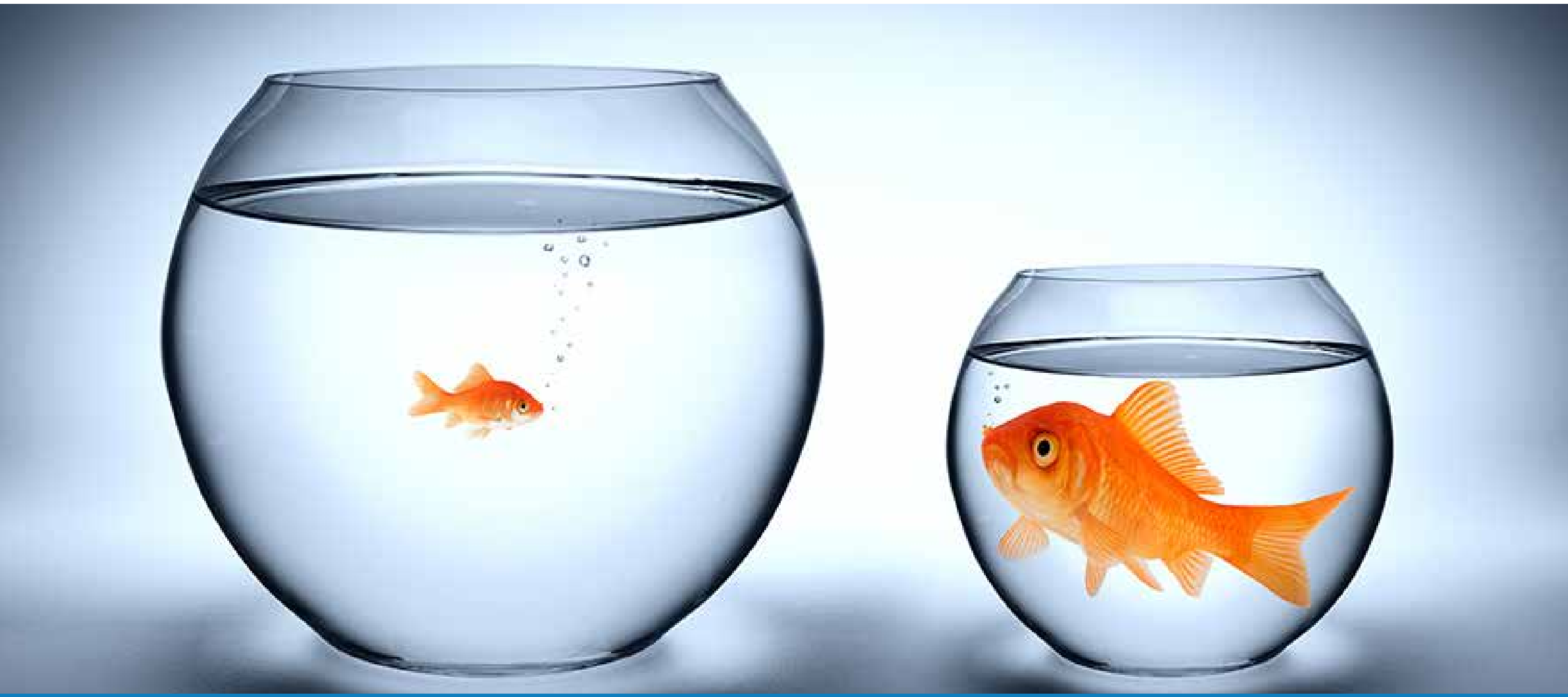
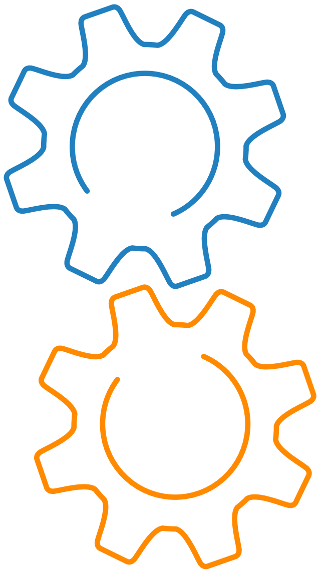
SECURITY INNOVATION



Evolving Beyond One-Size-Fits-All Software Security Training

SUPPORTING MODERN SOFTWARE DEVELOPMENT ROLES WITH MODULARIZED CYBERSECURITY CURRICULUM

THE CHANGING LANDSCAPE OF SOFTWARE DEVELOPMENT



The rapid changes in application development methods over the last several years are putting tremendous pressure on security teams to keep up with relevant cybersecurity training.

