Build Security In With Coverity
Give Developers Early Feedback to Identify Security Issues

Ashutosh Kumar, Product Marketing Manager
James Croall, Director, Technical Product Management
11/5/2020
Topics Covered

**Evolving AppSec paradigm**

- Importance of onboarding developers

**Feedback mechanism in the CI pipeline**

- Feedback in the IDE
- Feedback in a pull request
Evolving AppSec Paradigm
The Tussle

“The only thing more dangerous than a developer is a developer conspiring with security. The two working together gives us means, motive, and opportunity.”

The Phoenix Project

“Compliance is not optional. It’s the law. My job is to keep them out of orange jumpsuits, and so I did what I had to do.”

The Phoenix Project
The Perception Around Security

Governances, rules, and gates  
Optimize for risk  
Waterfall inspired, out-of-band  
Highly centralized

“In any value stream, there is always a direction of flow, and there is always one and only constraint; any improvement not made at that constraint is an illusion.”

The Phoenix Project
The Evolution in Security

Engineering-led
Feature velocity-focused
Optimized for business and risk
Metric-driven, fast, in-line; everyone is responsible
Decentralized security
The Security Practices Today

Developers analyze code during development

Code repository

Build, integrate, test

QA and DevOps teams integrate automated security tools into CI/CD pipelines

Security teams empowered to independently scan projects for security dashboards / audits

© 2020 Synopsys, Inc.
Who Needs Static Analysis?

Security executive
Development executive
DevOps manager
Developer
Coverity: SAST Solution for Your Needs

Comprehensive  Accurate  Integrated  Fast  Intelligent
# Coverity Capabilities

<table>
<thead>
<tr>
<th>Language support</th>
<th>Security guidelines</th>
<th>Standards compliance</th>
<th>SDLC workflow</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fortran</td>
<td>OWASP</td>
<td>CERT, ISO 26262</td>
<td>Android Studio</td>
</tr>
<tr>
<td>.NET Core</td>
<td>Mobile Security</td>
<td>DISA, PCI</td>
<td>IntelliJ IDEA</td>
</tr>
<tr>
<td>node</td>
<td>OWASP</td>
<td>AUTOSAR</td>
<td>JIRA</td>
</tr>
<tr>
<td>Objective-C</td>
<td>CWE Top 25</td>
<td>MISRA</td>
<td>Bugzilla</td>
</tr>
<tr>
<td>JS</td>
<td>CWE</td>
<td>NVIDIA CUDA</td>
<td>eclipse</td>
</tr>
<tr>
<td>php</td>
<td>Open Web Application Security Project</td>
<td>NVIDIA CUDA</td>
<td>eclipse</td>
</tr>
<tr>
<td>python</td>
<td>CISQ</td>
<td>CISQ</td>
<td>git</td>
</tr>
<tr>
<td>C/C++</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Java</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C#</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.NET develop</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Scala</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Go</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Kotlin</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RAILS</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TS</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
• Coverity is compliant with CISQ’s new Automated Source Code Data Protection Measure
  – Coverity covers all of the 36 parent weaknesses in the ASCDP Measure
  – The Synopsys Polaris platform aggregates weaknesses by Technical Risk Indicators, including those representing non-compliance with data protection guidelines
Static Analysis for Security Teams

Benefits to multiple teams throughout the software life cycle

Coverity static analysis helps you:
✓ Reduce security exposure in your apps
✓ Reduce project risk by finding problems earlier
✓ Lower your software development costs

Cost of fixing vulnerabilities

© 2020 Synopsys, Inc.
Static Analysis Helping Developers

Benefits to multiple teams throughout the software life cycle

Developers value:
• Accurate, actionable findings—bugs and vulnerabilities
• Easily integrated, unobtrusive tools
• Focus on fixing defects, instead of triaging and managing tool integrations

Coverity strengths:
✓ Large library of checkers > depth of analysis
✓ High accuracy findings (low FP rate)
✓ Fast, comprehensive analysis during development
✓ Integrated into popular IDEs
Build Feedback Mechanism in CI Pipeline
Advantages of Shifting Left

- Reduce security issues going into production code
- Lower cost of fixing issues = Better ROI
- Improve application robustness and process agility
- Improved application security = Preserved company reputation
Feedback to Developers

Feedback in the IDE as the code is being written

Feedback on pull requests as the code is being merged
Code Data Flow

- Just-in-time feedback to developers
- Feedback via pull requests
- Feedback is provided through the existing workflow.
- Security weaknesses are blocked from propagating upstream
Feedback in the IDE

Synopsys Code Sight™

The Code Sight IDE plug-in

• Helps development teams shift left
• Helps developers find and fix issues before checking in code
• Simplifies—Just open a file
• Scans for thousands of different potential problems as you code

Developer

“Is there a way I can look at findings locally?”

“Can you help me find mistakes as I am coding?”
Feedback in the IDE

- Code Sight system architecture
- Code Sight supported IDEs
- Coverity® with Code Sight
- Live demo
Code Sight System Architecture

1. Code Sight IDE plugin
2. Incremental, high-fidelity analysis
3. CI/CD integration
4. Run analysis on the platform
5. Central issue triage, management and reporting
6. Alerts and notifications

• Invoke analysis
• Local issue triage and management

Polaris Software Integrity Platform™ central server in the public/private cloud or Coverity Connect central server for on-premise deployment
Code Sight Supported IDEs

- Eclipse and compatible IDEs
- Microsoft Visual Studio
- JetBrains IDEs such as IntelliJ
Coverity With Code Sight

Main editor

Project view
Coverity With Code Sight

- Crisp issues view
- Prioritized vulnerabilities by category
- Triage and dismiss vulnerabilities
- Link to CWE description
- Dataflow view: main and supporting events
- Related eLearning courses

Scroll down
LIVE Demonstration: Feedback With Code Sight
Feedback in a Pull Request

- Pull request analysis: Key factors
- Live demo
Feedback in a Pull Request

- Incremental scan
- Result summary
- Ability to create issue tickets
LIVE Demonstration: Setting Up Incremental Analysis in GitLab CI
More Questions?

• Visit the Synopsys Software Integrity Community page:
  • https://community.synopsys.com/s/article/Prevent-Security-Weaknesses-from-Escaping-a-GitLab-Merge-Request
Questions
Thank You