SDL at Scale
Growing Security Champions

Ryan O'Boyle

Black Hat Europe 2018

Ryan O'Boyle

Ryan O'Boyle is the Manager, Product Security at CA Veracode. Prior to joining Veracode, he helped create the internal penetration testing team at Fidelity Investments. He has presented at conferences including AppSec USA, AppSec EU, and RSA Europe. Throughout his career, Ryan has focused on not only finding software vulnerabilities but helping developers fix and avoid them altogether. Throughout his life, Mr. O%27Boyle has collected many stories about apostrophes.

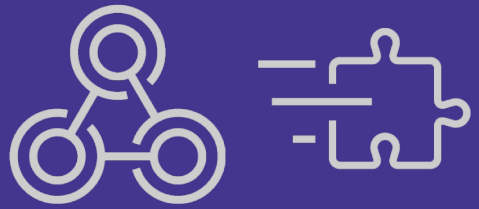
What Is a Security Champion?

A product team member responsible for ensuring security is incorporated into the team's products and processes.

Seed the Program

Get Commitment

Security



Development

Management

Seed the Program

Build a Security Culture

Lunch and Learns

Post-Con Summaries

Capture the Flag (CTF)

Find Your Champions

Find Your Champions

Influential team members

- Trailblazers
- Seniority, skills, passion

Developers, testers, any role

Not ramping up on the product

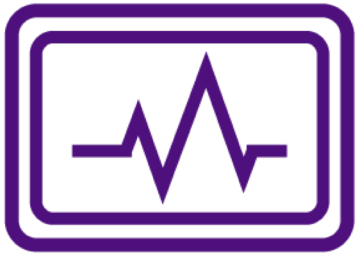
Not overloaded


Put Them in Action

Phase 1

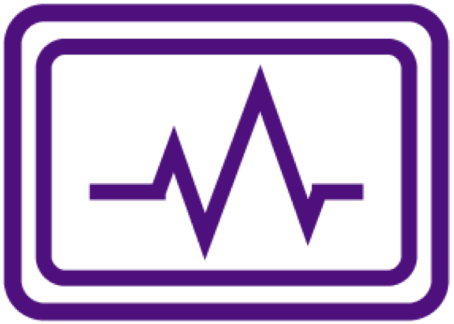
Phase 2






Reward Them

Stay in Touch



Monitor Progress

Product Security Maturity Model (SAMPLE)

	Base	Beginner	Intermediate	Advanced	Expert
Training	• No formal security training	• Some team members (≥10%) have taken a basic secure development eLearning course	• **All** team members have taken a basic secure development eLearning course • Security Champions and Team Leads have taken additional advanced or domain specific training	• All team members take a secure development eLearning course **annually** • Security Champions and Team Leads have taken additional advanced or domain specific training • Security Champions and Team Leads conduct informal learning sessions for other team members	• All team members take a secure development eLearning course annually • Security Champions and Team Leads have taken additional advanced or domain specific training • Security Champions and Team Leads routinely conduct **formal and informal** learning sessions for other team members
Secure Design	• Security is not a design consideration	• Security requirements are generally defined after development has started or completed	• Threat modeling before major components or features • Security requirements are defined before major components or features	• Threat modeling before **all** components or features • Security requirements are defined before **all** components or features • Threat modeling is incorporated into the story planning/grooming process • Security requirements are defined as story Acceptance Criteria on most (>50%) relevant stories	• Threat modeling before all components or features • Security requirements are defined before all components or features • Threat modeling is incorporated into the story planning/grooming process • Security Acceptance Criteria defined for **all** relevant stories
Security Code Review	• No security specific code review	• Major components are reviewed by Security Team or 3rd party • Only the most critical findings are addressed	• Security team review of high risk stories • High and critical findings are addressed	• Some peer Security Review within teams • Security team review of high risk stories • Automated code checks • Most findings (critical, high, medium) are addressed within 30 days	• Peer Security Review of all pull requests • Security team review of high risk stories • Custom automated code checks • Holistic review of product by Security Team or 3rd party periodically • All findings are addressed rapidly (≤7 days)
Security Testing	• No security testing	• Annual 3rd party Pen. Test (where required by policy) • Only the most critical findings are addressed	• Annual 3rd party Pen. Test (where required by policy) • Ad hoc SAST and/or DAST • High and critical findings are addressed	• Annual 3rd party Pen. Test (where required by policy) • SAST and DAST on regular basis (e.g., per-release, monthly) • Test plans include security requirements • Most findings (critical, high, medium) are addressed within 30 days	• Annual 3rd party Pen. Test • Continuous SAST and DAST integrated into build and bug tracking systems • Security testing integrated into unit and feature tests • All findings are addressed rapidly (≤7 days)
Third Party	• Security is not a consideration when managing third party assets	• List of third party assets and versioning information is documented	• List of third party assets and versioning information is documented **using a repeatable scripted process** • Security track record is taken into account when choosing third party assets	• List of third party assets and versioning information is documented using a repeatable scripted process • Third party assets are chosen based on proven security track record • Team has setup alerts when new security events that effect the product become available and have a process defined for applying relevant patches or configuration changes	• List of third party assets and versioning information is documented **with no manual effort** • Third party assets are chosen based on proven security track record • Team has setup alerts when new security events that effect the product become available and have a process defined for applying relevant patches or configuration changes

Reflect & Iterate

Questions?

Thank you