Automating Security in DevOps Pipelines

DJ Schleen, Security Architect, Aetna
@dschleen

SANS Secure DevOps Summit and Training
October 10, 2017

Healthier Lifestyles

- Helping our customers and our community live healthier lifestyles since 1853
- In 1954, Aetna ordered our first computer – an IBM 650
- Acquisitions and affiliates:
  - Medicity in 2011
  - iTriage in 2011
  - Coventry in 2013
  - bswift in 2014
  - Many many others…
- Some affiliates have been practicing DevOps since their inception – without knowing it

aetna

What does DevOps Mean for Security?

An Unprecedented Opportunity

- Everyone supports the goals of the Business
- Automation and Tool Adoption
- Creativity, Collaboration and innovation
- Repeatable Processes
- Resiliency/Scalability
- Continual Feedback
- Anyone can commit to production – as fast as possible
- Application as Code / Infrastructure as Code / Security as Code
- Reduction of production operational impacts
- Easily auditable processes
- React Quickly to Software Vulnerabilities and their remediation

DevOps supports the vision of a continuous delivery culture, and the addition of security controls into an automated environment.

success = **people** * process * tools

Program Overview

Investing in People

People

- Enhance current CBTs
- Engage Security Champions
- Cross train Security staff
- Engage APLC / Agile teams
- Invest in QA Maven program

…will yield significant benefits

Process

- Establish Gates
- Defect Management
- Risk Management
- Real time KPI
- Governance

Technology

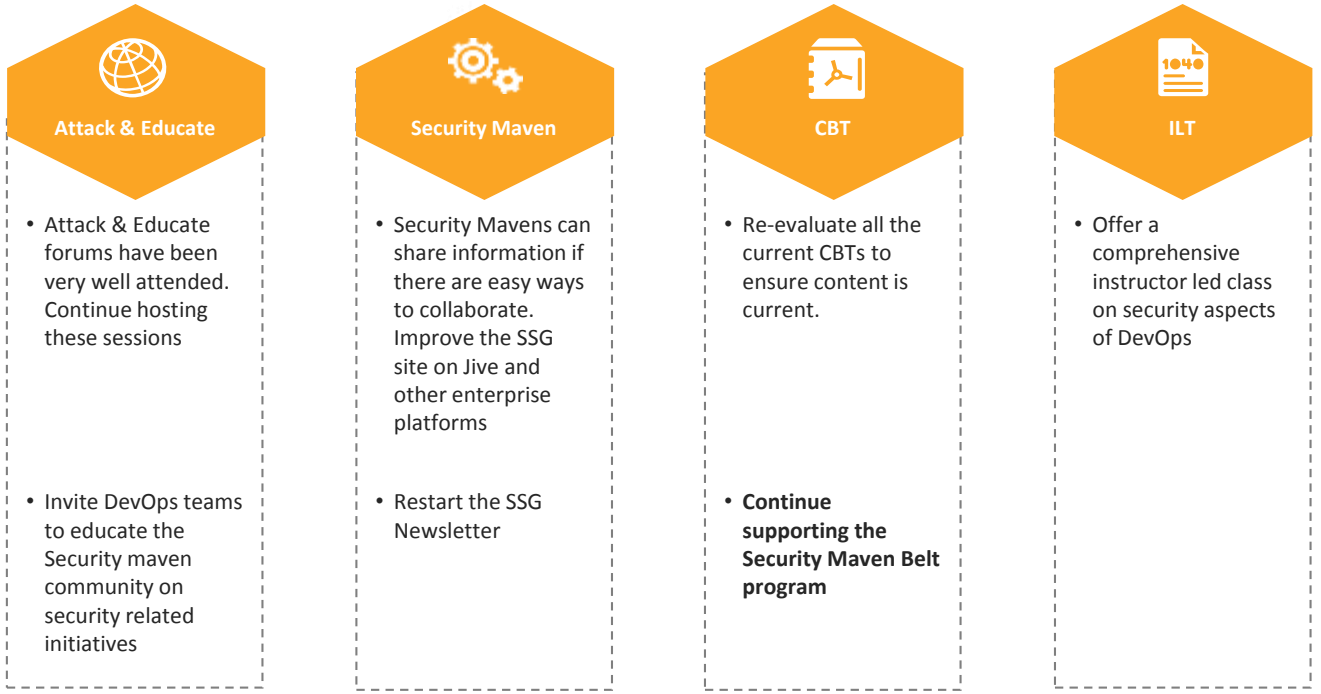
- ARA – Threat Modeling
- Static Analysis / Open Source Security
- Dynamic Analysis / Ethical Hacking
- Container Security
- Run time protection

