



Continuous Authorization With DevSecOps

Hasan Yasar

Technical Manager, Adjunct Faculty, SEI | CMU

Copyright 2018 Carnegie Mellon University. All Rights Reserved.

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8702-15-D-0002 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

NO WARRANTY. THIS CARNEGIE MELLON UNIVERSITY AND SOFTWARE ENGINEERING INSTITUTE MATERIAL IS FURNISHED ON AN "AS-IS" BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY, OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT.

[DISTRIBUTION STATEMENT A] This material has been approved for public release and unlimited distribution. Please see Copyright notice for non-US Government use and distribution.

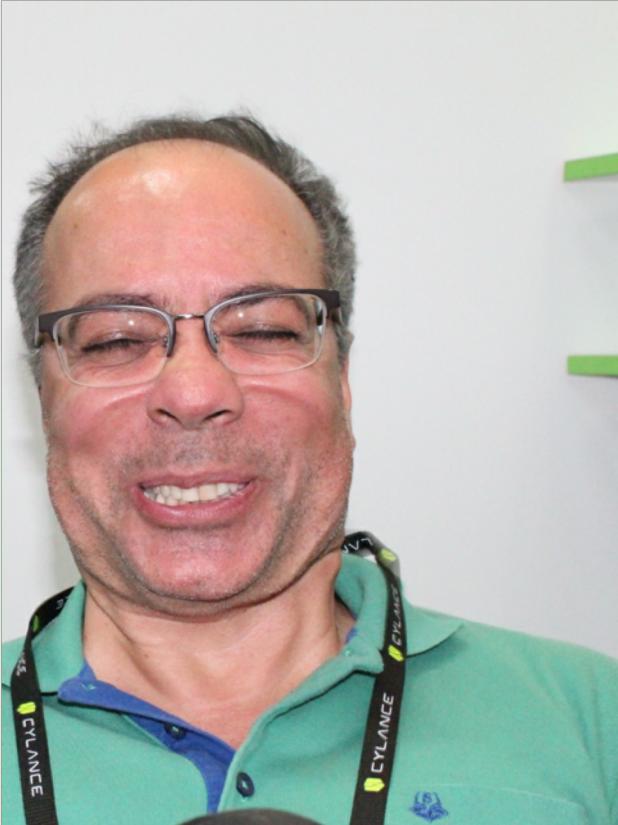
This material may be reproduced in its entirety, without modification, and freely distributed in written or electronic form without requesting formal permission. Permission is required for any other use. Requests for permission should be directed to the Software Engineering Institute at permission@sei.cmu.edu.

Carnegie Mellon® and CERT® are registered in the U.S. Patent and Trademark Office by Carnegie Mellon University.

DM18-1207

With the speed of DevOps...

It is me!
I felt the speed of
DevOps



Release securely
with
DevSecOps

Continuous Authorization with DevSecOps

Why DevOps?

DevOps ?

DevOps is a set of principles and practices emphasizing collaboration and communication between software development teams and IT operations staff along with acquirers, suppliers, and other stakeholders in the lifecycle of a software system¹

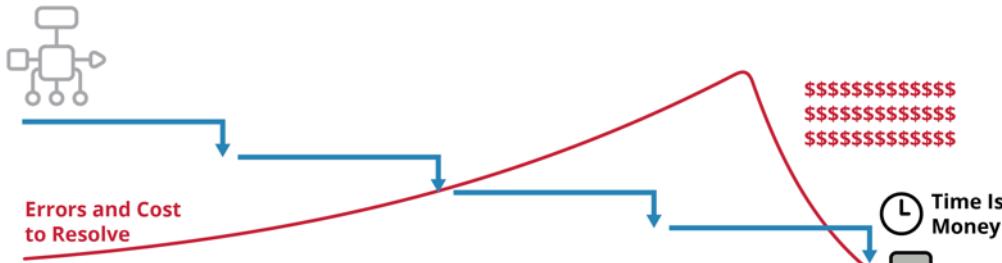
Four Fundamental Principles

1. *Collaboration*: between all stakeholders
2. *Infrastructure as code (IaC)*: assets are versioned, scripted, and shared
3. *Automation*: deployment, testing, provisioning, any manual or human-error-prone process
4. *Monitoring*: any metric in development or operation that can inform priorities, direction, and policy

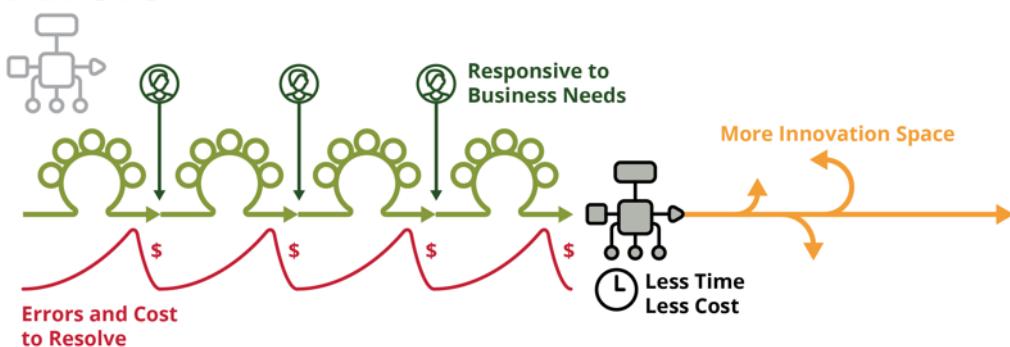
[1] IEEE P2675 DevOps Standard for Building Reliable and Secure Systems Including Application Build, Package, and Deployment

Benefits of DevOps

WATERFALL

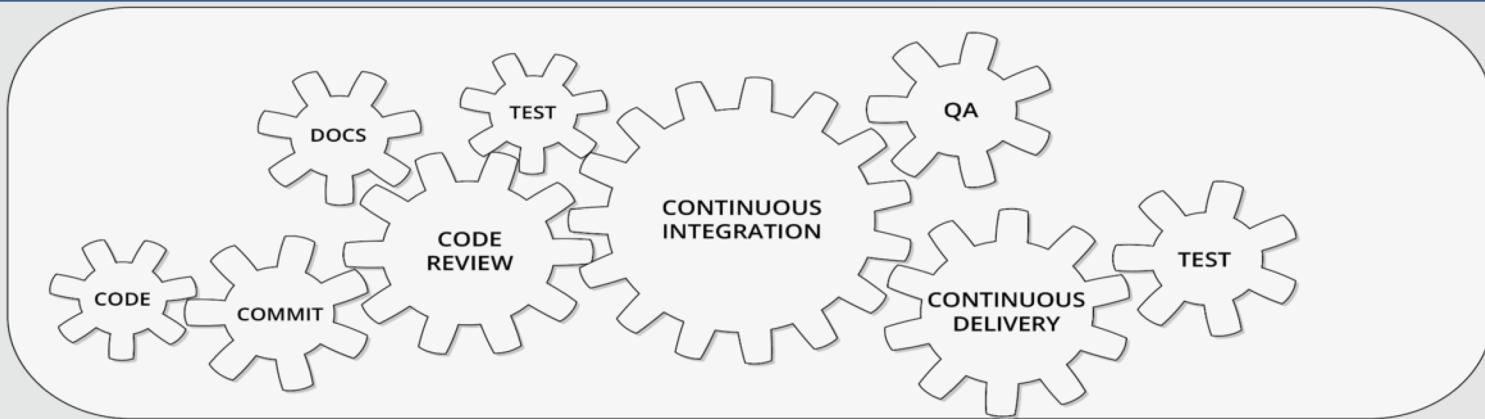


DEVOPS



- Reduced errors during deployment
- Reduced time to deploy and resolve discovered errors
- **Repeatable** steps
- **Continuous availability** of pipeline and application
- Increased innovation time
- **Responsiveness** to business needs
- **Traceability** throughout the application lifecycle
- Increased stability and quality
- **Continuous feedback**

