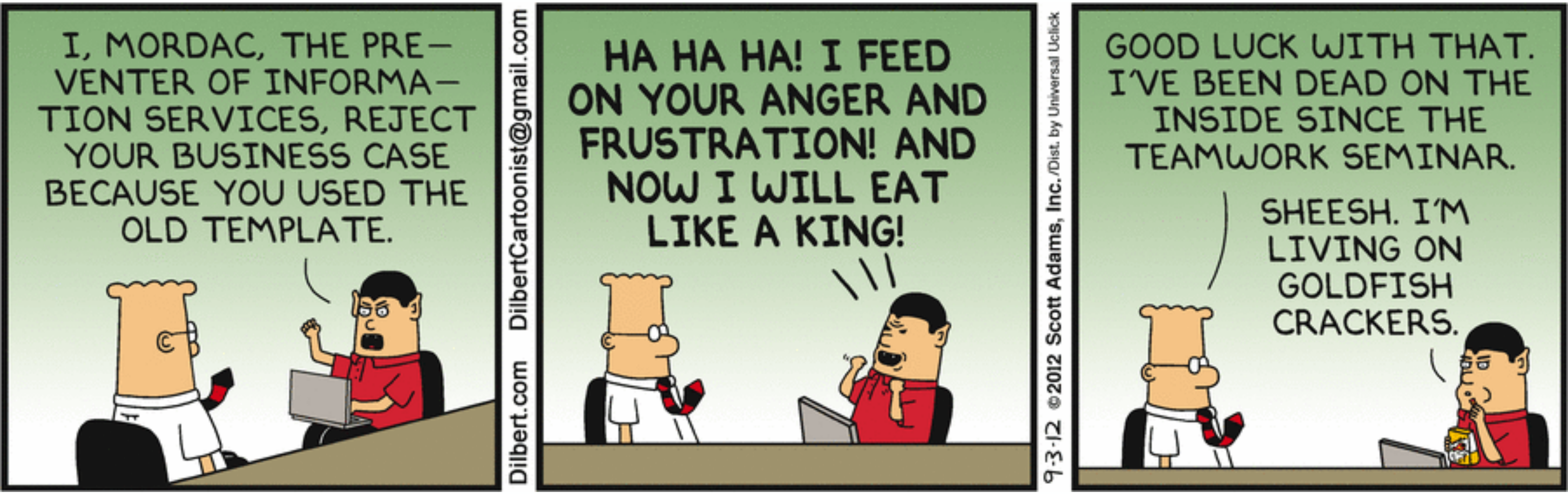
Be transparent and share information.

How do you view Audit, Security, and Compliance?

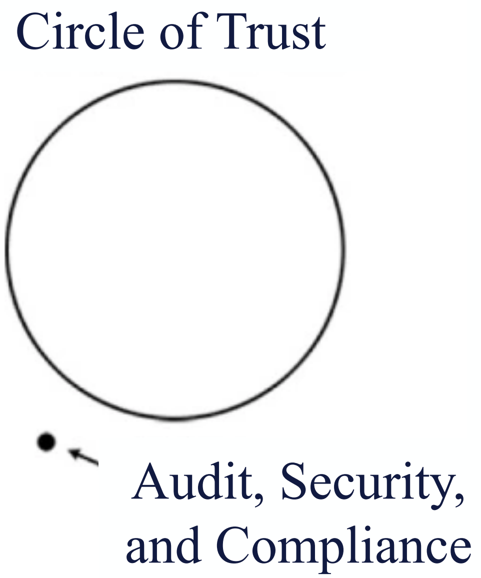
#choices


MORDAC, THE PREVENTER OF INFORMATION SERVICES.
SECURITY IS MORE IMPORTANT THAN USABILITY.
IN A PERFECT WORLD, NO ONE WOULD BE ABLE TO USE ANYTHING.
To complete the log-in procedure, stare directly at the sun.
www.dilbert.com scottadams@aol.com
11-16-07 © 2007 Scott Adams, Inc./Dist. by UFS, Inc.

Unrealistic.


