.conf18

splunk>

AWS Security Automation and Orchestration

The Foundation of Application Migration and Modernization for Regulated Industry

Tim Sandage | Sr. Security Partner Strategist

Forward-Looking Statements

During the course of this presentation, we may make forward-looking statements regarding future events or the expected performance of the company. We caution you that such statements reflect our current expectations and estimates based on factors currently known to us and that actual events or results could differ materially. For important factors that may cause actual results to differ from those contained in our forward-looking statements, please review our filings with the SEC.

The forward-looking statements made in this presentation are being made as of the time and date of its live presentation. If reviewed after its live presentation, this presentation may not contain current or accurate information. We do not assume any obligation to update any forward-looking statements we may make. In addition, any information about our roadmap outlines our general product direction and is subject to change at any time without notice. It is for informational purposes only and shall not be incorporated into any contract or other commitment. Splunk undertakes no obligation either to develop the features or functionality described or to include any such feature or functionality in a future release.

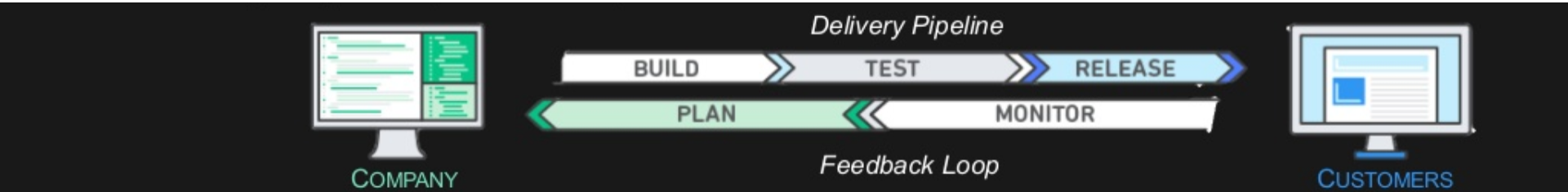
splunk> .conf18

Session Overview

- Security Automation and Orchestration
  - What is Secure DevOps?
  - What happens to our regulated customers today, in adopting DevSecOps models?
  - Introduction to Security Automation and Orchestration.
- Stages of Security Automation and Orchestration Adoption
  - Cloud Migrator
  - Cloud Forward
  - Cloud Native
- Collaborative Social Engineering
  - Review of GitHub as the collaborative hub for effective service transformation on AWS.
  - Review of GitHub at the core of Security Automation and Orchestration

What is DevSecOps?

- Union of software development and operations
- Migration of Agile continuous development into continuous integration, continuous delivery, and continuous compliance.
- DevSecOps Model
  - No Silos – Puts emphasis on communication, collaboration and cohesion between disciplines
  - Best practices for change, configuration, and deployment automation
  - Deliver apps/services at a faster pace
  - High speed product updates
  - Everything is code



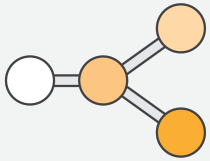
Devsecops Processes: 4 Major Phases

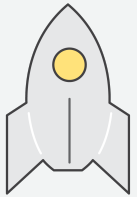
- Check-in source code
- Peer review new code

- Compile code
- Unit tests
- Style checkers
- Code metrics
- Create container images