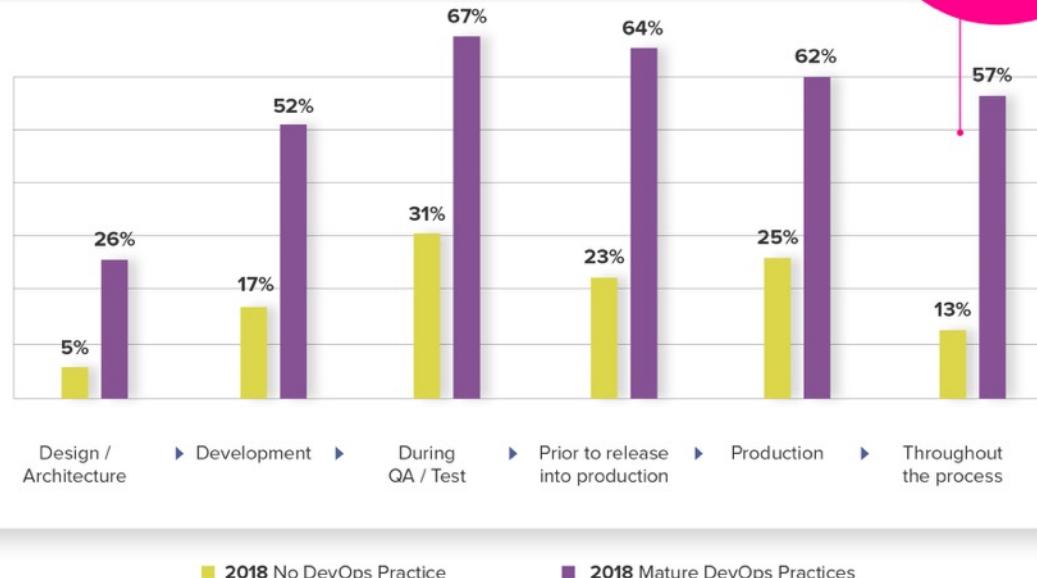
Security Requires Automation with IaC, CI, and CD



- Security requirements and traceability
 - Risk Management Framework: (1) categorize, (2) select controls, (3) implement, (4) assess, (5) authorize, (6) monitor
- Code review and static analysis
- Automated security testing and verification
- Automated dependency vulnerability analysis
- Immutable system, infrastructure (re-)provisioning

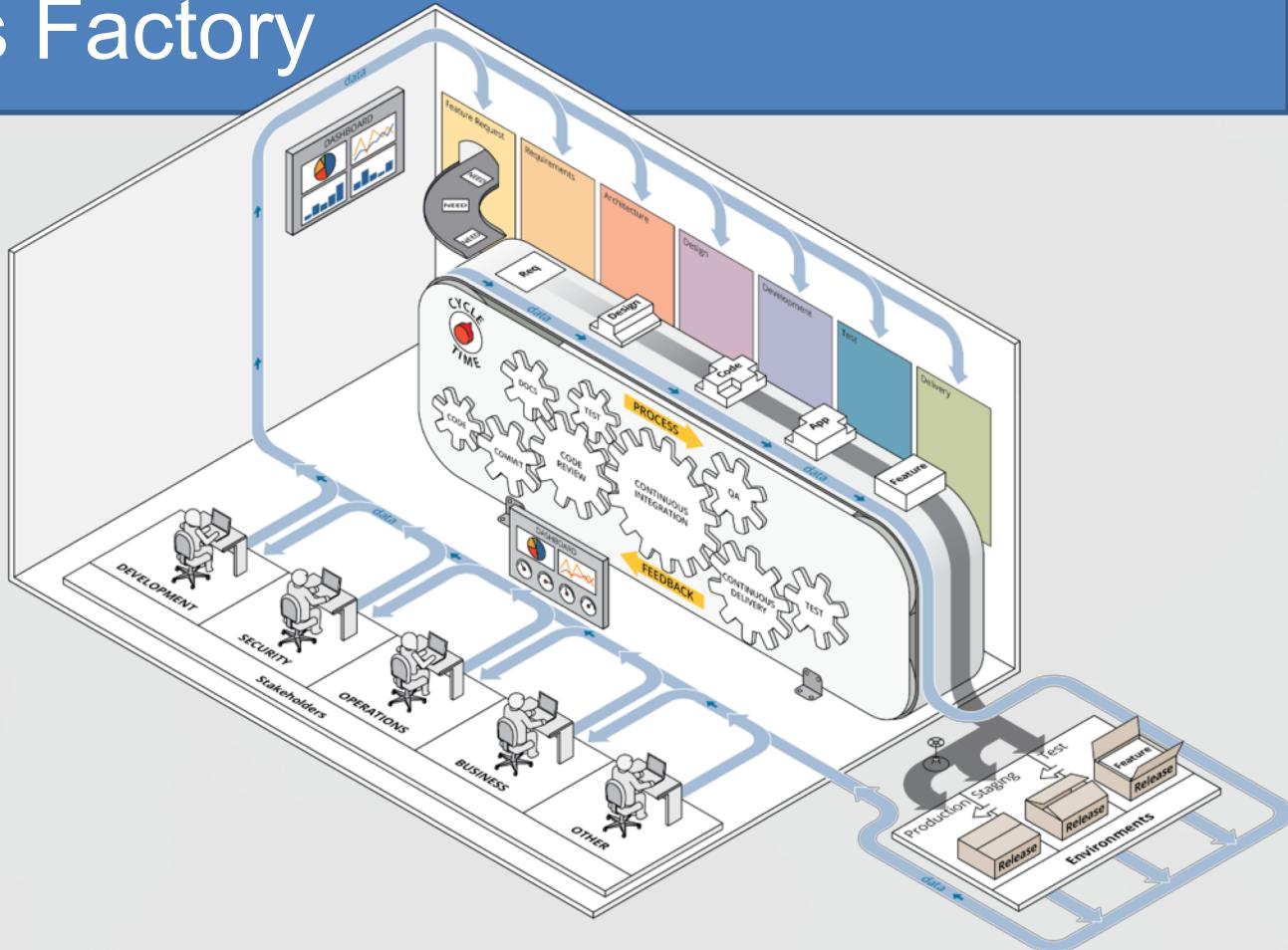
At what point in the development process does your organization perform automated application security analysis?

Mature DevOps practices are 338% more likely to integrate automated security.



The DevOps Factory

- Feature to deployment
- Iterative and incremental development
- Automation in every phase of the SDLC
- Continuous feedback
- Metrics and measurement
- Complete engagement with all stakeholders
- Transparency and traceability across the lifecycle



Continuous Authorization with DevSecOps

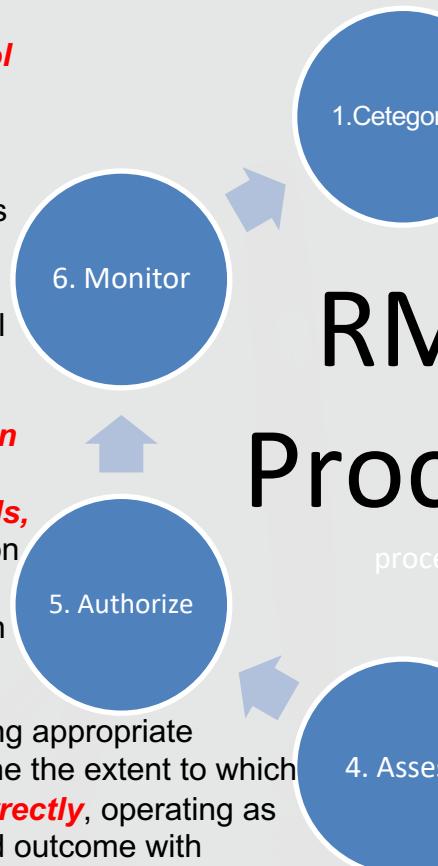
RMF, ATO &Compliances requirements

What is Risk Management Framework (RMF)?

- Information security framework for the entire federal government that replaces legacy Certification and Accreditation (C&A) Processes applied to information systems
- RMF is a key component of an organization's information security program used in the overall management of organizational risk
- NIST Special Publication 800-37, "Guide for Applying the Risk Management Framework to Federal Information Systems", transforms the traditional Certification and Accreditation(C&A) process into the six-step Risk Management Framework (RMF).
- The Risk Management Framework (RMF) provides a disciplined and structured process that integrates information security and risk management activities into the system development lifecycle

RMF Process

process



1. **Categorize** the information system and the **information processed, stored, and transmitted** by that system based on an impact analysis
2. **Select** an initial set of baseline security controls for the information system based on the security categorization; tailoring and supplementing the security control baseline as needed based on an **organizational assessment of risk and local conditions**.
3. **Implement** the security controls and describe how **the controls are employed within the information system and its environment of operation**.