I, MORDAC, THE PRE-VENTER OF INFORMATION SERVICES, REJECT YOUR BUSINESS CASE BECAUSE YOU USED THE OLD TEMPLATE.
Dilbert.com DilbertCartoonist@gmail.com
HA HA HA! I FEED ON YOUR ANGER AND FRUSTRATION! AND NOW I WILL EAT LIKE A KING!
9-3-12 ©2012 Scott Adams, Inc./Dist. by Universal Uclick
GOOD LUCK WITH THAT. I'VE BEEN DEAD ON THE INSIDE SINCE THE TEAMWORK SEMINAR.
SHEESH. I'M LIVING ON GOLDFISH CRACKERS.

Obstructionist.



Untrustworthy.

Change the story…

Why do Audit, Security, and Compliance exist?

A) To make more work for everyone.

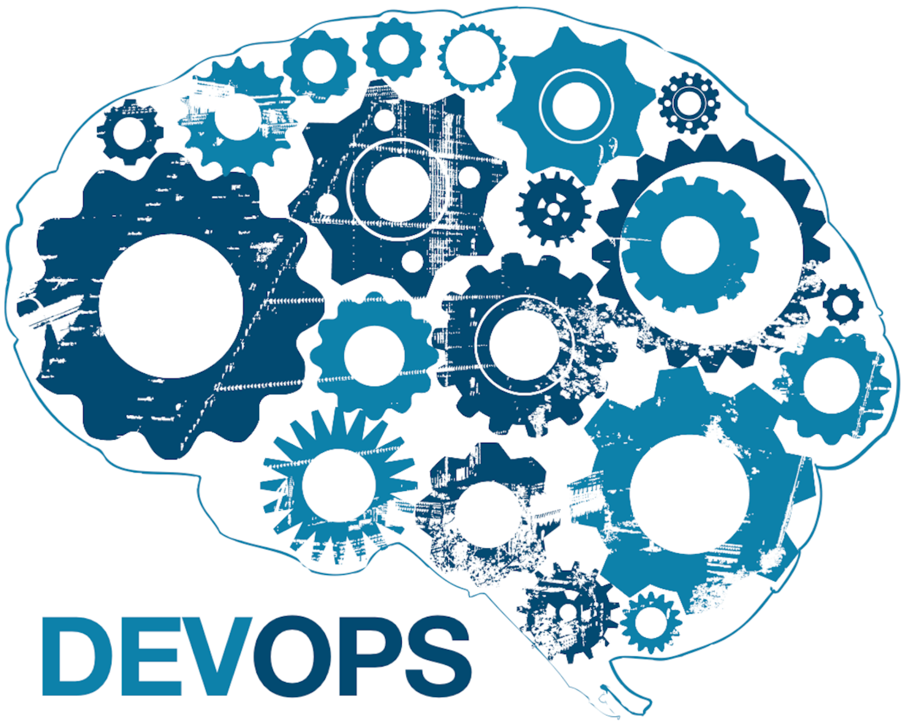
B) To prevent progress.

C) To assign blame.

D) To help the organization manage risk so bad things don’t happen to us.

Put yourself in their shoes.

Show empathy.



"DevOps, a movement of people who care about developing and operating reliable, secure, high performance systems at scale, has always — intentionally — lacked a definition or manifesto."

--2014 State of DevOps Report

It’s just DevOps.

Not DevSecOps…

Or DevAudOps…

Or DevCompOps…

Or DevSecAudCompOps…

…because **we all care** about developing and operating reliable, secure, high performance systems at scale.