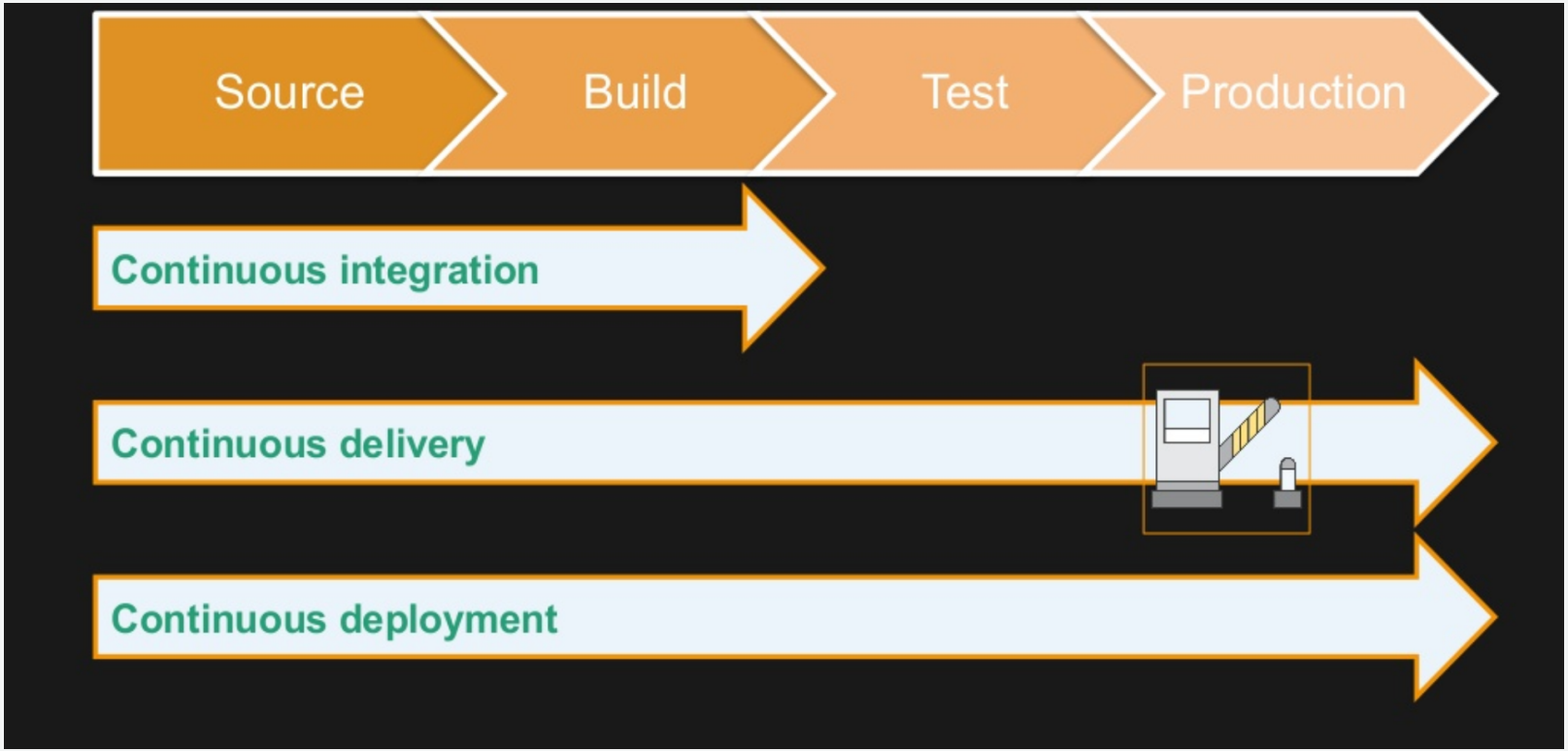
- Integration tests with other systems
- Load testing
- UI tests
- SecOps Scanning

- Deployment to production environments
- Continuous Monitoring





DevSecOps Release Processes:



Problem Statement – Why Can't we be Agile?

Security and risk management leaders continue to labor over **"How"** do they secure current, legacy and cloud resources consistently within their limited constraints.

While cloud services has provided streamlined ways to achieve innovation through the principles of DevSecOps and Developer Self-Service, regulated customers are still under mandate to follow strict security, governance, and accreditation standards, which are delivered during the production deployment phase.

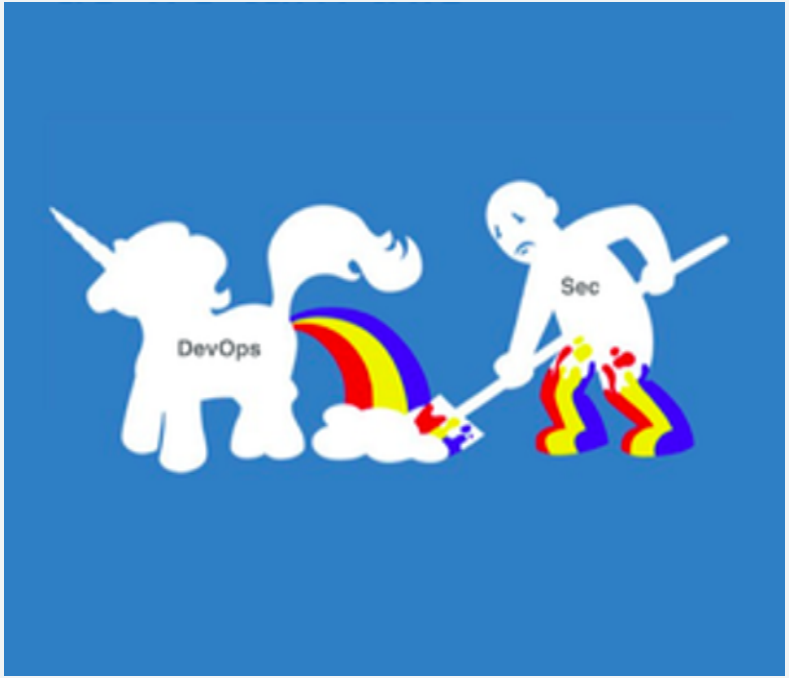
Developer Self-Service – In a Compliance Oriented World

DevOps enables the CI/CD pipeline which is the basis of automation within AWS.

The biggest challenge is breaking out of the traditional security structures and eliminating the divide between developers, operations, and security.

The CI/CD pipeline is the foundation for creating a repeatable, reliable and constantly improving process for taking software from concept to a secure, complaint production solution.