6. **Monitor** the security controls in the information system on an **ongoing basis including assessing control effectiveness, documenting changes to the system or its environment of operation**,

conducting security impact analyses of the associated changes, and reporting the security state of the system to designated organizational officials.

5. **Authorize** information system operation based on a **determination of the risk to organizational operations and assets, individuals, other organizations**, and the Nation resulting from the operation of the information system and the decision that this risk is acceptable.

4. **Assess** the security controls using appropriate assessment procedures to determine the extent to which the **controls are implemented correctly**, operating as intended, and producing the desired outcome with respect to meeting the security requirements for the system

RMF characteristics – NIST 800-37

- Promotes the concept of near real-time risk management and ongoing information system authorization through the implementation of robust ***continuous monitoring processes***;
- Encourages the use of ***automation*** to provide senior leaders the necessary information to make cost-effective, risk-based decisions with regard to the organizational information systems supporting their core missions and business functions;
- Integrates information security into the enterprise architecture and ***system development life cycle***;
- Provides emphasis on the selection, implementation, assessment, and ***monitoring*** of security controls, and the authorization of information systems;
- Links risk management processes at ***the information system level*** to risk management processes at the ***organization level*** through a risk executive (function); and
- Establishes ***responsibility*** and ***accountability*** for security controls deployed within organizational information systems and inherited by those systems

Authorization with monitoring (NIST 800-137)



Next , Authorization to Operate(ATO)

A typical security controls are about **924** for a selected systems and all are in Excel format;

Control Number	Family	Control Title	Control Text	Confidentiality	Integrity	Availability	Supplemental Guidance
		TOOLS					
SI-4 (10)	SI	INFORMATION SYSTEM MONITORING VISIBILITY OF ENCRYPTED COMMUNICATIONS	The organization makes provisions so that [Assignment: organization-defined encrypted communications traffic] is visible to [Assignment: organization-defined information system monitoring tools].	High Moderate	High Moderate	High Moderate	Supplemental Guidance: Organizations balance the potentially conflicting needs for encrypting communications traffic and for having insight into such traffic from a monitoring perspective. For some organizations, the need to ensure the confidentiality of communications traffic is paramount; for others, mission-assurance is of greater concern. Organizations determine whether the visibility requirement applies to internal encrypted traffic, encrypted traffic intended for external destinations, or a subset of the traffic types.
SI-4 (11)	SI	INFORMATION SYSTEM MONITORING ANALYZE COMMUNICATIONS TRAFFIC ANOMALIES	The organization analyzes outbound communications traffic at the external boundary of the information system and selected [Assignment: organization-defined interior points within the system (e.g., subnetworks, subsystems)] to discover anomalies.	High Moderate Low	High Moderate Low	High Moderate Low	Supplemental Guidance: Anomalies within organizational information systems include, for example, large file transfers, long-time persistent connections, unusual protocols and ports in use, and attempted communications with suspected malicious external addresses.
SI-4 (12)	SI	INFORMATION SYSTEM MONITORING AUTOMATED ALERTS	The organization employs automated mechanisms to alert security personnel of the following inappropriate or unusual activities with security implications: [Assignment: organization-defined activities that trigger alerts].	High Moderate Low	High Moderate Low	High Moderate Low	Supplemental Guidance: This control enhancement focuses on the security alerts generated by organizations and transmitted using automated means. In contrast to the alerts generated by information systems in SI-4 (5), which tend to focus on information sources internal to the systems (e.g., audit records), the sources of information for this enhancement can include other entities as well (e.g., suspicious activity reports, reports on potential insider threats). Related controls: AC-18, IA-3.
SI-4 (13)	SI	INFORMATION SYSTEM MONITORING ANALYZE TRAFFIC / EVENT PATTERNS	The organization: (a) Analyzes communications traffic/event patterns for the information system; (b) Develops profiles representing common traffic patterns and/or events; and (c) Uses the traffic/event profiles in tuning system-monitoring devices to reduce the number of false positives and the number of false negatives.				

Compliance, Legal Requirements

- There are many compliances and legal requirements

GDPR: General Data Protection Regulation

FISMA :Federal Information Security Management

SOX : Sarbanes–Oxley

HIPAA : Health Insurance Portability and Accountability

PCI DSS: Payment Card Industry Data Security Standard

NIST :National Institute of Standards and Technology,
And many more..

- All requires
 - Reporting,
 - Auditing
 - Traceability

❖ Volkswagen emission case: SW Engineer 40 months in prison, VW \$20Billion in fines

<https://www.nytimes.com/2017/08/25/business/volkswagen-engineer-prison-diesel-cheating.html>

