RSAConference2016

San Francisco | February 29 – March 4 | Moscone Center

Connect to Protect

SESSION ID: ASD-T09

Product Security at Internet Scale

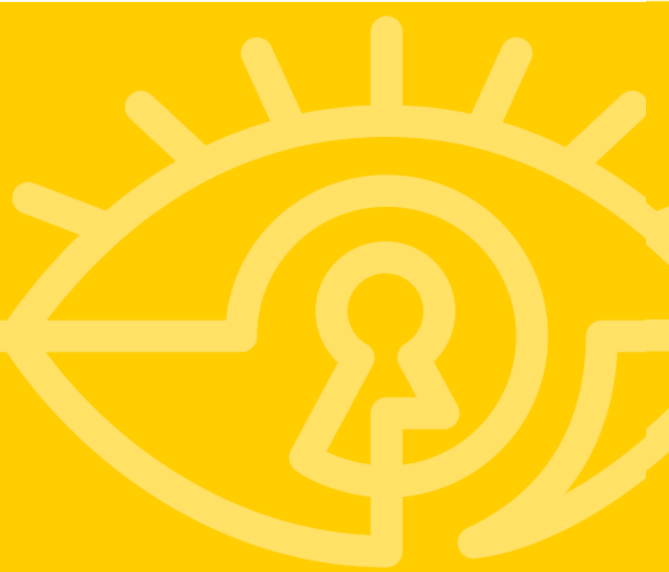
Michael Murray

Director – Product Development Security
GE Healthcare
@mmurray

#RSAC

Overview

- Organization Culture - Two ways to get to internet scale:
  - Start the company building toward it
  - Start as a legacy company and transition to it
- Changing the SDL
  1. Instrumentation == securable
  2. Move security as close to developer as possible
  3. Find security bottlenecks/constraints and eliminate them



From Legacy to Scale

Subhead if needed

The Transition to Cloud

Ignoring Cloud

Cloud products are relatively low risk (in context of company revenue/delivery)

Security provided ad hoc to each cloud product

High duplication of effort, low level of standardization

Often, security is largely ignored at each individual cloud product

Discovering Cloud

Cloud platform security becomes paramount importance

Security becomes standardized as part of platform

IT and platform security become duplicative

Duplicate functions lead to large amounts of waste or encourage risk taking

Cloud Integrated

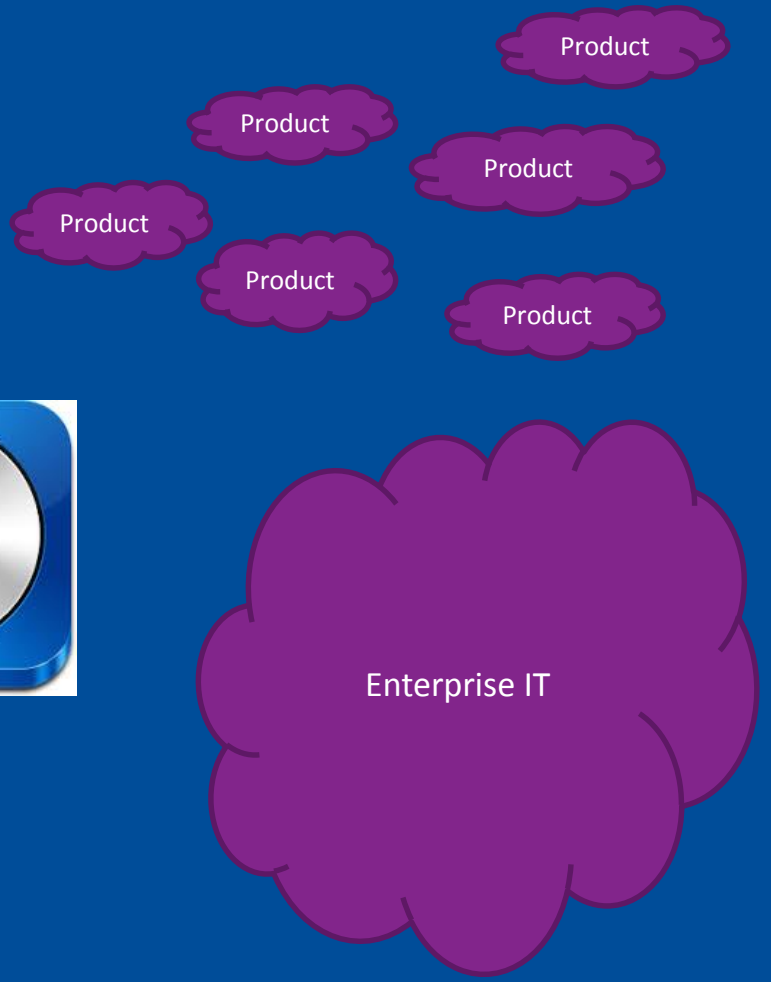
Security is integral to delivery – security risk of platform compromise is risk to entire business