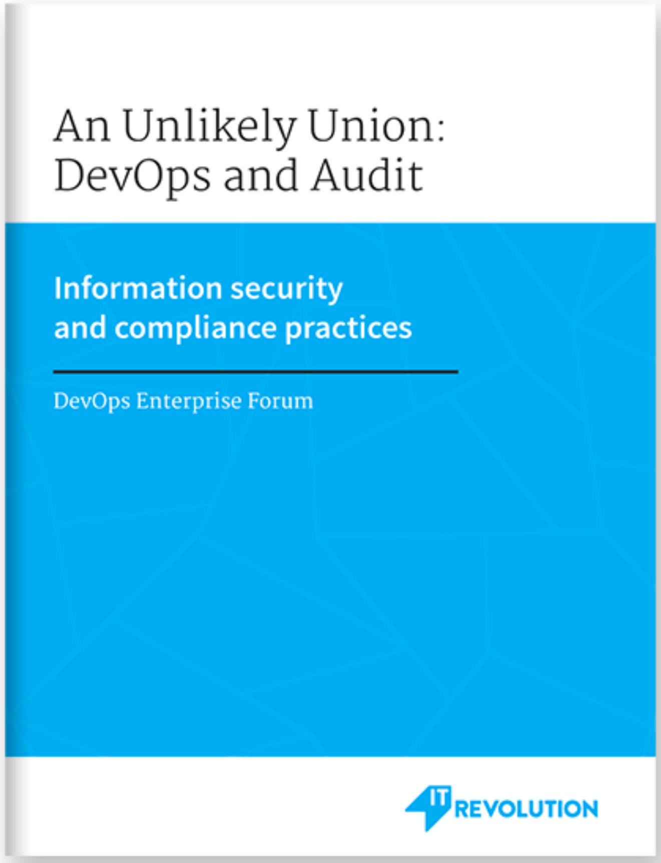
Resources.

#lotstoread

DevOps Audit Defense Toolkit

Provide guidance for how management and auditors should conduct audits in organizations where DevOps practices are in use.

https://cdn2.hubspot.net/hubfs/228391/Corporate/DevOps_Audit_Defense_Toolkit_v1.0.pdf?t=1453925000299



An Unlikely Union:
DevOps and Audit

Information security
and compliance practices

DevOps Enterprise Forum

IT REVOLUTION

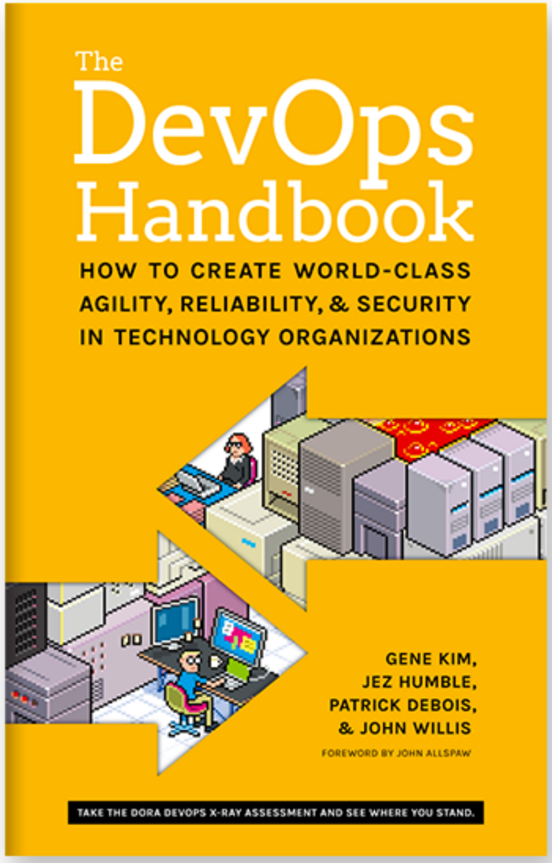
Provides high-level guidance on three major concerns about DevOps practices: DevOps and Change Control, DevOps and Security, DevOps and Separation of Duties.

https://itrevolution.com/book/devops-and-audit/

The Phoenix Project
A Novel About IT, DevOps, and Helping Your Business Win
REVISED WITH NEW RESOURCE GUIDE
Gene Kim, Kevin Behr, and George Spafford

Follow the transformation of John Pesche, the black-binder-wielding Chief Information Security Officer whose constant meddling under the guise of improving security has turned him into a pariah.