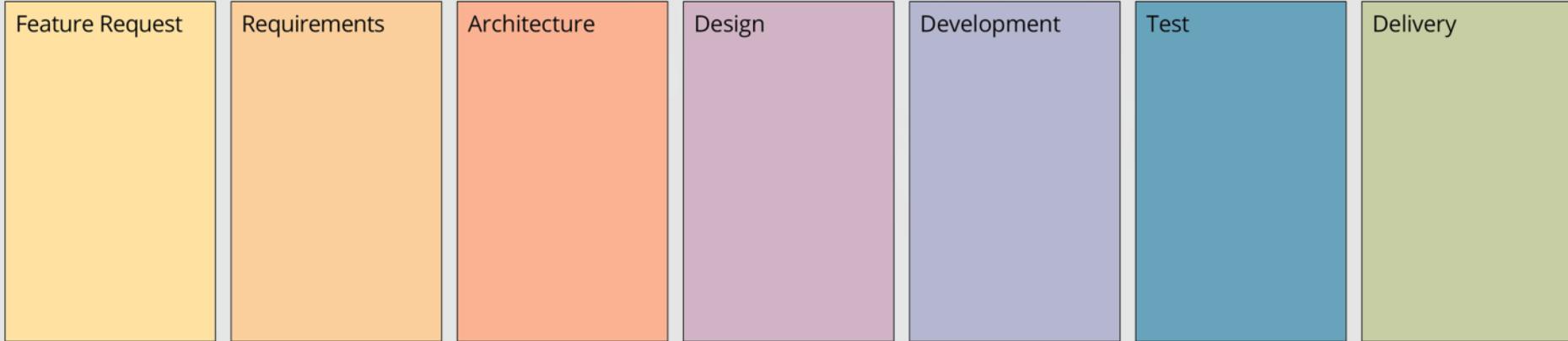
Continuous Authorization with DevSecOps

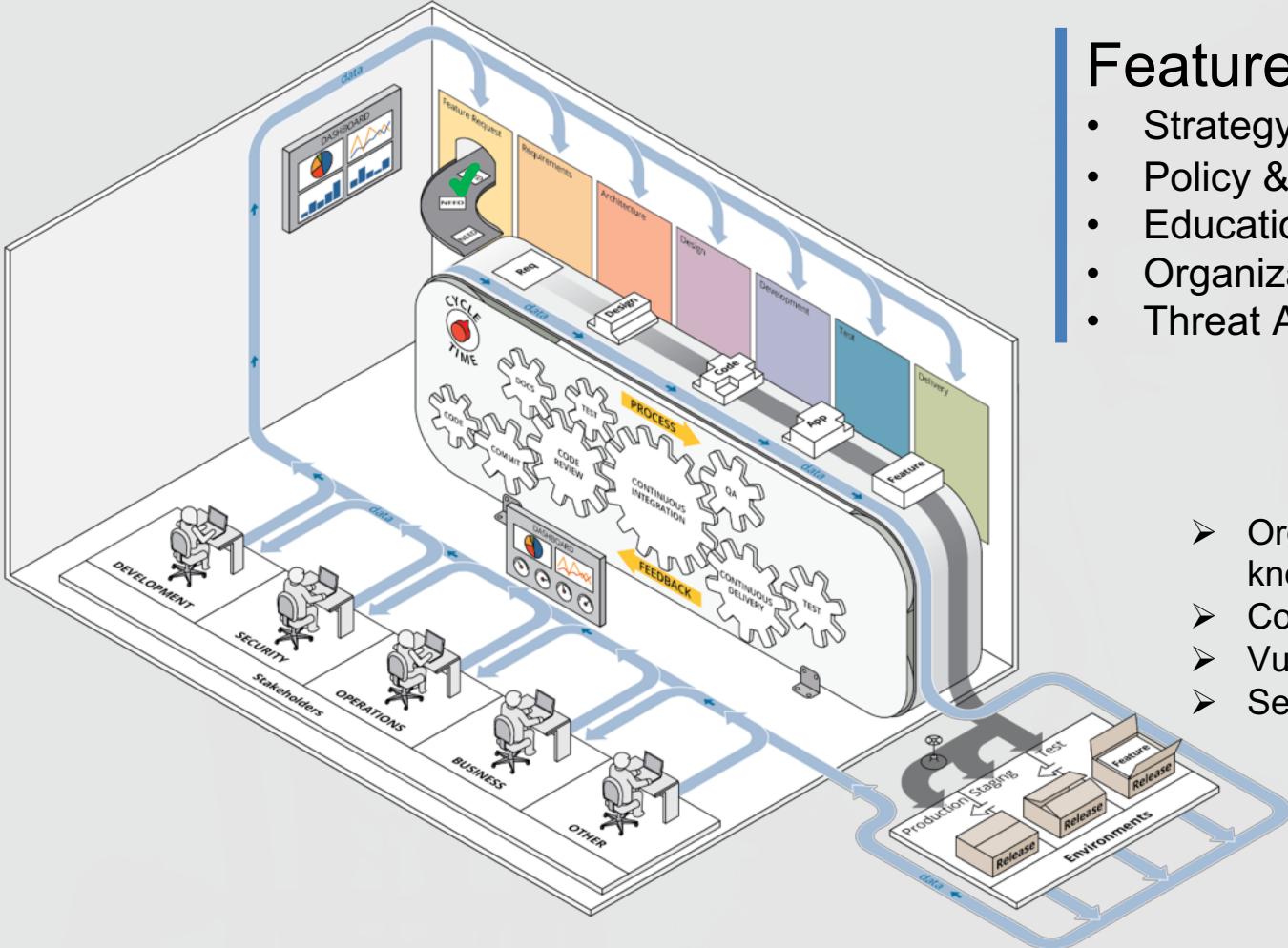
How to do in DevSecOps?

What is DevSecOps?

DevSecOps is a model on integrating the software development and operational process considering security activities: requirements, design, coding, testing , delivery , deployment and incident response.

DevOps Phases – *on each iteration/sprint*



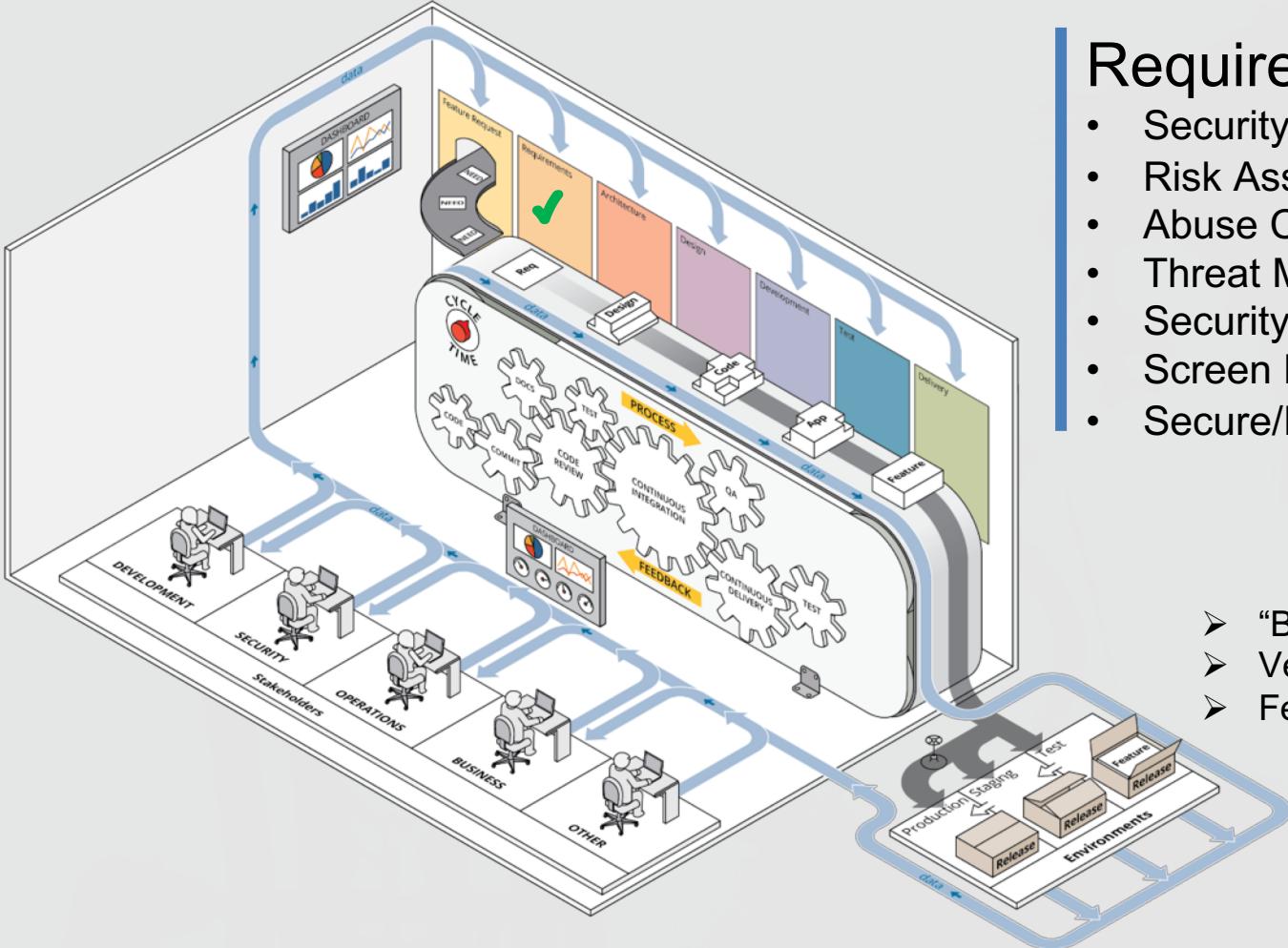


Feature Request

- Strategy & Metrics
- Policy & Governance
- Education & Security Guidance
- Organizational Risk Factors
- Threat Assessment



- Organizational awareness and knowledge
- Common attack vectors
- Vulnerability management
- Security Development Plan



Requirements

- Security Requirements (SFR/SAR)
- Risk Assessment
- Abuse Case Development
- Threat Modelling
- Security Stories
- Screen Development Tools
- Secure/Hardened Environments