Because IT services are platform integrated, IT security and platform security become single effort

Waste and redundancy eliminated

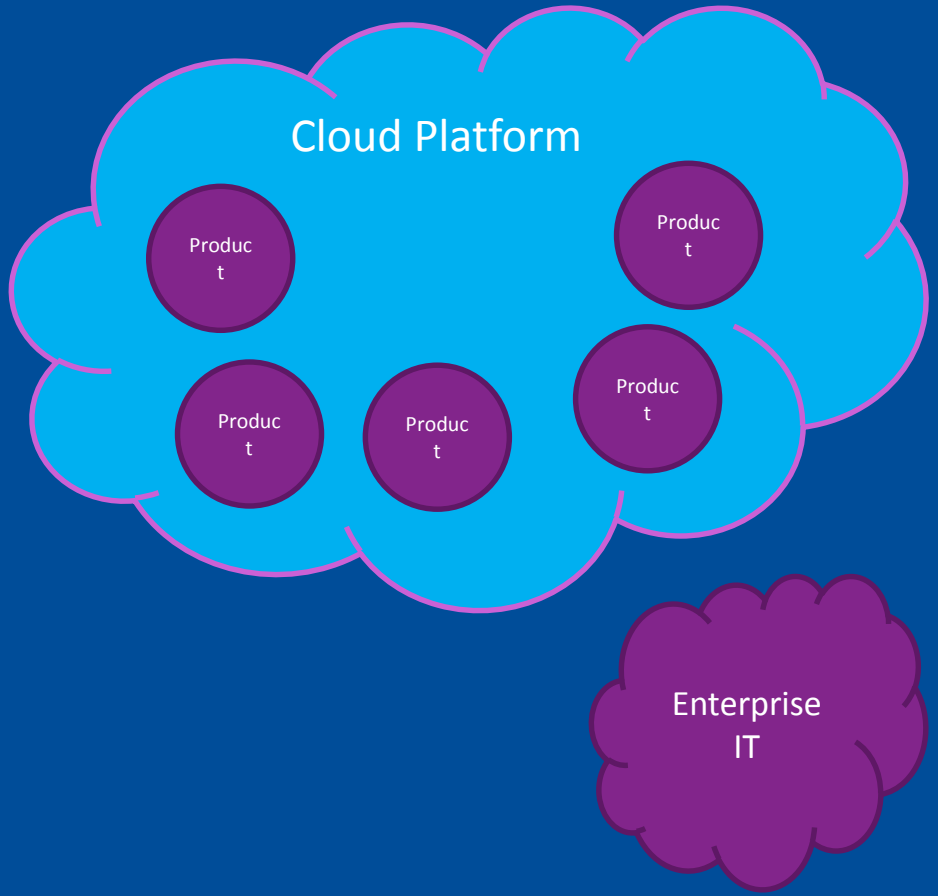
Ignoring the Cloud

- Products are largely shrinkwrapped
- Some cloud products , usually run as one-off environments by distributed product teams
- Usually no coordination between orgs that support cloud programs
- IT organization cloud program usually has no overlap with hosted products



