DevOps and the Future of Change Management

Mike Long @meekrosoft

www.merkely.com
“Ultimately the winners in banking will have the capabilities of a world-class software company”

Rich Fairbank Capital One Founder & CEO
Software is Changing

- **Monolithic**
  - Slow Changing
  - Big Servers

- **Loosely Coupled**
  - Rapidly Updated
  - Many Small Servers and Functions

Timeline:
- 1 change/year
- 10 changes/year
- 100 changes/year
- 1000 changes/year
- 10000 changes/year
- 100000 changes/year
Change Management Must Change

The old ways of managing change can’t keep up with modern devops teams:

- Change Windows
- CAB Meetings
- Manual Documentation
- Batched Changes

These methods are proven to be ineffective at mitigating risk*

We found that external approvals were negatively correlated with lead time, deployment frequency, and restore time, and had no correlation with change fail rate. In short, approval by an external body (such as a manager or CAB) simply doesn’t work to increase the stability of production systems, measured by the time to restore service and change fail rate. However, it certainly slows things down. It is, in fact, worse than having no change approval process at all.

Why is it like this?

Security in change management

47. APRA envisages that a regulated entity would implement controls to manage changes to information assets, including changes to hardware, software, data, and configuration (both where the change is planned and in response to an emergency) with the aim of maintaining information security. This would typically include:

a) security testing [including reviews] to identify vulnerabilities and confirm information security requirements have been met. The nature of testing would be commensurate with the scope of the change and the sensitivity and criticality of the impacted information asset [refer to Attachment H for examples of common testing techniques];

b) approval of changes prior to deployment into the production environment;

c) segregation of duty controls which prevent personnel from deploying their own software changes to production;

d) changes are developed and verified in another environment, sufficiently segregated from production so as to avoid any compromise of information security;
What does an audit look like?

- What is your software development process?
- How do you ensure conformance to the process? Can you prove it?
- What software is currently in production?
- What changes were in the last release?
- What was deployed on 13/02/2020?
- Who accepted and signed off on the risk?

Regulated Industries are obliged to manage change.
What is Change Management Anyway?

Governance Framework

Requirements

Definition

Implementation

Proof
Traditional Change Management

Governance Framework

Requirements
- Definition
- Implementation
- Proof

Traditional Change Management
- Wiki Documentation
- Manual build, QA, and deployment
- Meeting minutes and Change Documentation

Works with low change volume

SDLC
1.1 ....
1.2 ....
1.2.1 ....

Release Docs
Version: 2021.01.21
Approved by: Boss
....
....
....

Approved by: Boss
Change Management in “DevOps”

Governance Framework

Requirements

- Definition
- Implementation
- Proof

Wiki Documentation

- SDLC
  - 1.1 ..
  - 1.2 ..
  - 1.2.1 ..

DevOps Pipelines

Works With medium change volume

Meeting minutes and Change Documentation

Release Docs
Version: 2021.01.21
Approved by: Boss
...
...
...

Approved by: Boss
Lead Time for Changes with “DevOps”

Change Management as a gate

High ceremony CM compliance leads to batching, longer lead times, and higher release risks.

Lead time
DevOps Change Management

Governance Framework

- Definition
- Implementation
- Proof

Live Documentation

DevOps Pipelines

Automated Audit trail

Requirements

DevOps Change Management

Works With high change volume

Release Docs
Version: 2021.01.21
Approved by: Boss
....
....
....

merkely.com
Lead Time for Changes with DevOps

Change Management as a gate

High ceremony CM compliance leads to batching, longer lead times, and higher release risks.

Continuous Change Management

Continuous compliance leads to incremental delivery, shorter lead times, and lower release risks.
Change Management Comparison

Governance Framework

Requirements

- Definition
- Implementation
- Proof

Traditional Change Management

- Wiki Documentation
- Manual build, QA, and deployment
- Meeting minutes and Change Documentation

Change Management in “DevOps”

- Wiki Documentation
- DevOps Pipelines
- Meeting minutes and Change Documentation

DevOps Change Management

- Live Documentation
- DevOps Pipelines
- Automated Audit trail
Insider Threat

“At Google, we want to do as much as we can to minimize the potential for Google personnel to use their organizational knowledge or access to user data in an unauthorized way—this includes running an unauthorized job”

https://cloud.google.com/security/binary-authorization-for-borg/
A Secure Chain of Custody across the value stream

Cryptographically ensure deployments are known, approved and compliant.

- Product Management
- Binary Provenance
- Release Approval
- Deployment Control
- Production
- Monitoring
Binary Provenance

Fingerprints in a controlled build process provides a tamper-proof identity for all binaries across the value stream.
Control Evidence

- Build
  - Create Artifact
  - Pull Request Data
  - Unit Test Result

- Test
  - Functional Test Result

- Security Analysis
  - Analysis Result

- Deploy to Staging
  - Log Deployment

- Deploy?

A Secure Audit Trail to prove compliance based on open standards
Documented Approvals

Automated change reports generated via version control or continuous integration events.
Automated Deployment Controls

Automatically ensure **only compliant software is deployed** by verifying binary provenance automatically as part of your deployment process.
Merkely delivers Change Management at DevOps scale

- A Secure Chain of Custody
- Risk Controls as Code
- Provable audit trails
- Real time process documentation
- Empowered Teams
Merkely

Mike Long
@meekrosoft
mike@merkely.com
+47 48676360