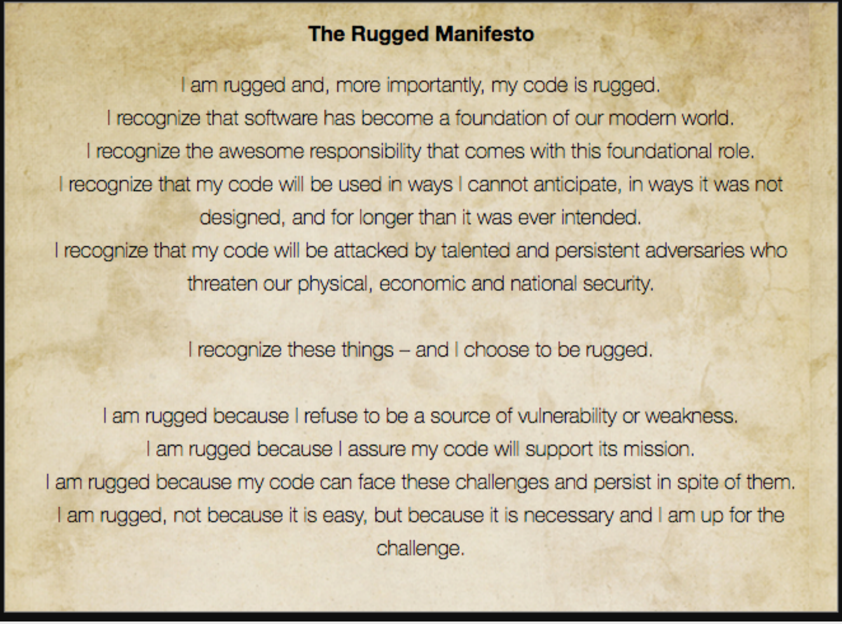
https://itrevolution.com/book/the-phoenix-project/


The DevOps Handbook
HOW TO CREATE WORLD-CLASS AGILITY, RELIABILITY, & SECURITY IN TECHNOLOGY ORGANIZATIONS
GENE KIM, JEZ HUMBLE, PATRICK DEBOIS, & JOHN WILLIS
FOREWORD BY JOHN ALLSPAW
TAKE THE DORA DEVOPS X-RAY ASSESSMENT AND SEE WHERE YOU STAND.

Includes two chapters on integrating audit, security, and compliance with a number of case studies.

https://itrevolution.com/book/the-devops-handbook/

Posted on **September 4, 2012**



The Rugged Manifesto

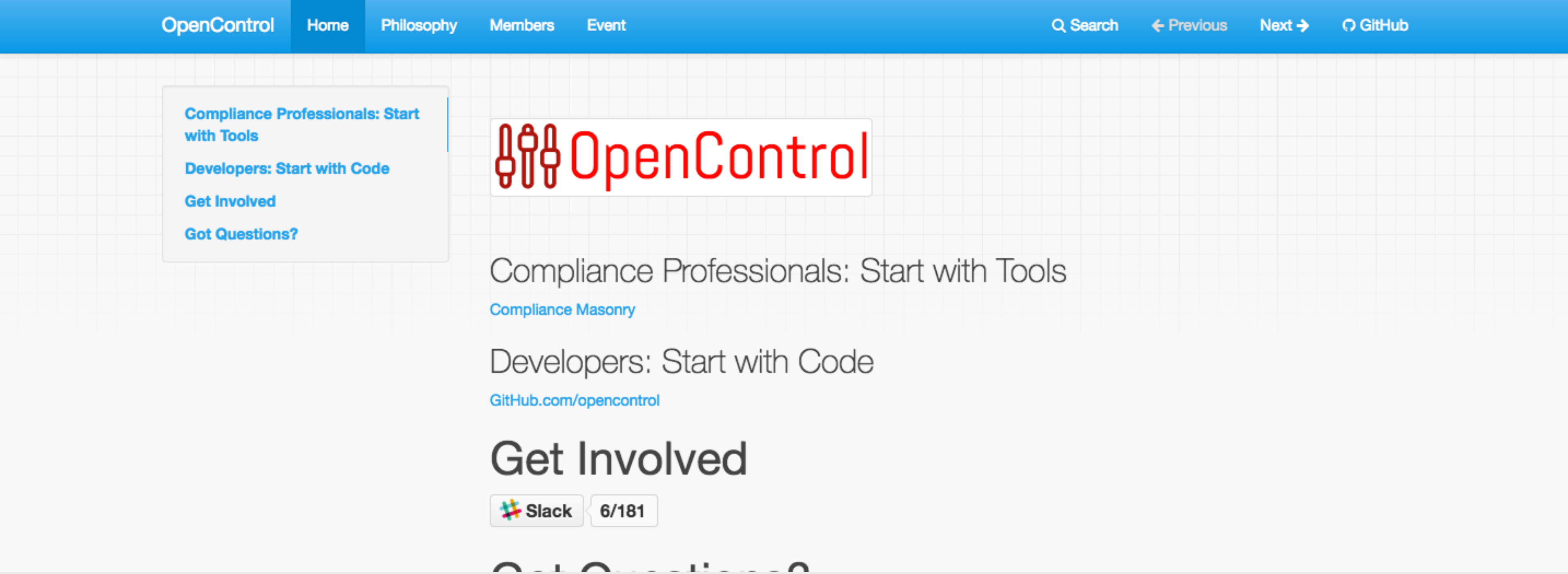
I am rugged and, more importantly, my code is rugged.
I recognize that software has become a foundation of our modern world.
I recognize the awesome responsibility that comes with this foundational role.
I recognize that my code will be used in ways I cannot anticipate, in ways it was not designed, and 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.