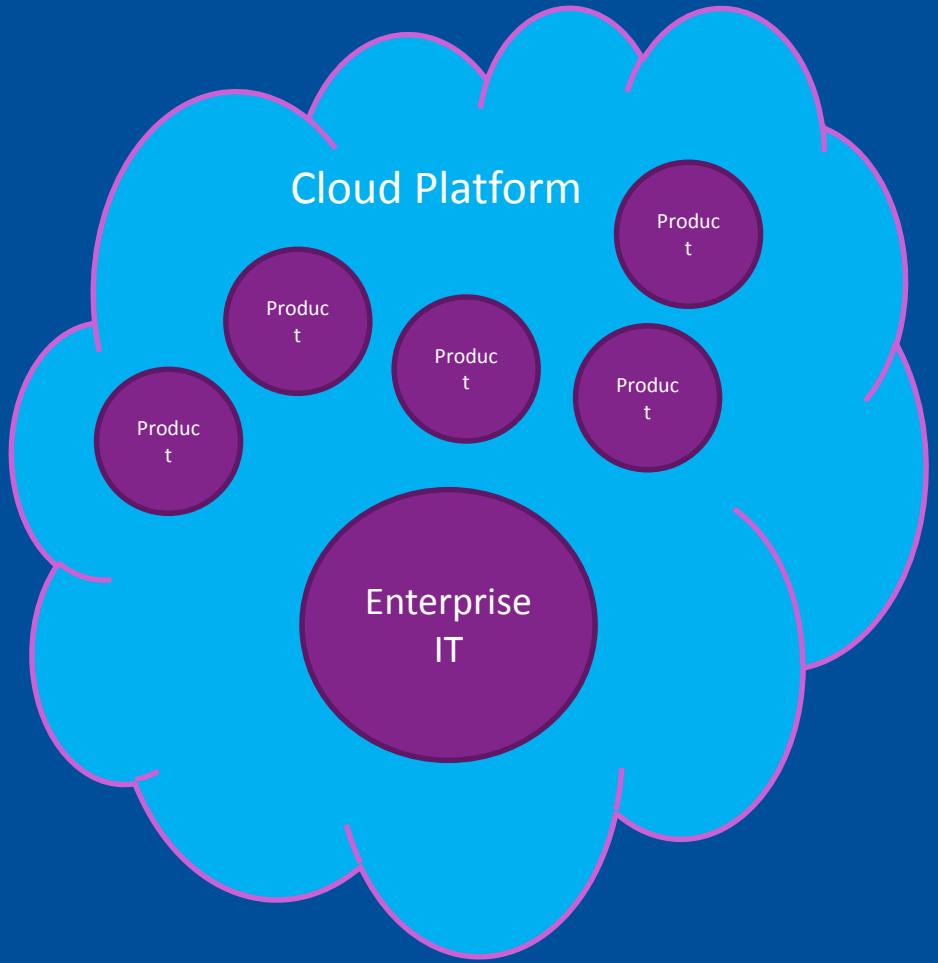
Discovering the Cloud

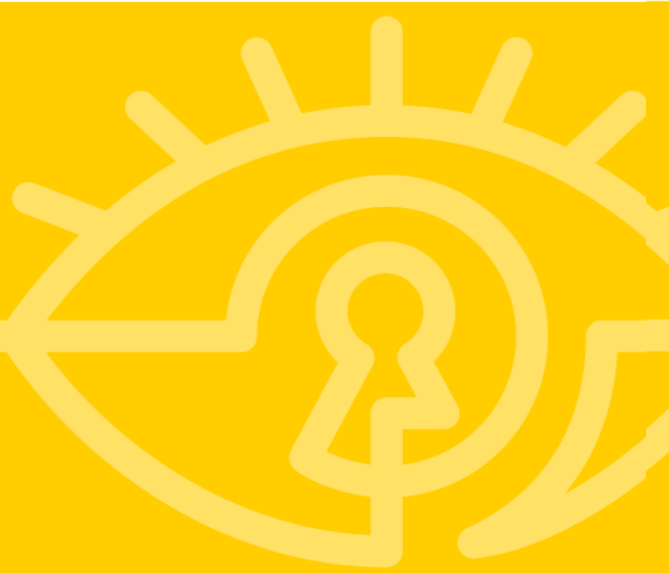
- Product organization declares intent to make cloud delivery the strategic initiative of product delivery
- Platforming initiatives take precedence over building-delivery of shrinkwrapped services
- IT organization is seen as separate from product platform initiatives



Cloud Centric

- Product delivery is indistinguishable from delivery of IT services
- Product platform becomes the central technology for delivery of all product relevant services
- IT organization and product delivery become indistinguishable