…and will result in an **improved security posture**

	Aetna's Current State	Aetna's Future State
SDLC	• APLC, SAFe, SRWs	• DevOps
Security findings	• Software defect management is separate from Security findings • Security findings not part of the SDA data mart	• Consistent and real time defect Management • Powerful data and analytics to drive down defect density, improve security posture, and identify threats

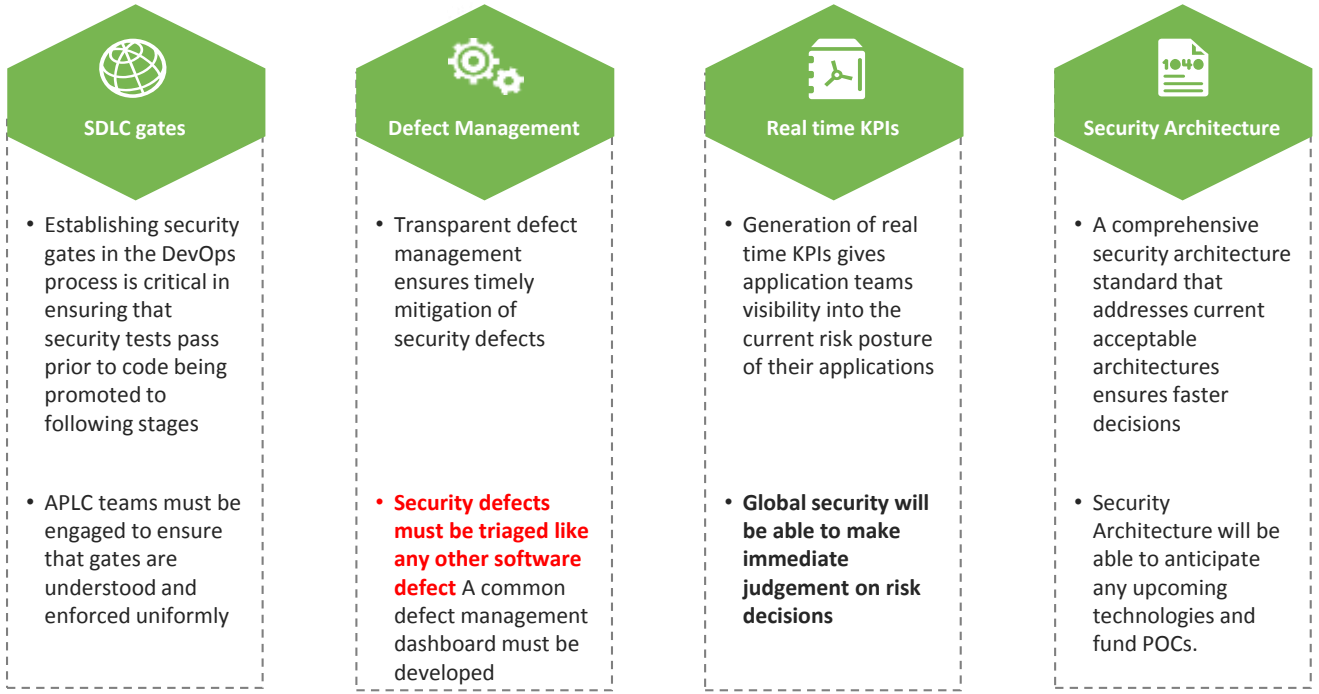
Investing in our People

Value: Ensuring a trained and motivated security community is critical to successfully implementing security in a DevOps process



Building Security into the Process

Value: Ensuring a process where security artifacts are built into the SDLC is critical in ensuring the delivery of secure software across the enterprise



Specialized Security Tools

Value: While the principles of software security is the same, there are significant differences in technology stacks and the tools we must support to secure each stack.

Open Source

- Open Source Software Management Program has been implemented. Javascript, Java and PHP are currently supported. ~200 applications onboarded
- Plugin to Jenkins has been implemented.

Static Analysis

- IDE Based tools used effectively throughout the enterprise.
- Centralized SAST offers broad language support with more coverage

Dynamic Analysis

- Automated triggers when new applications hit staging and production
- Continuous nightly scans
- Output exported and injected into our WAF

Ethical Hacking

- Ethical hacking is an intentionally manual test that can take ~2 weeks to complete.
- Continues investment in training staff to keep up testing capabilities.

Container Security

- Implemented toolsets to ensure security of the containers
- Notarization and cryptographic signatures
- Policy Governance
- Evaluating RASP tools to ensure application layer security and enable faster promotion to production

Security in the pipeline…

What are our Goals?