The Rugged Manifesto.

https://www.ruggedsoftware.org/


OpenControl Home Philosophy Members Event Search Previous Next GitHub

Compliance Professionals: Start with Tools
Developers: Start with Code
Get Involved
Got Questions?

Compliance Professionals: Start with Tools

Compliance Masonry

Developers: Start with Code

GitHub.com/opencontrol

Get Involved

Slack 6/181

OpenControl.

http://open-control.org/

Hacking the federal bureaucracy.

Mark Schwartz

Former CIO, USCIS

DevOps Enterprise Summit

October 21 - 23, 2014

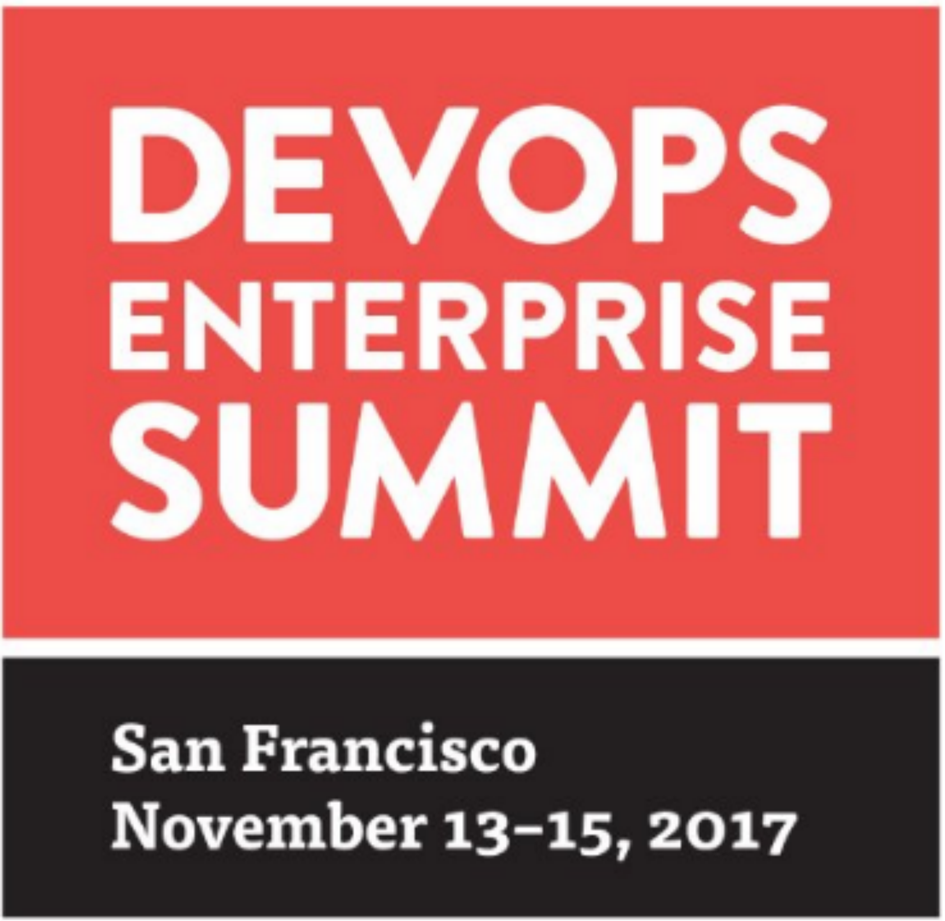
https://www.youtube.com/watch?v=QwHVlJtqhaI

Positioning agile and continuous delivery for auditors and examiners.