Security Concerns at Scale

Subhead if needed

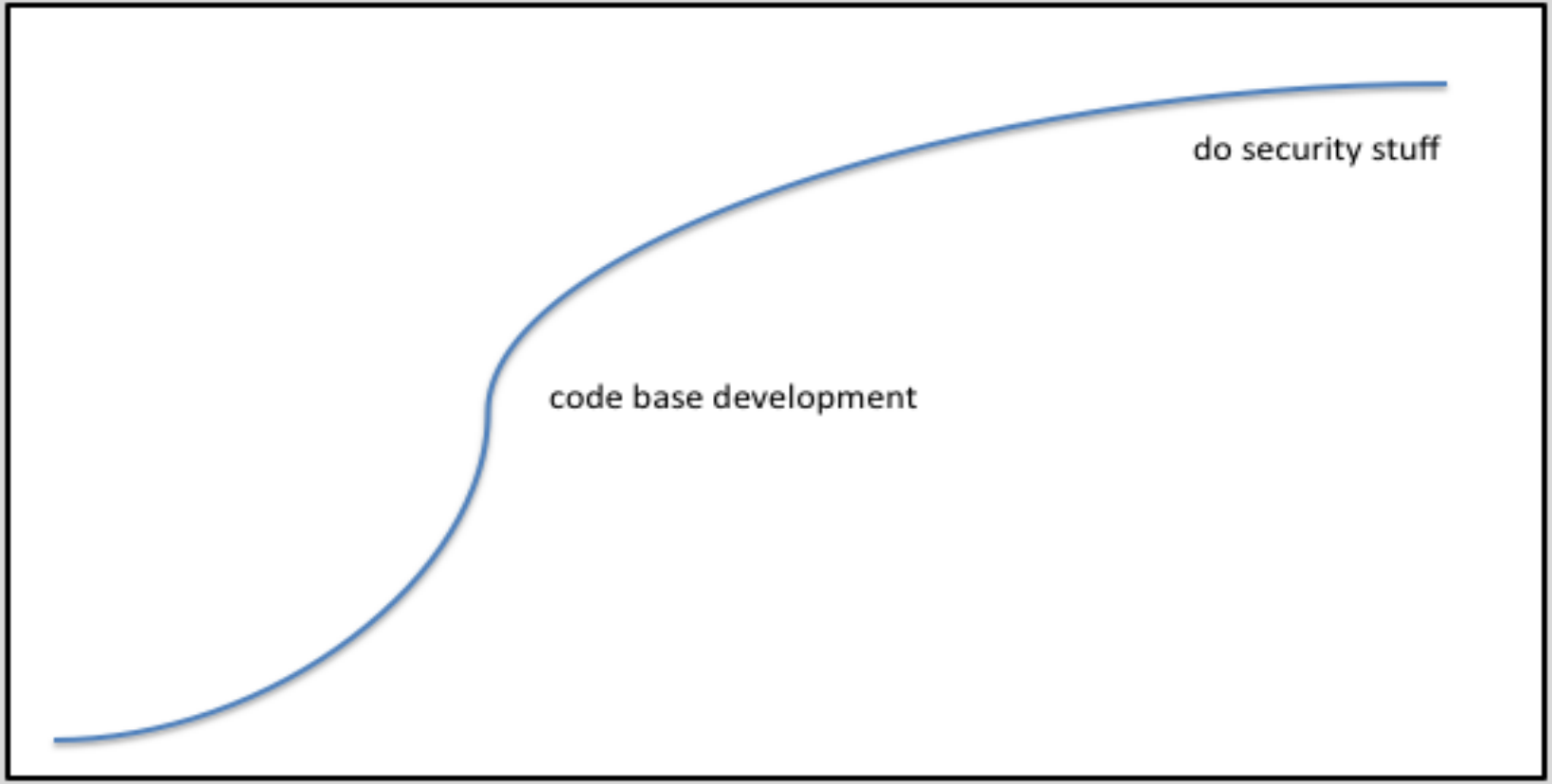
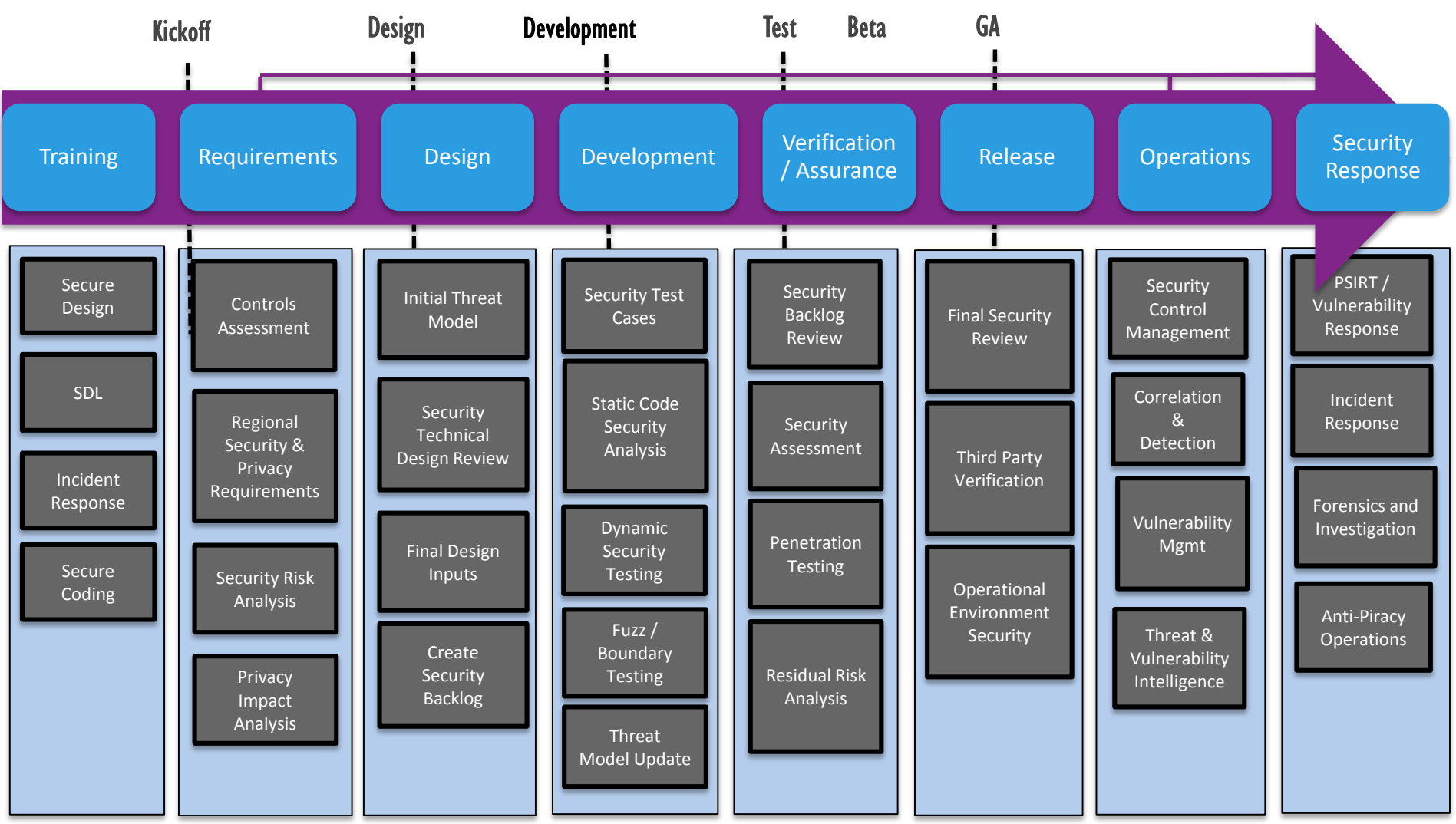