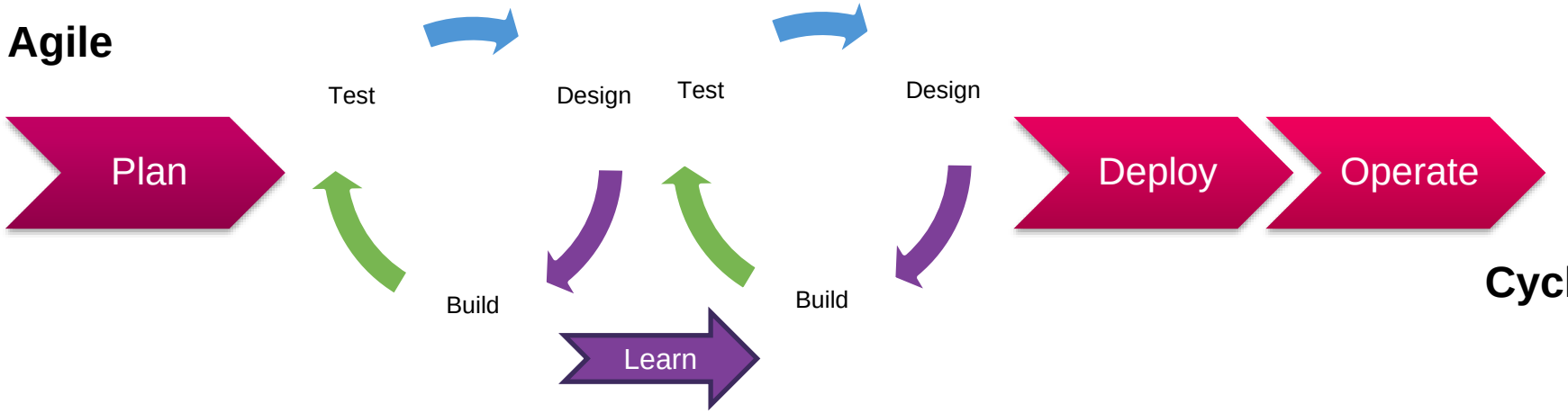
- Integrate Security knowledge and Secure coding practices into our DevOps teams
- Expand the traditional cultural mindset of Security Teams and start getting our hands dirty by coding
- To not just ”automate the scan button”
- Ensure that software passes through a well defined and automated set of gateways that assess code security
- Utilize automation to stop or pause the delivery pipeline when critical vulnerabilities are detected or manual intervention is necessary.
- Provide development teams with not just vulnerability information, but with actionable remediation guidance
- To not forget about software after production deployments. You are only as good as your last security assessment

Security pros need to code too! Code your security into the pipeline and go enjoy a cold beer with the #DevOps peeps.

#SecurityAsCode #IaC

Is Agile “agile” enough?