Simon Storm

Director, Promontory Interfinancial Network

https://www.youtube.com/watch?v=P2C7uIHgotA



DEVOPS ENTERPRISE SUMMIT

San Francisco
November 13-15, 2017

DevOps in real-life.

Hear from experts about how they did it.

https://events.itrevolution.com/

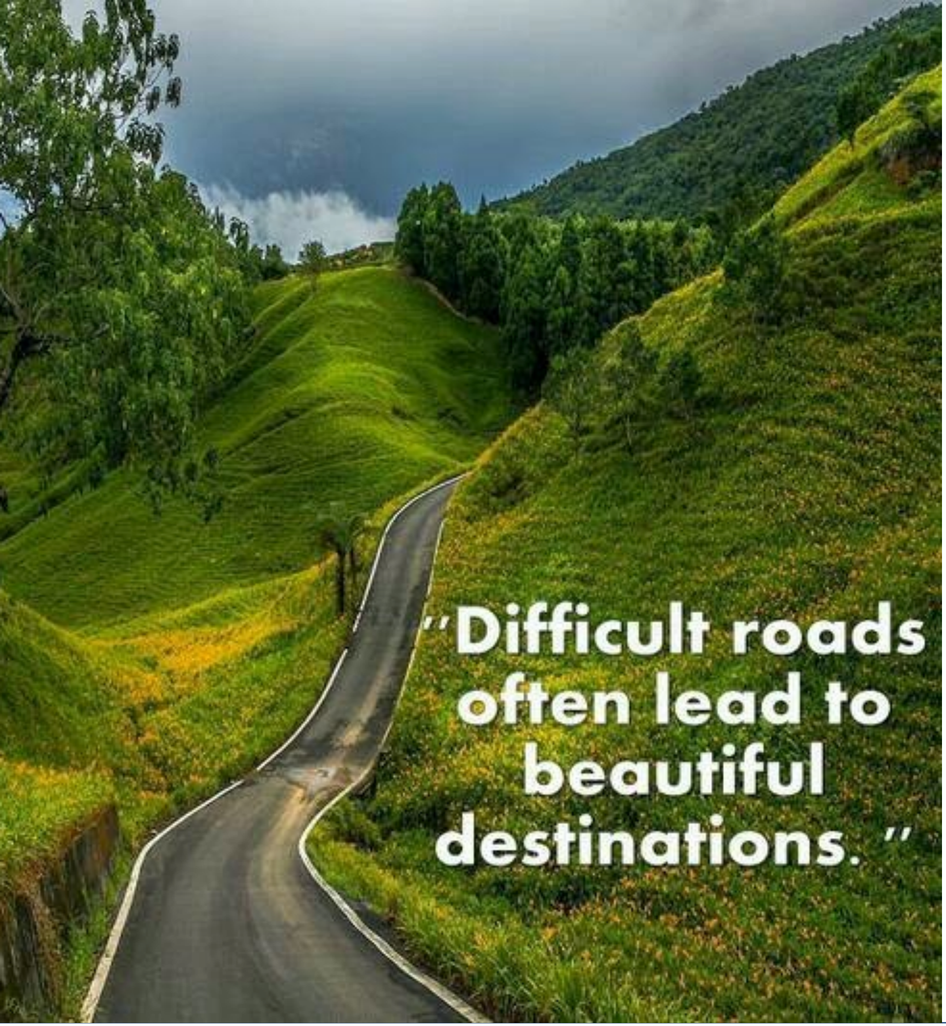
Final thought.

We’re all on the same team.

Change is possible.

It may be bumpy.

And slower than we would like.



But it will be worth it.

Jeff Gallimore

jeff.gallimore@excella.com

@jgallimore

http://itsanicelife.com

https://www.linkedin.com/in/jgallimore

fin.