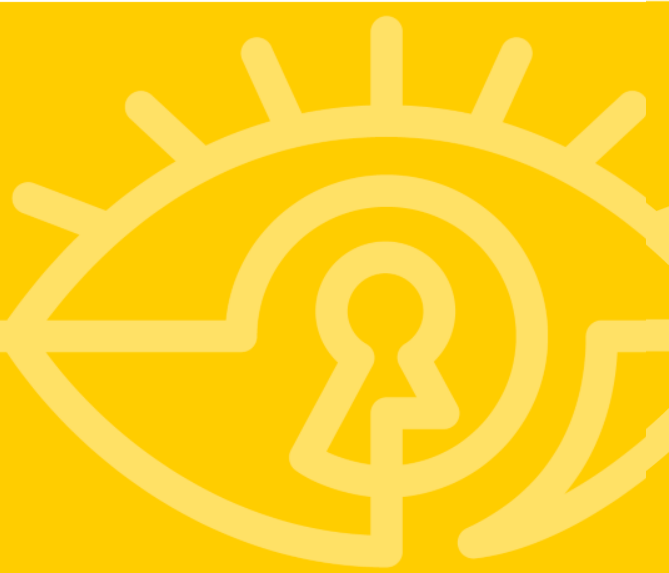
Diagram from: https://medium.com/startup-securely/minimum-security-budget-msb-2ca99488cb53#.sj0swii7r


Kickoff
Design
Development
Test
Beta
GA
Training
Requirements
Design
Development
Verification / Assurance
Release
Operations
Security Response
Secure Design
SDL
Incident Response
Secure Coding
Controls Assessment
Regional Security & Privacy Requirements
Security Risk Analysis
Privacy Impact Analysis
Initial Threat Model
Security Technical Design Review
Final Design Inputs
Create Security Backlog
Security Test Cases
Static Code Security Analysis
Dynamic Security Testing
Fuzz / Boundary Testing
Threat Model Update
Security Backlog Review
Security Assessment
Penetration Testing
Residual Risk Analysis
Final Security Review
Third Party Verification
Operational Environment Security
Security Control Management
Correlation & Detection
Vulnerability Mgmt
Threat & Vulnerability Intelligence
PSIRT / Vulnerability Response
Incident Response
Forensics and Investigation
Anti-Piracy Operations

Making Security Agile:

"Instead of completing all bucket requirements each sprint, product teams must complete only one SDL requirement from each bucket of related tasks during each sprint"

Security Development Lifecycle for Agile Development, Version 1.0
Microsoft



Scale Means Speed

Subhead if needed

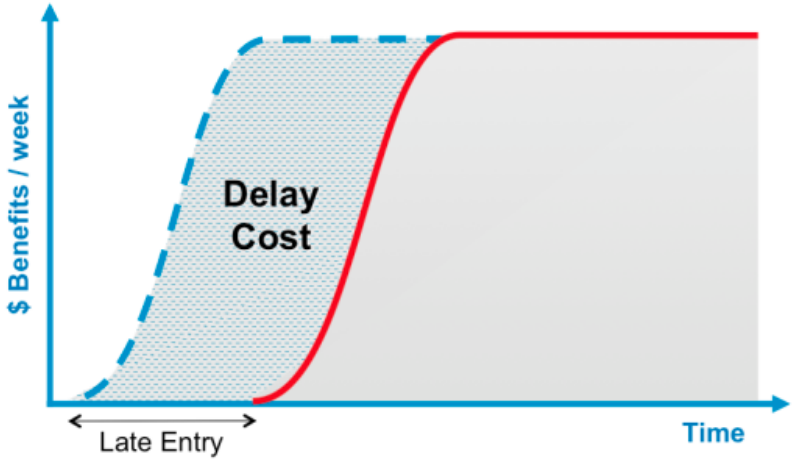
JOBS I'VE BEEN FIRED FROM

FEDEX DRIVER
CRANE OPERATOR
SURGEON
AIR TRAFFIC CONTROLLER
PHARMACIST
MUSEUM CURATOR
WAITER
DOG WALKER
OIL TANKER CAPTAIN
VIOLINIST
MARS ROVER DRIVER
MASSAGE THERAPIST