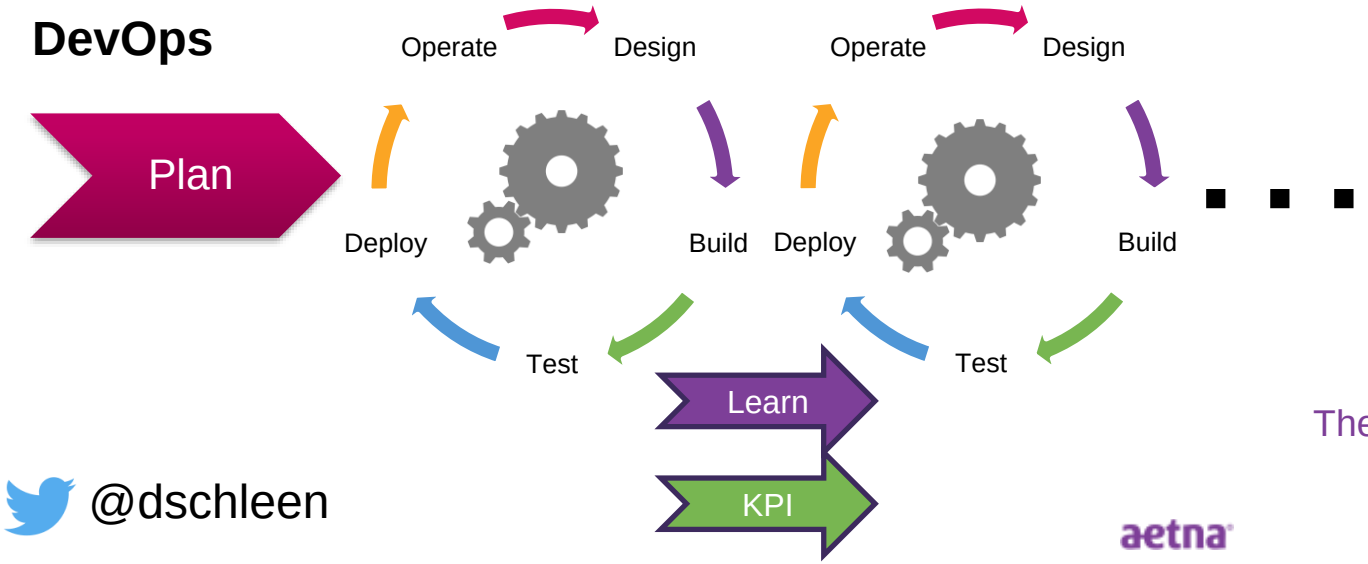
Agile



10 – 20 Releases

Cycle Time: Weeks - Months

DevOps



Your imagination is the limit...

Cycle Time: Minutes - Hours

The faster you can deploy to production, the quicker you can remediate critical vulnerabilities and zero days

@dschleen

aetna

Faster Cycle Times, Increased Flow

CI & IDE Integrated Security Tools and Controls: Support both Static Analysis and Open Source Analysis

Embed controls into the applications/platforms

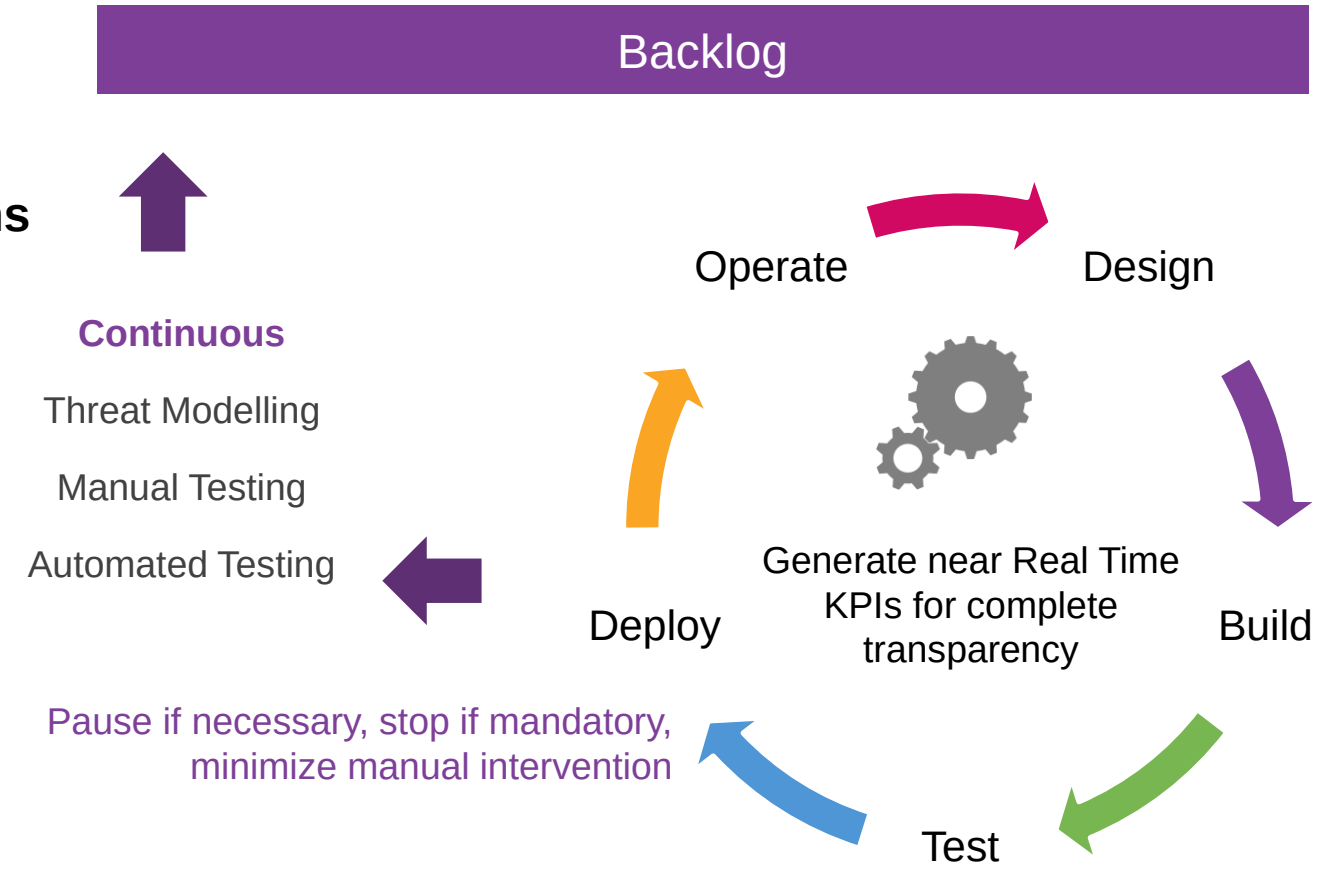
- Runtime Application Self-Protection (RASP)
- Container Security