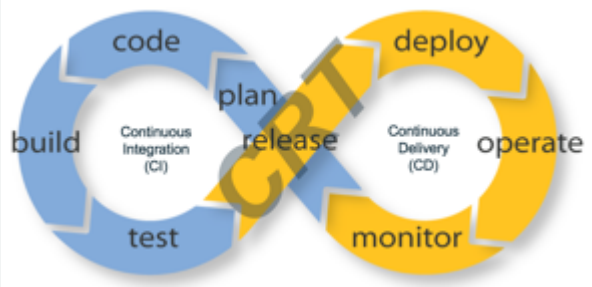
AWSome! But what actually happens in Regulated environments today?



Solution Overview: SAO

Develop an **AWS Security Automation and Orchestration (SAO)** repository for constraining, tracking, publishing continuous security configurations, integration, deployments and treatments which are certified against common security frameworks (e.g. FedRAMP, DoD CC SRG, IRS 1075,CIS, PCI, etc.)

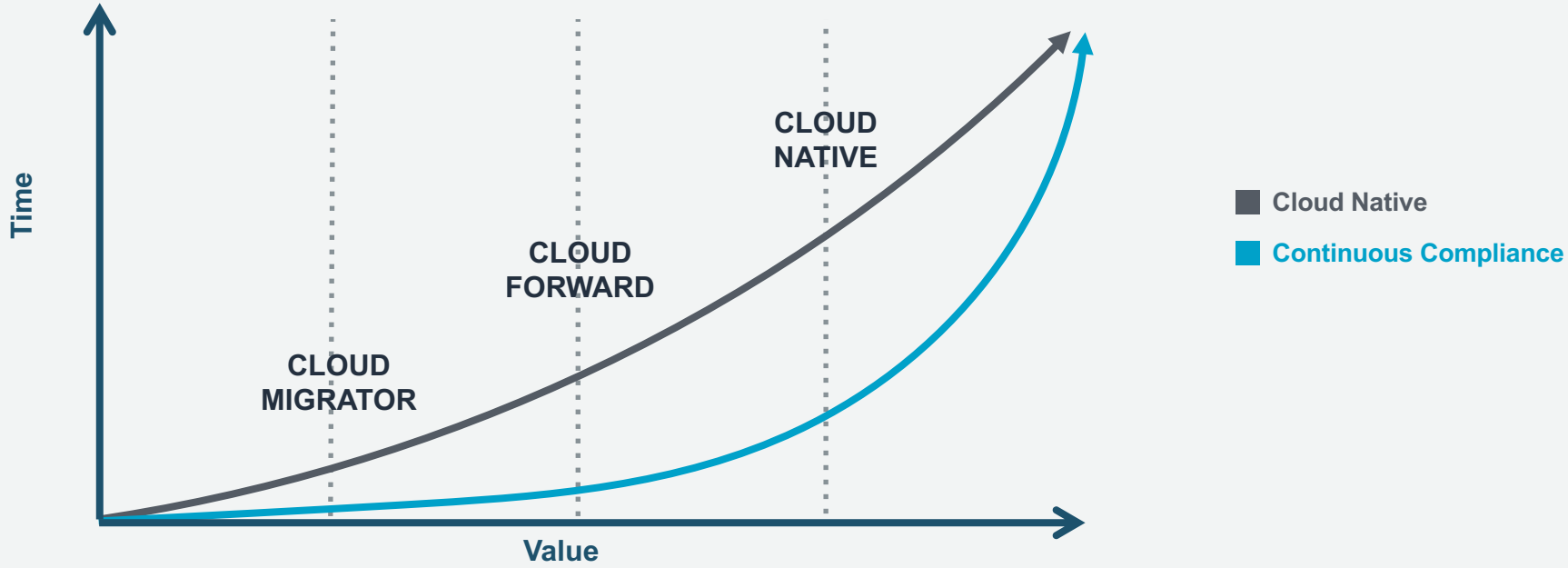
SAO will facilitate the orientation and association of **DevOps** and **Security** practices, changes and coordination of **Continuous Integration (CI)**, **Continuous Delivery (CD)** and **Continuous Risk Treatment (CRT)*** of an AWS customer account and/or multiple accounts.



* CRT is a process and technology approached which is designed to detect, maintain and in *MOST* case correct security, compliance and threats associated with an organization's solution and service deployment within their AWS account. CRT processes monitor security controls in real-time to ensure the risk and/or threat treatment (Control Intent) is working as designed or at least within an intended margin of acceptance base on guard rails, swim lanes and/or rules built into the control to allow for business operations.

splunk> .conf18

Stages of Security Automation and Orchestration



Splunk Slides

The Result: "*AWS Trust Boundary In a Box*"

1. Templates, Scripts, Functions and Recipes for securely deploying regulated workloads "Type Accreditation" (Pre-Audited), for all stages of Cloud Service adoption, (Migrator, Forward, Native)

2. Defined operational security and compliance tolerances scripts, functions and treatments (e.g. Guard Rails) for constrained secure operations across the DevOPS CI/CD and CRT through the use of **Governance as Code** (GoC) practices

3. Deployable Continuous Risk Treatments (CRT) resources (e.g. AWS & Partners solutions)

Thank You!

Don't forget to rate this session
in the .conf18 mobile app