HTTP://XKCD.COM/1428/

Cost of delay

- Agile developers love to discuss cost of delay
- Every unit of delay in software costs us some amount of revenue opportunity and some amount of waste of development resources
- The key to reducing cost is reducing the amount of wasted time during the development lifecycle
- This cost is usually significantly larger than most other costs in the development lifecycle



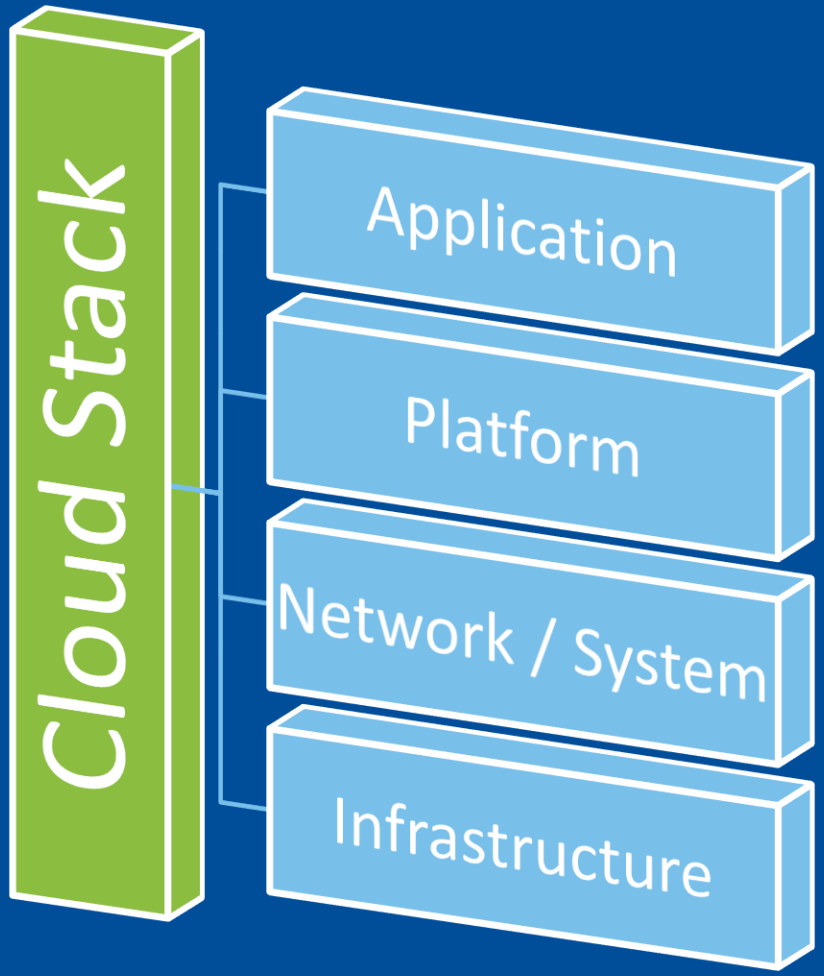
Example:

- Program with **$20M/year revenue and $5M/year development** cost
- Weekly revenue: **$385K**
- Development: **$96K**
- **Cost of Delay: $481K / week**