Manual Security Testing (out of band)

- Dynamic Assessment
- Ethical Hacking
- App Red Team

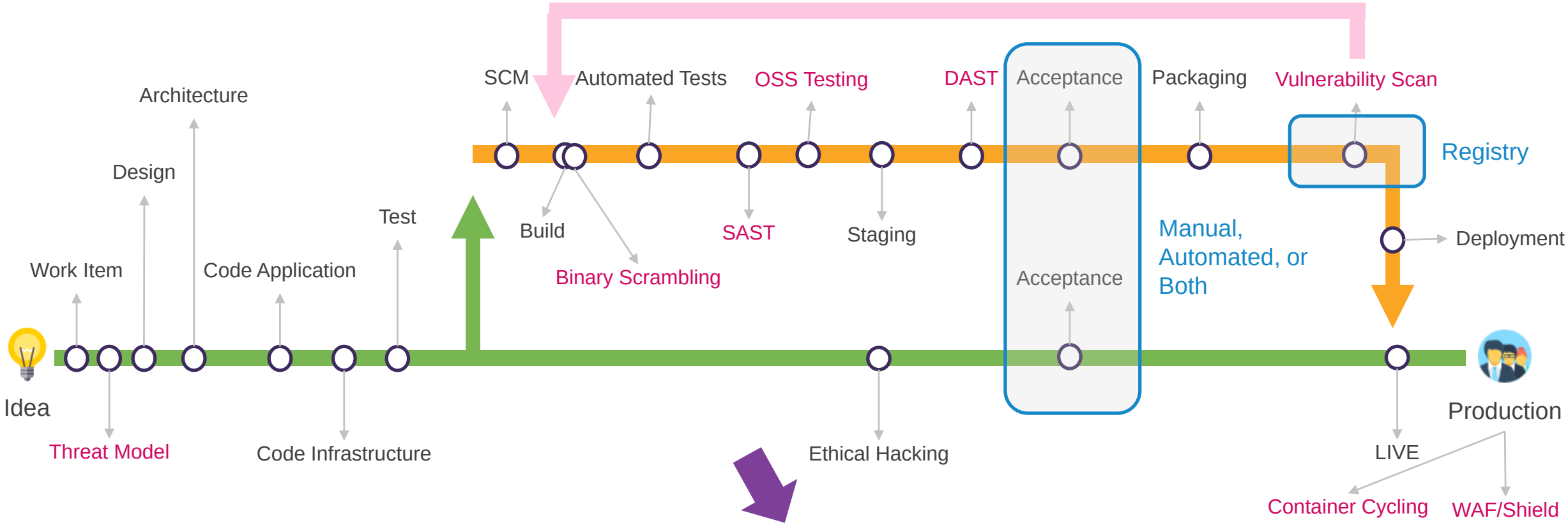
Automated Security Testing (in band)

- Automated SAST/DAST, etc.



Icon made by Google from www.flaticon.com

A *moment* in the life of a feature



...coding **Security** in without ***disruption***

Challenges

- **Integration of Security tools in a manner that supports objectives for actionable continuous feedback**
- Choosing the toolsets that provide as least disruption as possible to DevOps teams.
- Extracting the proper KPI's and indicators from the process and making them actionable.
- Tailoring the People, Process, and Tooling to unique environments
- Evolving the habits of 3,500+ developers
- Traditional infrastructure mindsets. There's a new way of doing things
- Multiple "flavors" of DevOps in different parts of the organization

Key Takeaways

- Don’t fear deploying rapidly and often into production.
- Always gather information in the form of KPIs and make them ACTIONABLE
- Support the organization with tools, techniques, and best practices
- Automate. EVERYTHING.
- Defects are defects - regardless if they are a code defect or security vulnerability.
- Code your Infrastructure – eliminate access to physical or cloud based machines
- Choose tools that interfere minimally with flow. Manual tasks like penetration testing should be done out of band
- People turn dreams into reality

Use automation to enable traceability of production bound features and deliverables.

Thank you!

@dschleen

aetna®