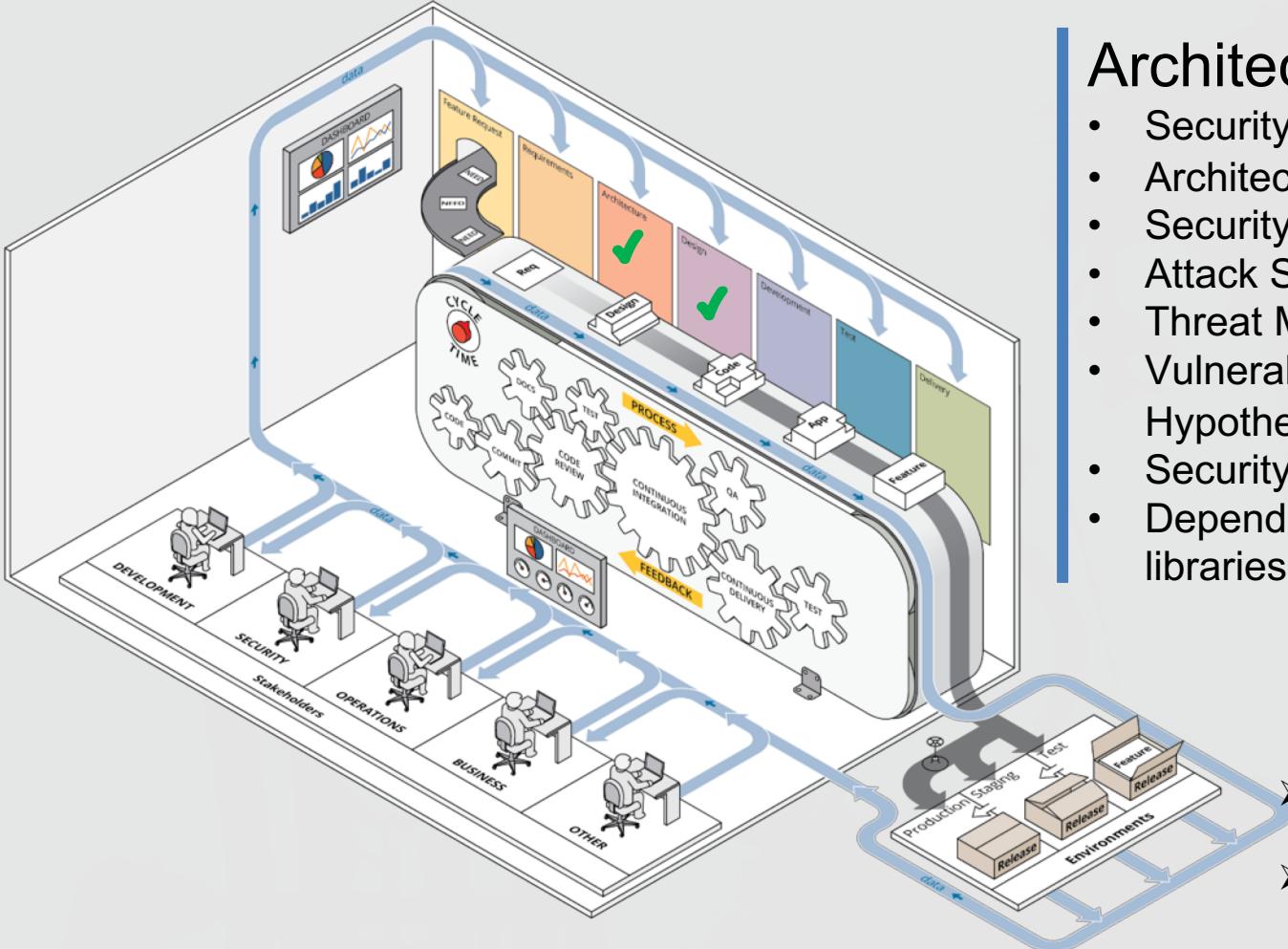
- “Baked in” Security Thoughts
- Verify Security Requirements
- Feature based security controls

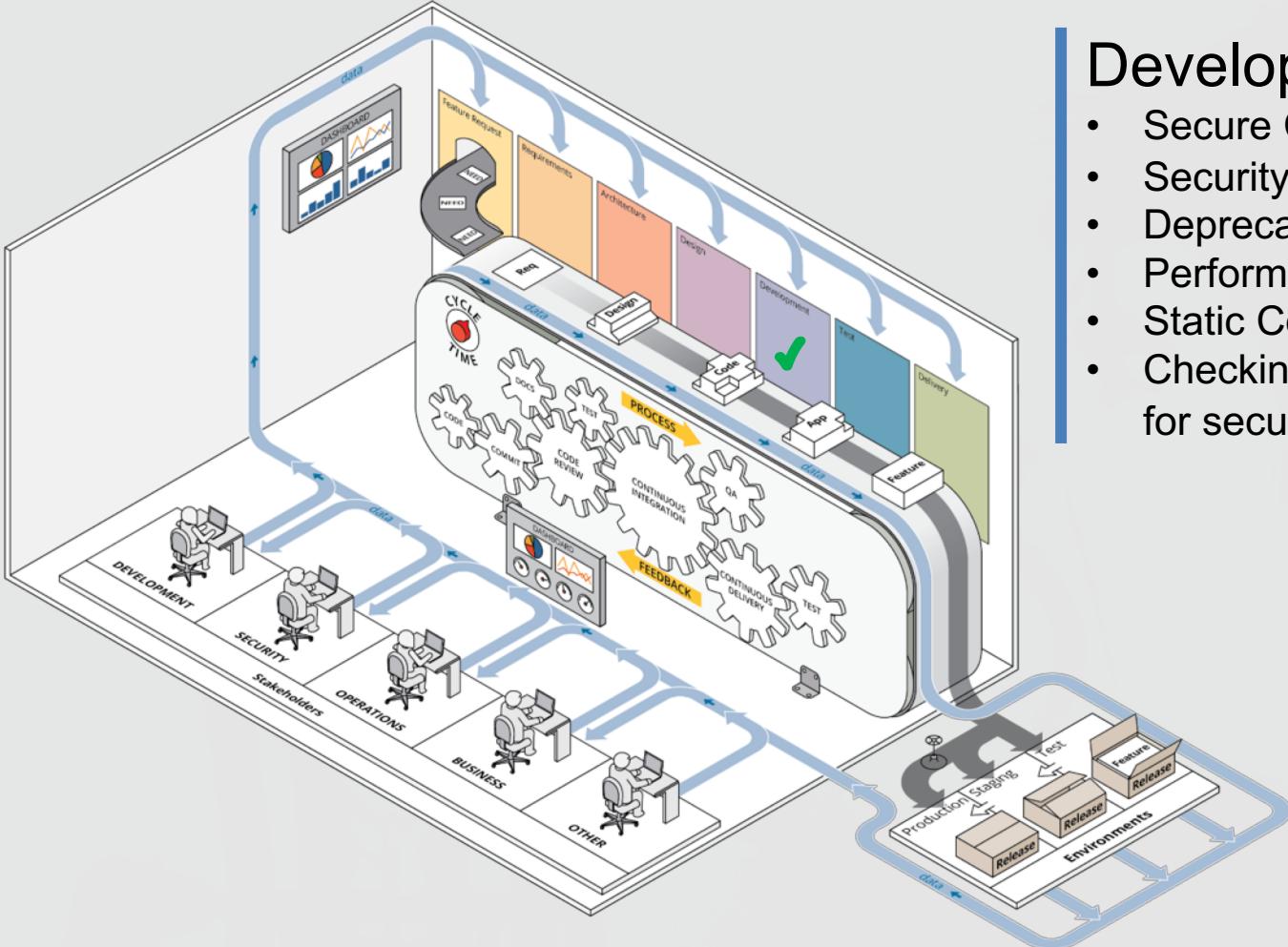
Architecture & Design

- Security Architecture
- Architectural Risk Analysis
- Security Design Requirements
- Attack Surface Analysis
- Threat Modelling
- Vulnerability Analysis and Flow Hypothesis
- Security Design Review
- Dependencies List, Open-source libraries



