DevSecOps by Design

How to Incorporate Security & Compliance earlier than Testing and Scanning.

Trevor Young
Chief Product Officer
Security Compass
About Security Compass

Trevor Young
Chief Product Officer, Security Compass

founded in
2004
by security professionals

15%
of Fortune 100 are our customers

Developer Centric
Threat Modeling
platform

Application
Security Training
& ISC² Certification
DevSecOps By Design

Products are built in a way that protects against malicious cyber actors gaining access to devices, data & connected infrastructure. Key activities include:

- Training & Awareness
- Coding Practices, Policies & Controls
- Secure Architecture Design & Requirements
- Threat Modeling
Traditional Approach to Security

Software Development Life Cycle (SDLC)

Before Coding
- Penetration Testing
- Compliance Reviews
- Software Scans - Identify

After Coding

TIME

REACTIVE  SLOW

ATO
Typical Organizations Focus on Testing & Patching

Figure 3: DevSecOps Software Lifecycle
Design Stages in a DevSecOps Lifecycle

Sources: DoD Enterprise DevSecOps Reference Design, Security Compass
What Is Driving DevSecOps By Design?

**HIGH PRIORITY for US Fed**
- Expansion of mandated or recommended security and compliance best practices

**CUSTOMER DEMAND**
- Very time consuming to fix security vulnerabilities discovered later in the SDLC

**TIME & MONEY**
- Increasingly customers and regulators are demanding secure software practices.

50%

Federal Agencies surveyed indicate shifting left is a Top 3 priority

149 DAYS

to fix critical issues

$4.35M ($9.4M US)

Global average total cost of data breach

Source: 2023 Federal DoD Perspectives on Application Security (Golfdale / Security Compass)

Source: AppSec Stats Flash Report (NTT Security)

Source: IBM Data Breach Report 2022
What's Preventing Adoption of DevSecOps by Design?

**MANUAL WORK**

Developers are struggling to keep up with security and compliance requirements

42% staying up to date

Source: 2022 Developer Perspectives on Application Security (Golfdale / Security Compass)

**SCALING CHALLENGES**

There aren’t enough experts to scale application security activities.

1:100 AppSec to Developer Ratio

Source: Stop Checking Boxes And Start Effectively Securing Development Pipelines (Forbes)

**COLLABORATION CHALLENGES**

74% of developers first engage with security after the design phase

Sources: 2021 State of Threat Modeling, 2022 Developer Perspectives on Application Security (Golfdale / Security Compass)
A 'DevSecOps by Design' culture with executive support, motivated teams and a phased adoption can overcome these challenges.
Three Step Framework for Security by Design

1. **EDUCATE**
   - Provide foundational role-specific security awareness training for development teams

2. **EMBED**
   - Cultivate depth of security knowledge through coaching, to advocate for a Security by Design culture

3. **EMPOWER**
   - Implement security requirements, threat modeling, and personalized training to realize Security by Design
In this step, organizations lay the foundation for security by design by raising awareness of security in software development teams.

With a foundation of basic awareness, development teams will be ready to tackle more advanced security concepts, and become ready to embrace process change necessary for security-by-design.

Key outcomes:
- Risk reduction through developer education
- Increased security awareness
Example Course Modules to Secure and Defend Against Costly and Damaging Vulnerabilities

**Fundamentals**
- OWASP TOP10
- Secure Design
- Coding
- Testing

**Secure Coding**
- Defending Web APIs
- .NET
- php
- C
- python
- django
- node
- React

**Secure Mobile**
- Mobile Security Fundamentals
- iOS
- Android

**Operational Security**
- OpSec, DevSecOps, Defending Databases, Defending Containers

**Compliance**
- Privacy Fundamentals,
  - CCPA
  - HIPAA
  - GDPR
  - PCI-DSS
  - PCI Secure Software Lifecycle
  - PCI SSF

**General Awareness**
- Security Awareness
  - DevSecOps for Managers
  + new courses planned.
Support Implementation and Process change with Security Champions or Coaches

- Begin change management process for embracing more comprehensive security by design
- Assign champions on scrum teams for advanced training and best practices
- Allow AppSec team to focus on policy, critical monitoring, opportunities to scale
- Start small - Setup a few teams to iterate until it's effective before rolling out across departments
- Leverage vendors (ie. expertise, service & support plans etc.)
Automate security by design through Developer Centric tools and processes to reduce disruption

- API's & SDKs are great (if you have the time)
- Personalized Just-In-Time Training and Hands-On learning
- Start small (critical items only) to reduce burden

Key outcomes:

- Faster time to market for secure software
- Lower risk from software
- Security-by-design fully integrated into existing development processes
Case Study
MEET OUR SOFTWARE SOLUTIONS COMPANY

215 APPLICATIONS

1:250 APPSEC EXPERT TO DEVELOPER

REQUIREMENTS

- OWASP SAMM
- FedRAMP
- NIST
- PCI-DSS
- HIPAA
- CJIS (Criminal Justice Information Systems)
- CMMC
BEFORE

Meetings & Spreadsheets
[Hours]

Manual Analysis
[Days to weeks]

Security Control Implementation
[Days to weeks]

Often not in Scope

What Are We Working On?
What Could Go Wrong?
What Do We Do About It?
Did We Do A Good Job?
SECURITY BY DESIGN

1. EDUCATE
   Provide foundational role-specific security awareness training for development teams

2. EMBED
   Cultivate depth of security knowledge to advocate for a Security by Design culture

3. EMPOWER
   Implement security requirements, threat modeling, and just in time training to realize Security by Design
In this step, our customer laid the foundation for security by design by raising awareness of security in software development teams.

With a foundation of basic awareness, development teams are ready to tackle more advanced security concepts, and are ready to embrace the process change necessary for security-by-design.

Key outcomes:
- Risk reduction through developer education
- Increased security awareness
Transition from basic developer security awareness to fostering security expertise with breadth & depth of training

Incorporate basic program design and workflow for adopting Developer Centric Threat Modeling

Begin change management process for embracing more comprehensive security by design

Key outcomes:
- Increased security expertise in development teams, reduce burden on central security teams
- Visibility of current state risk and cost metrics and quantified goals for security-by-design program
- Increase chances of success of adopting security by design by beginning a change management process
Fully embrace security by design through Developer Centric Threat Modeling

Key outcomes:
- Faster time to market for secure software
- Lower risk from software
- Security-by-design fully integrated into existing development processes
AFTER

Sources: DoD Enterprise DevSecOps Reference Design, Security Compass
Your developers are not security or compliance experts... and you can’t expect them to be.

But with the right tools and processes you can protect your brand and deliver secure products faster than you ever thought possible!
Thank You

For more information, contact us at www.securitycompass.com