VERACODE

Want to Be a Rockstar Developer?

CRANK UP YOUR SECURITY SKILLS

HERE'S WHERE TO START

More than 76 percent of college-educated developers who responded to our DevSecOps Global Skills Survey said they were not required to complete any courses focused on security during school.[1] That's a pretty big number — and it means some of the responsibility to learn best practices in security is on developers like you. The good news? You can get there with a combination of the right tools and resources, some help from your security team, and good ol' fashioned grit that can send your career to center stage.

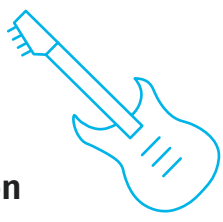
1

Treat Every Day Like Rehearsal Day

Practice may not always make perfect in a world of persistent security risks, but it'll prepare you to take on threats. Using hands-on tools like Security Labs with exercises in your preferred languages and frameworks will help you conquer modern application threats by patching real code for practice. It'll train you to spot flaws early on too, so your team can hit tight development deadlines. Brings a whole new meaning to "We Are the Champions."

2

Become a Security Champion



Speaking of champions, did you know that's one way for you to really shine on your team? Becoming a security champion shows management (and your peers) that you keep security at top of mind and work to improve the quality of your code. It opens the lines of communication between your team and the security team to relay best practices and needs, while raising awareness about issues your team faces. A real powerhouse move.

3

Study Your Missteps (and Correct Them)

Software development is kind of like playing a game of chess. You need to watch your opponents and learn from them while safeguarding your data from sabotage. Dan Murphy, Principal Software Engineer at Veracode, suggests seeking out more information on failures in software development — including famous security flaws and how they happened — to improve your skills. Subreddits like /r/netsec are a great place to start, as is Hacker News. Learn from the greats, as they say, and then do it better.

4

Think Like an Attacker

In the spirit of "Walk Like an Egyptian," we want you to think like an attacker. Tim Jarrett, Director of Product Management at Veracode, says that learning how to break things like a cyberattacker is key to becoming a security-minded developer. Set up Google Alerts for security news and read about how security researchers find flaws so you can absorb what they do best. Imitation is the sincerest form of flattery — and can help you hit the charts like a Rockstar developer.

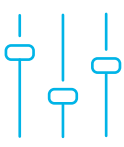
5

Know Common Flaws Inside and Out

Cross-Site Scripting is one of the most common security flaws, and leads to a significant amount of security debt in many organizations.[2] But could you fix it quickly if you spotted the vulnerability in your code? Having that foundational knowledge is critical. If you're not armed with the know-how you need to write more secure code and fix common flaws fast, then you might fall behind in your development team's ensemble. Start here to learn about common vulnerabilities and improve your flaw know-how.

6

Scan Early, Scan Often, and Fix Flaws Fast



"He's going the distance. He's going for speed." Cake had it right — in the realm of secure software development, you need to have the right security solution in place to go the distance, such as Veracode's IDE Scan that provides fast security feedback as you code. If you get into the habit of scanning your code often and fixing flaws quickly, you'll have less buildup of security debt (which means less risk), all while winning the race to market.

7

Automate to Boost Speed and Efficiency

Looking to take the human element out of the picture and speed up production as much as possible? Encourage your manager to integrate automated security tools into your SDLC. For example, automated code scanning in your IDE, with automated generation of tickets related to your IDE, is a great way for you to get feedback quickly and learn as you code to prevent new flaws.

8

Secure Your Open Source Code

Most developers rely on third-party open source code to meet the demands of rapid release cycles. In fact, we estimate that most software is comprised of about 90 percent open source code.[3] Make sure you're staying on top of open source security with tools like Software Composition Analysis that will help you keep track of and address vulnerabilities in open source libraries like a star.

It's time to get the band together.

LEARN MORE ABOUT JOINING FORCES WITH YOUR SECURITY TEAM TO CREATE SECURE CODE.

LEARN MORE

VERACODE

[1]Analyst Report DevSecOps Global Skill Survey; [2]State of Software Security Report, Volume 10; [3]Understanding Your Open Source Risk eBook