- Verify and Validate Security Design
- Personnel data- privacy



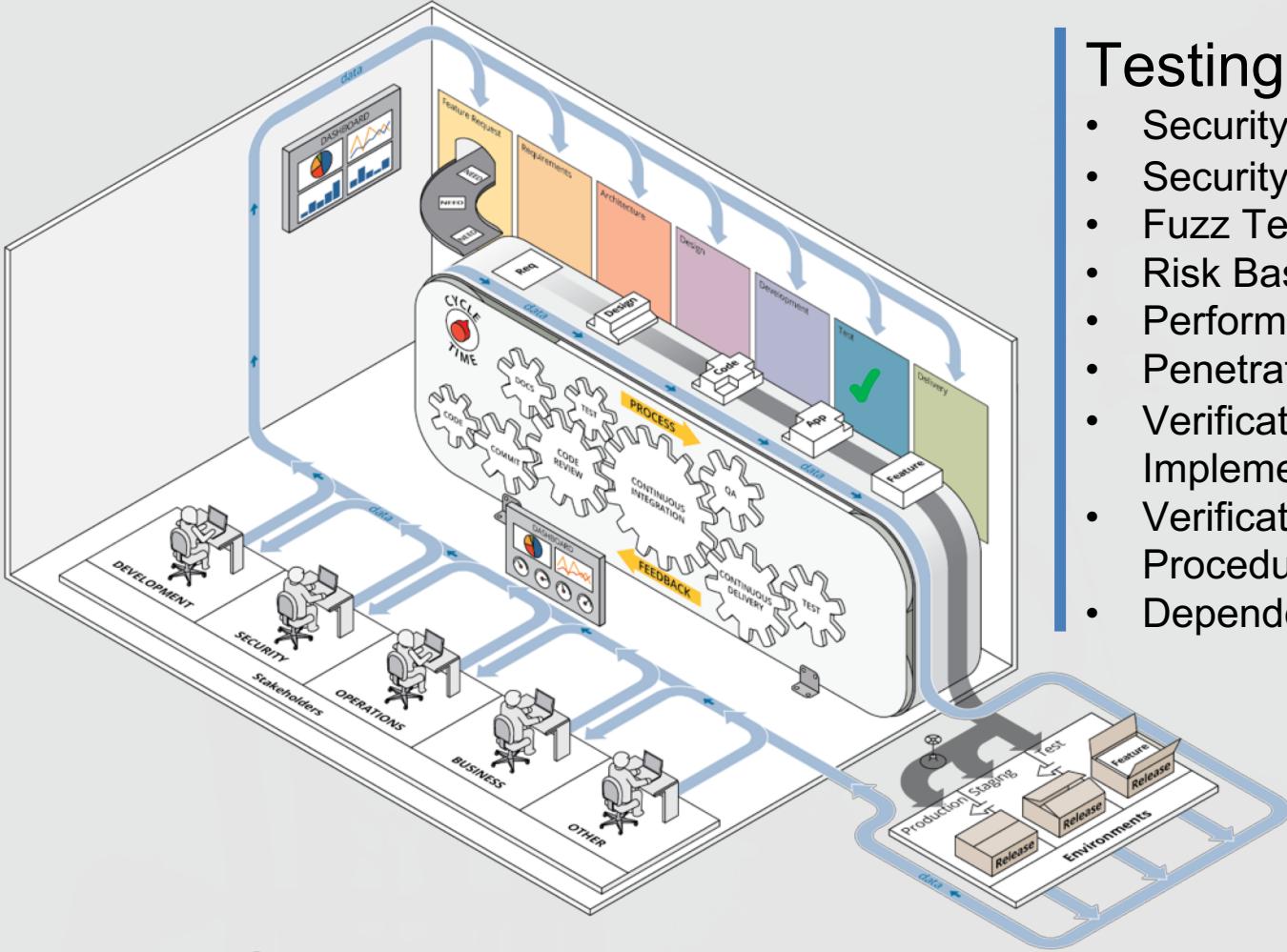


Development

- Secure Coding Practices
- Security Focused Code Review
- Deprecate Unsafe Functions
- Perform Security Unit Testing
- Static Code Analysis
- Checking of process and procedures for secure coding & traceability



- Code Development Audit
- Unit Testing result
- Static Code Analysis results
- Code verification and validation on security practices
- Design validation

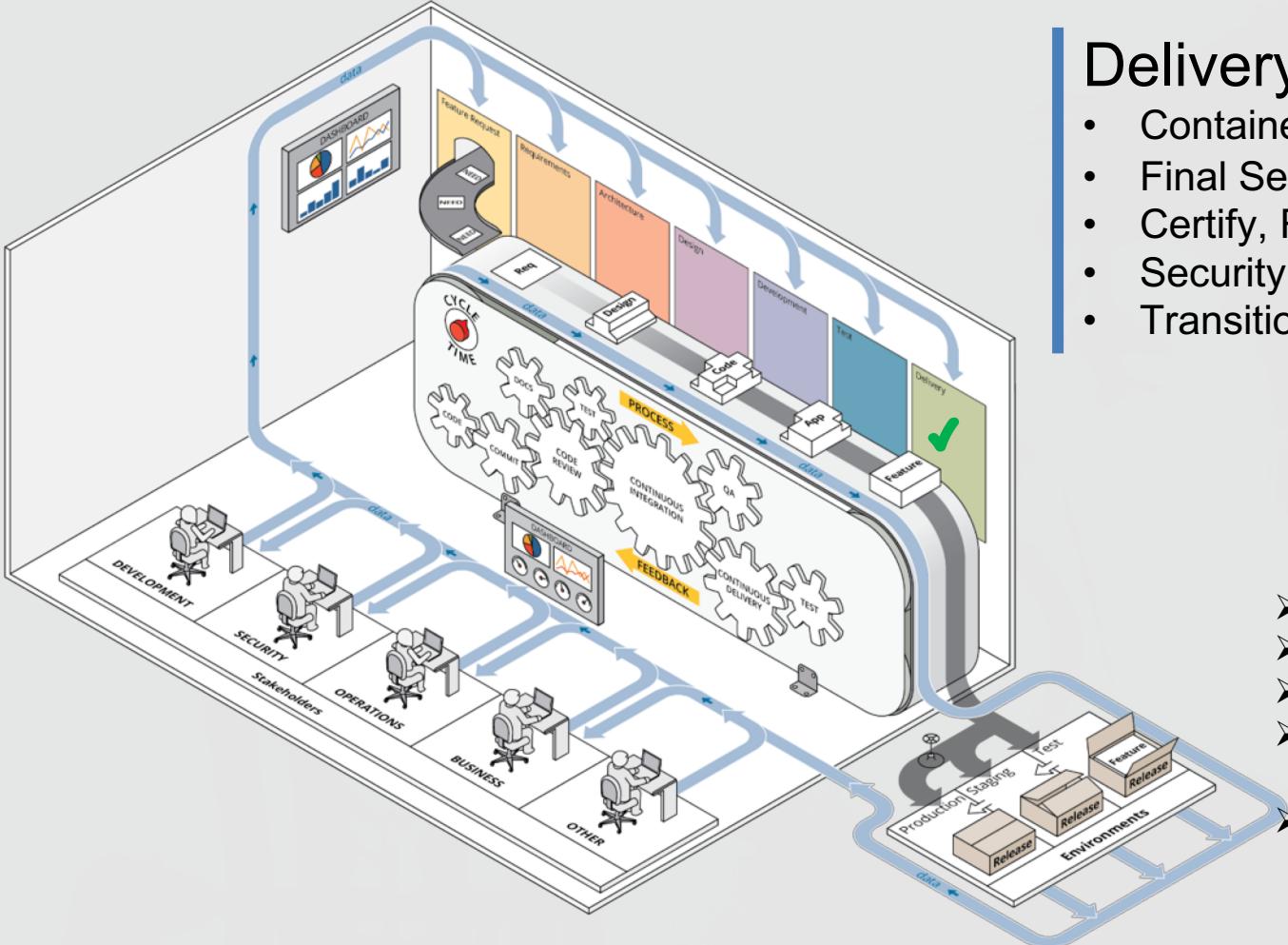


Testing

- Security Test Planning
- Security Testing
- Fuzz Testing
- Risk Based Security Testing
- Perform Dynamic Analysis
- Penetration Testing
- Verification of Security Implementation
- Verification of Process and Procedures
- Dependency Monitoring



- Test results,
- Data handling varication
- Validation of security features



Delivery

- Container Security
- Final Security Review
- Certify, Release and Archive
- Security Acceptance Testing
- Transition Incident Response Plan