Securability

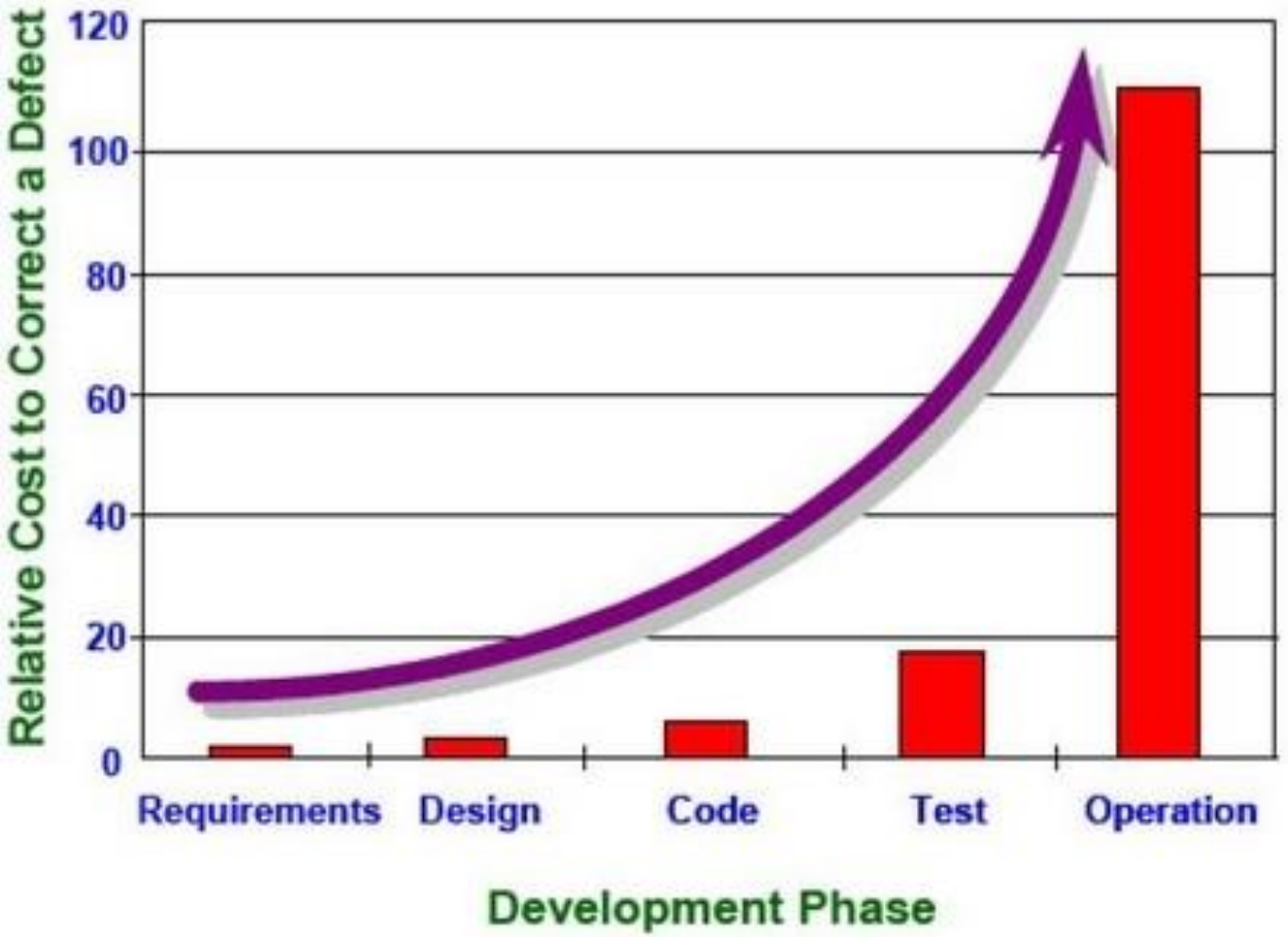
Would you rather….

1. Fix all the vulnerabilities
2. Implement robust logging and Instrumentation



- Security responsibilities exist at each layer – instrumentation required at each level
- Controls required for:
  - Protection (e.g. firewall)
  - Detection (e.g. IDS, AV, anti-malware)
  - Response (e.g. IR, forensics)
- You must heavily instrument each level of the stack to understand the control environment
- Better off leaving out controls in favor of more instrumentation

Move Security Toward the Developer


Relative Cost to Correct a Defect
120
100
80
60
40
20
0
Requirements
Design
Code
Test
Operation
Development Phase

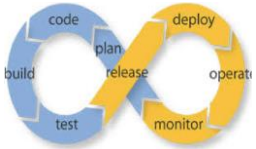
Automation

Security Backlog

- Keep and manage a backlog of security tasks and review regularly with management
- Drive high value security tasks
- Minimize surprises and encourage communication and accountability

Security Unit Testing

- Write ,teach and drive security focused test cases development across the development organization
- Allow security assurance for Agile “always shippable state”



CI/CD Security Testing

- Discover vulnerability early in the development lifecycle
- Integrate further security tests in to the CI/CD development pipeline.
- Additional Testing
  - Security Static analysis for all languages
  - Security dynamic testing
  - Fuzzing / Securtity Robustness

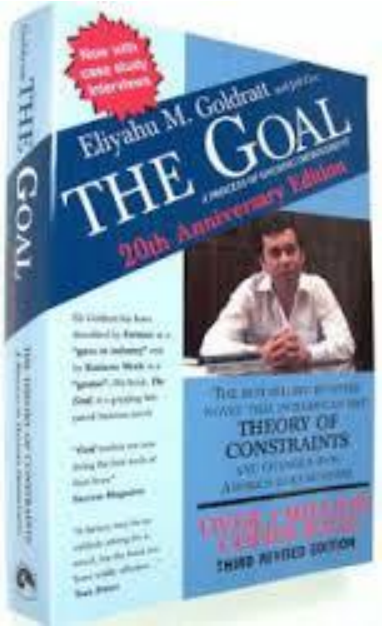
Penetration Testing

- Apply real-life attack simulation to detect advanced security vulnerabilities
- Work with penetration test vendors to come up with a devops compatible testing strategy