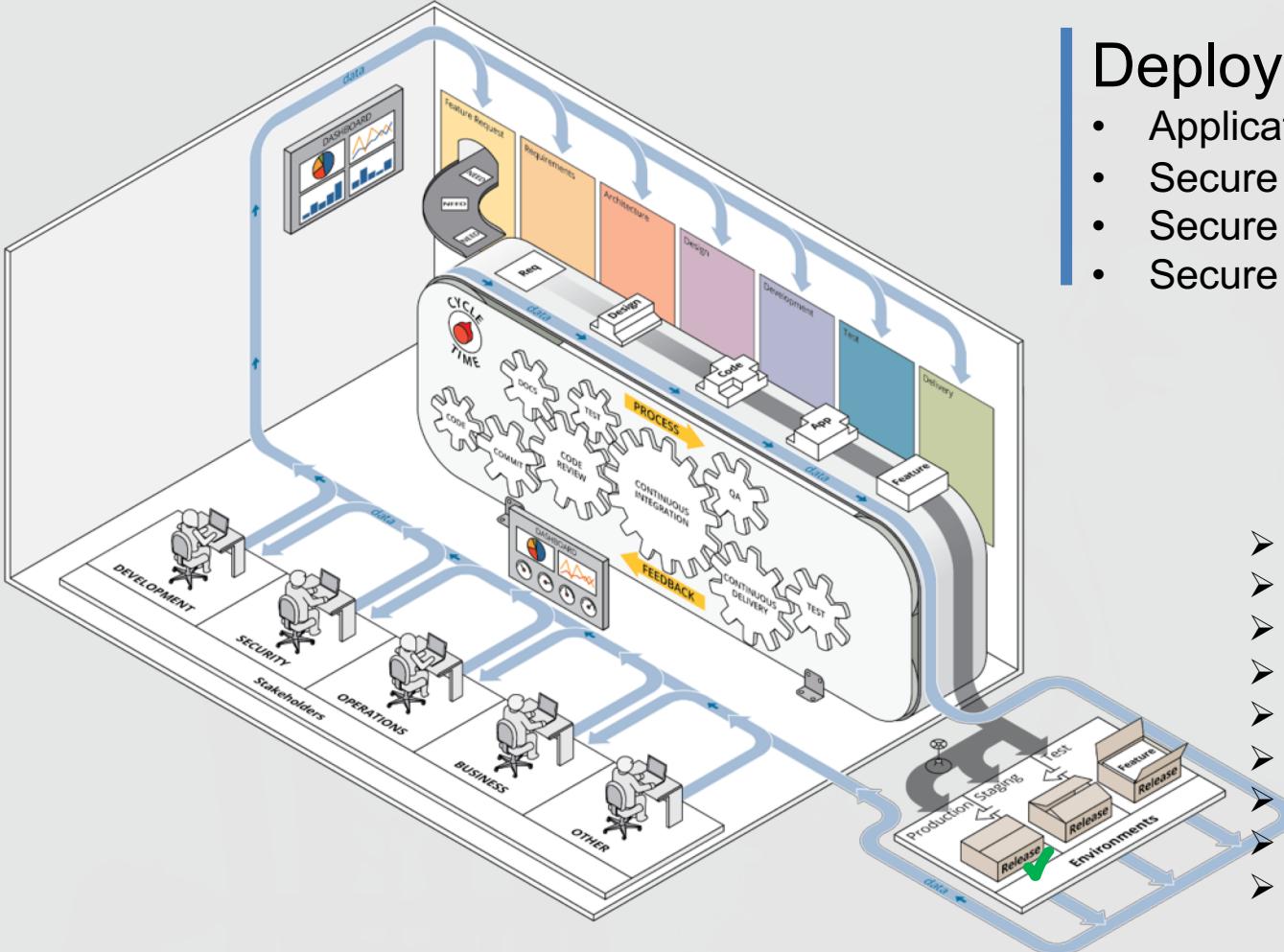
- Pre-approval
- Dependency checks
- Validate incident response
- Audit data access /rights /contents
- Environment verification

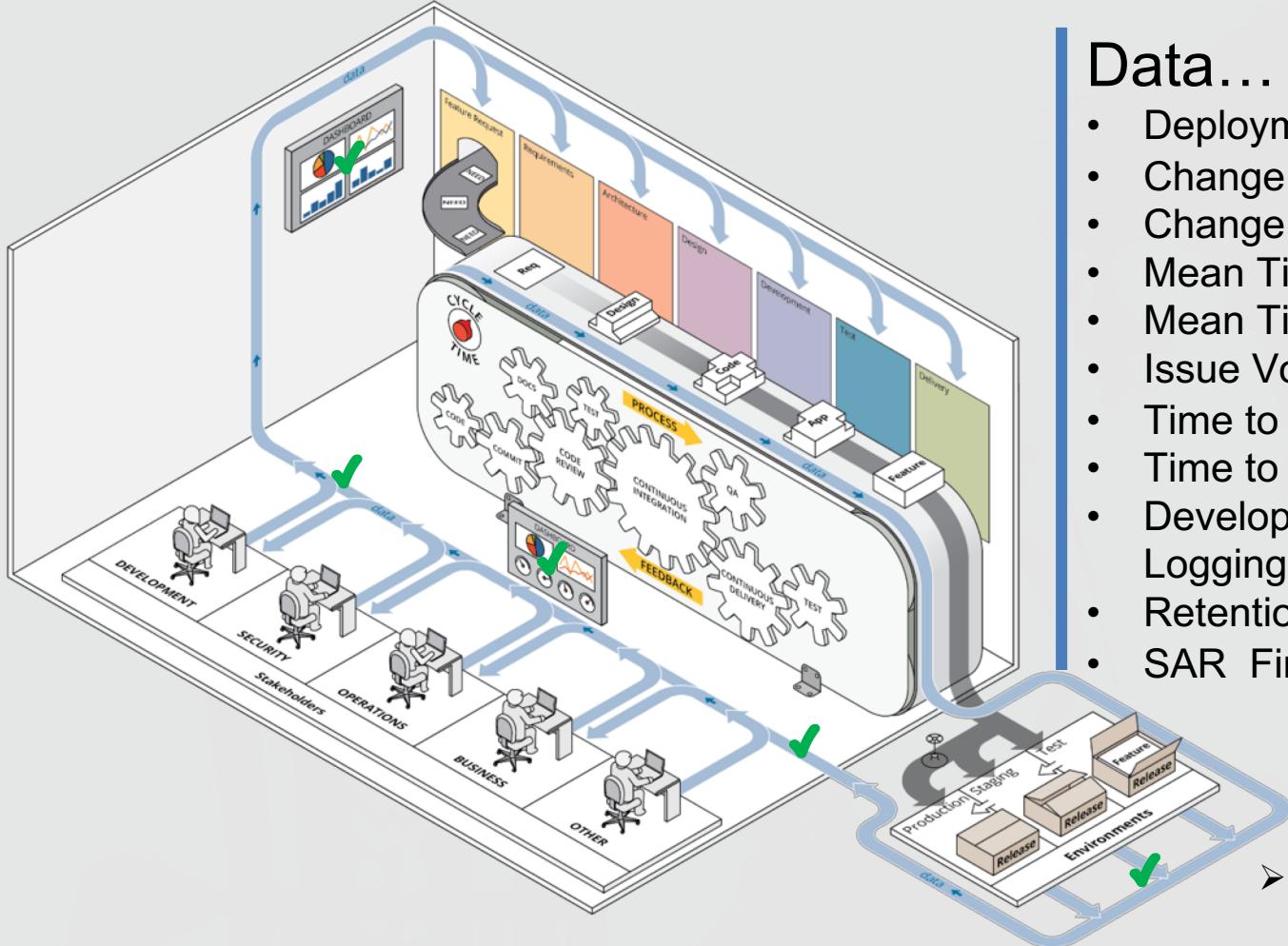
Deploy

- Application Security Monitoring
- Secure Deployment Process
- Secure Environment
- Secure Operational Enablement



- Security Dashboard
- Security Status
- Incident Response
- Rollback capabilities
- Application /Environments logs
- IDS/IPS logs
- Environment monitoring
- Resource usage
- Data handling process



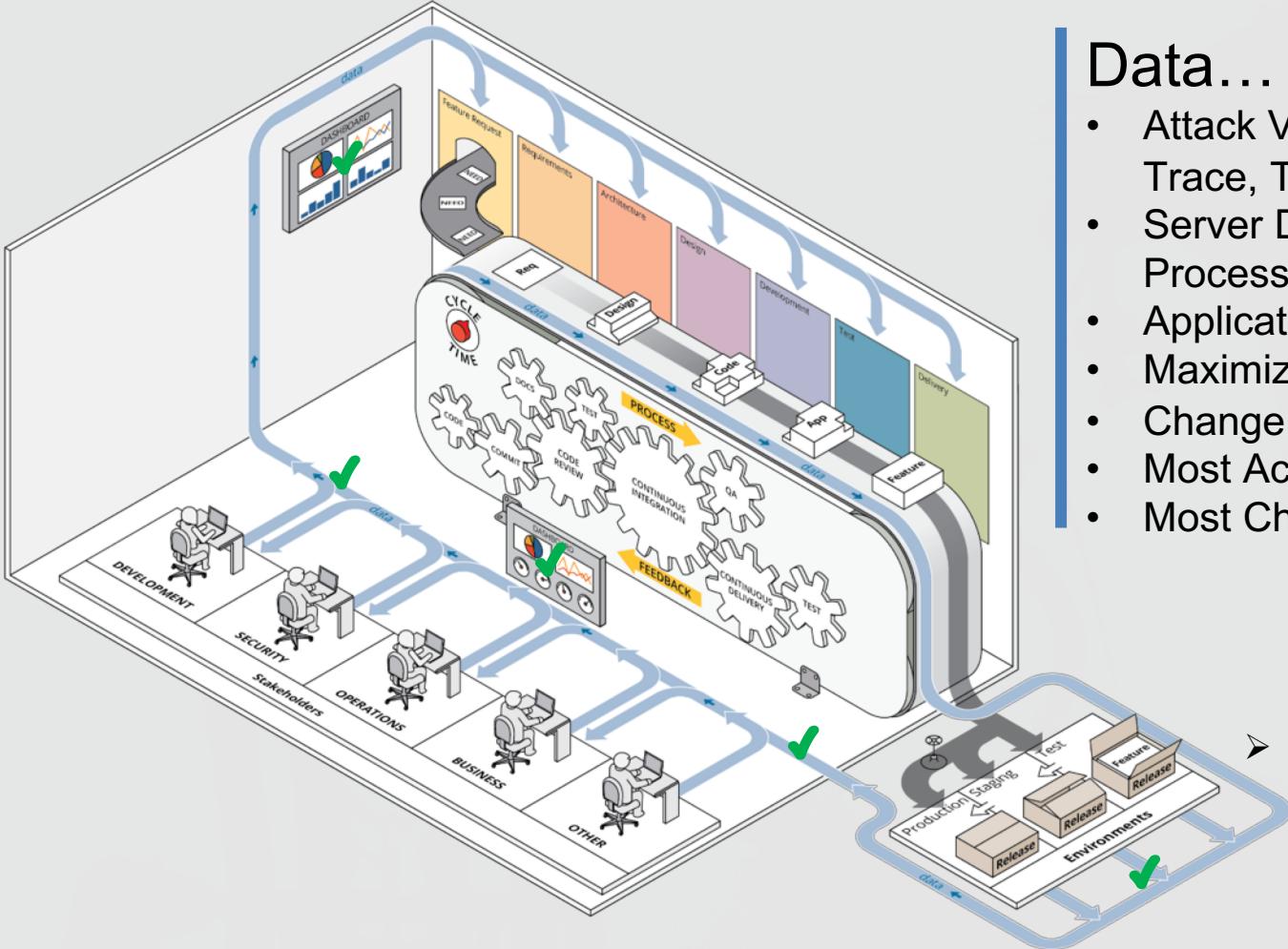


Data...

- Deployment Frequency
- Change Lead Time and Volume
- Change Failure Rate
- Mean Time To Recovery (MTTR)
- Mean Time to Detection (MTTD)
- Issue Volume and Resolution Time
- Time to Approval
- Time to Patch Vulnerabilities
- Development and Application Logging Availability
- Retention Control Compliance
- SAR Findings



➤ Continuous Monitoring to feed Continuous Authorization



Data...

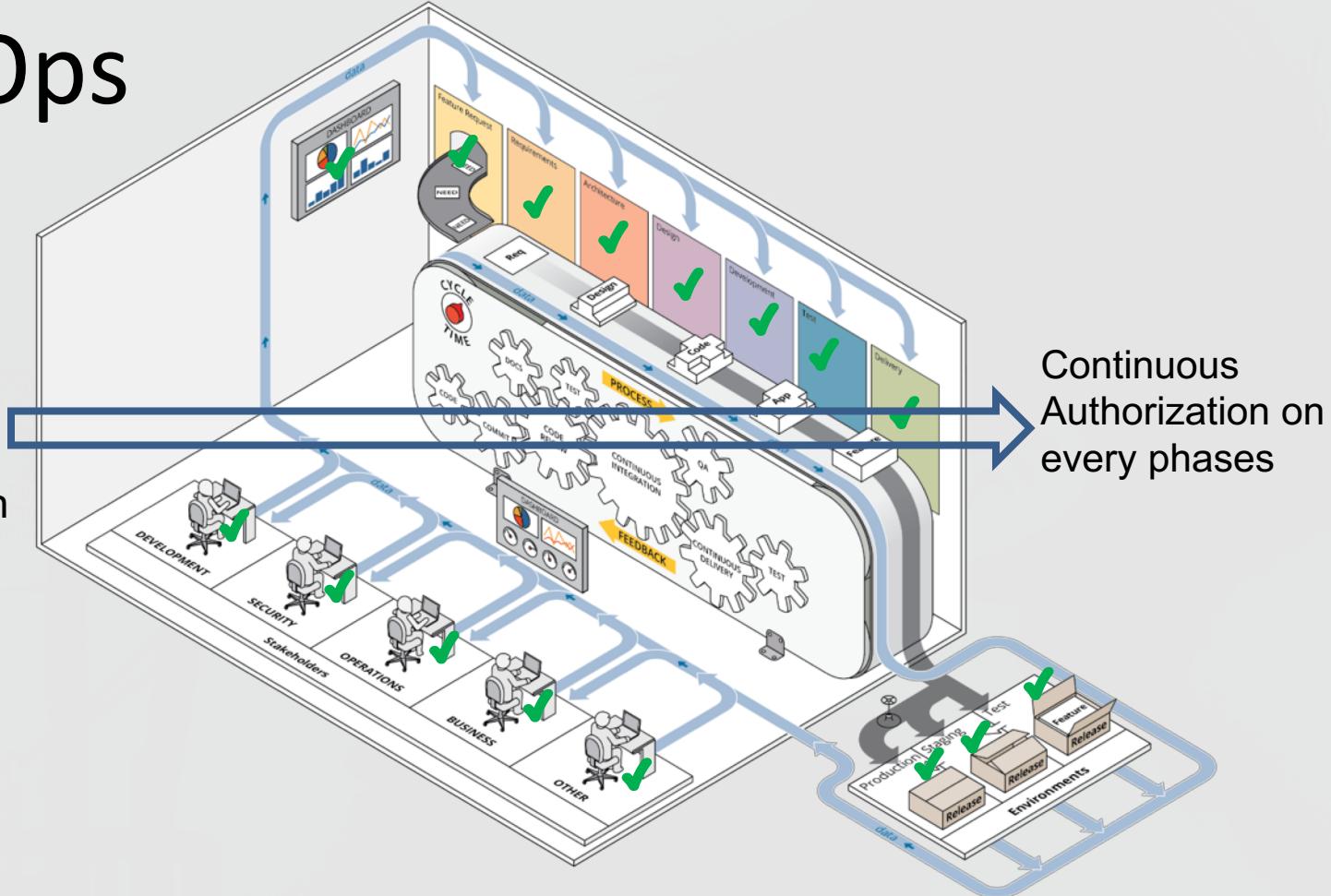
- Attack Vector Details (IP, Stack Trace, Time, Rate of Attack, etc)
- Server Disk Space, Load and Process Monitoring
- Application Performance
- Maximize Monitoring
- Change in Size to Code Base
- Most Active Code Contributors
- Most Changed Code Areas



➤ Continuous Monitoring to feed Continuous Authorization

DevSecOps

Security from inception to deployment and improvement with every delivery



For More Information

DevOps: <https://www.sei.cmu.edu/go/devops>

DevOps Blog: <https://insights.sei.cmu.edu/devops>

Webinars: <https://www.sei.cmu.edu/publications/webinars/index.cfm>

Podcasts: <https://www.sei.cmu.edu/publications/podcasts/index.cfm>

YouTube: <https://www.youtube.com/user/TheSEICMU>

SLS team GitHub Projects

- Once Click DevOps deployment
<https://github.com/SLS-ALL/devops-microcosm>
- Sample app with DevOps Process
https://github.com/SLS-ALL/flask_api_sample
 - Tagged checkpoints
 - v0.1.0: base Flask project
 - v0.2.0: Vagrant development configuration
 - v0.3.0: Test environment and Fabric deployment
 - v0.4.0: Upstart services, external configuration files
 - v0.5.0: Production environment
- On YouTube:
<https://www.youtube.com/watch?v=5nQIJ-FWA5A>

Thank You All Day DevOps Sponsors

Platinum Sponsors



Gold Sponsors



GitLab



GENERAL DYNAMICS
Information Technology



Carnegie
Mellon
University
Software
Engineering
Institute



SCALED AGILE[®]

Media Sponsors



Solutions
Review

Thank You All Day DevOps Supporters



Meet Me in the Slack Channel for Q&A

bit.ly/addo-slack

Let us connect @

Hasan Yasar

**Technical Manager,
Secure Lifecycle Solutions**

hyasar@sei.cmu.edu

[@securelifecycle](https://twitter.com/securelifecycle)

<https://www.linkedin.com/in/hasanyasar>