Elimination of Constraints

Theory of Constraints

- Any multi-step process has a limiting step – the constraint
- To optimize production, you must “exploit the constraint”
  - Find the constraint
  - Determine how to optimize throughput of that process step
  - This will create a new constraint somewhere else – repeat this process

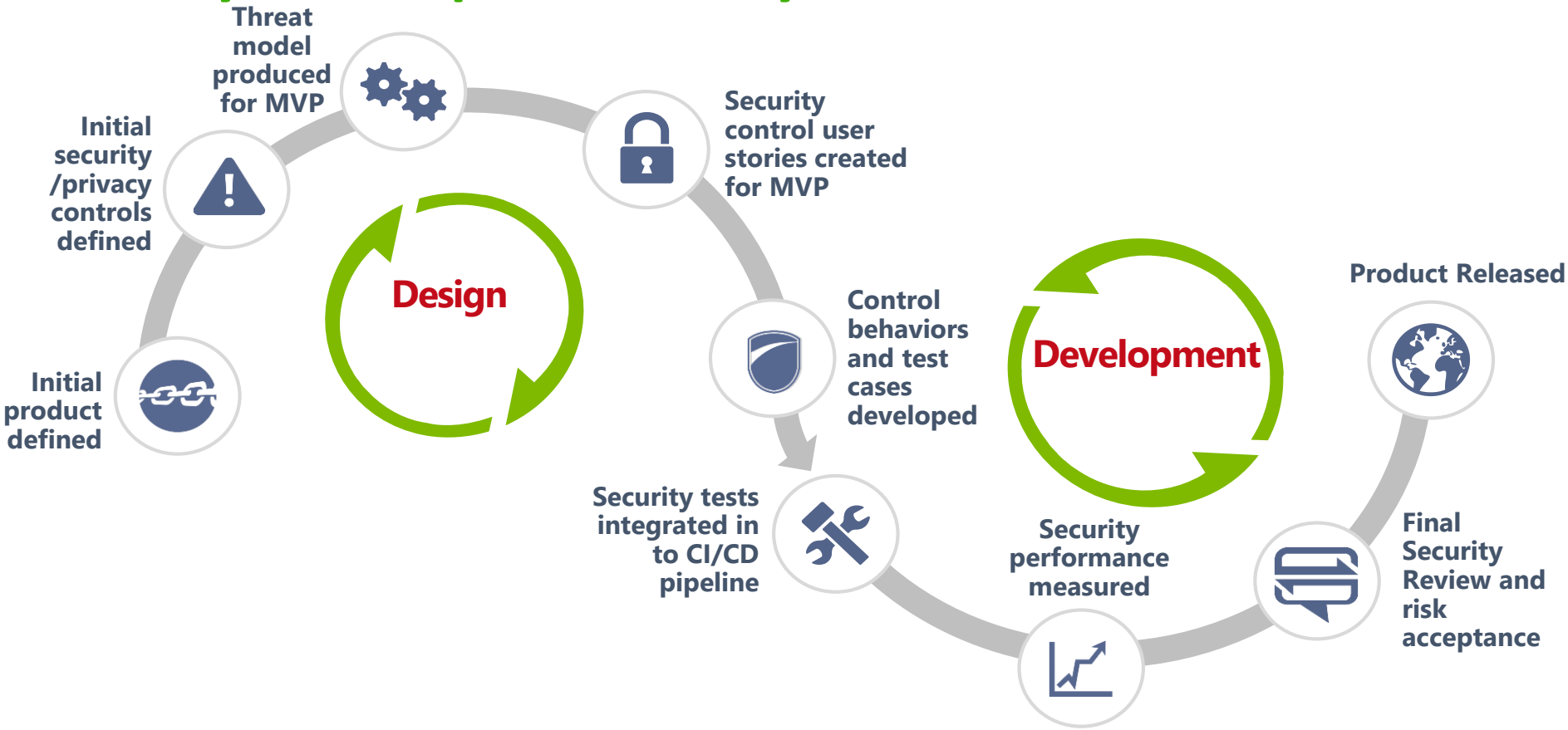


“The Goal”

Eliyahu Goldratt

Lookout

Security Development Lifecycle



Apply What You Have Learned

- Next week you should:
  - Determine what kind of company you are – did you start out as a legacy company who's moving to internet scale or building that way from the beginning? Where are you in the transition?
- In the first three months following this presentation you should:
  - Know where the overlaps between enterprise IT and product security concerns are and how those overlaps are likely to cause friction in the future
  - Be driving toward instrumentation as a key security strategy
  - Be able to identify security constraints and how they are slowing down your development initiatives
- Within six months you should:
  - Be able to measure concrete areas where you exploited constraints to speed up your product development and security process

RSAConference2016

San Francisco | February 29 – March 4 | Moscone Center

Connect to Protect

SESSION ID:

Questions?

Michael Murray

#RSAC