Agile development, DevOps practices, and continuous delivery/continuous integration (CI/CD) methods are completely rewriting the development playbook at many organizations:

- The software development lifecycle looks completely different at many organizations, with smaller, more frequent changes being made to code
- Team dynamics are shifting with operations and development staff working more closely together
- The technology stack is constantly evolving with new cloud platforms, containers, microservices architecture, automated testing, and other technologies taking root in the blink of an eye
- Roles are diversifying as developers self-provision infrastructure, operators control infrastructure with code and new roles like the site reliability engineer find their way to the org chart
- The pool of security stakeholders is broadening as the modern software factory touches everyone in the IT team today

CHALLENGES WITH TRADITIONAL SOFTWARE SECURITY TRAINING

In the face of all these changes most software security awareness programs continue to plod along with the same thing they've always been doing, yielding little to no risk reduction:

- **Providing cookie-cutter material**
  If they provide training at all, it's usually a couple hours covering the dangers of SQL injection, OWASP Top 10 risks, and the importance of considering cloud deployment. While this builds situational awareness, it does little to hone job specific know-how.
- **Focusing too much on developers**
  Rarely is training targeted at DevOps engineers, QA/test automation engineers, product owners, architects or any other of the growing contingent of software security and CI/CD stakeholders.. Instead they

continue to focus on code-level best practices that are only covering a small percentage of the risks introduced by the entire application delivery team.

- **Failing to consider emerging and hot technologies**
  Providing developers contextual detail tailored to a growing list of common coding languages or form factors like mobile or embedded environments is critical. The same holds true for deployment teams who often lack context around securely designing software in the context of microservices, running containers securely, or architecting secure cloud infrastructure.

As a result, software teams are left unprepared, never gaining the incremental practical knowledge they need to deliver secure software. And as they speed up their deployment cadence the modern enterprise is pushing out insecure software more voluminously than ever before.

MODERN SOFTWARE DEVELOPMENT TEAMS DESERVE MODERN SOFTWARE SECURITY TRAINING METHODS

Enterprises need to be able to tailor their training to a person's role and the technology platforms they work with. These stakeholders need training that doesn't waste their time—providing the level of security information and professional context appropriate for their daily work.

At the same time, security teams still need to be able to scale that knowledge across large teams. The only way to do that is by completely rethinking security training models.

Much in the same way that modern software development has had to decouple monolithic applications into loosely connected microservices, software security training must untether security knowledge contained in unwieldy, progressive learning paths. The knowledge should be broken into modularized curriculum components.

Mixing and matching these self-contained security learning modules gives security organizations the ability to provide curriculum tailored to roles and technical specialties without reinventing the wheel every time they design a new learning path.

THE OLD APPROACH ISN'T WORKING

Generalized software security training is hurting security risk postures. Software development and DevOps teams report that lack of training and awareness within their ranks is hurting the state of the software they deliver. Since these untrained teams are delivering software faster than ever through DevOps and Agile, that means they're also introducing new vulnerabilities more prolifically than ever.

Obviously, a big part of the problem is that many organizations are not training their software development teams in security principles at all. But even when companies do invest in software security training they're not getting the most of those investments because the material is outdated and the fundamental design of these courses simply does not sync up with modern software development methods.

Here's why.

Today's approach to 'developer' security training all tends to be built on progressive, linear learning tracks. Like in college, broad topics are presented in lengthy courses at the 100-level, 200-level, 300-level and so on.

The trouble with this is two-fold:

- The number of software security stakeholders has exploded beyond the core developer audience
- The variety of development languages, deployment automation tools, and infrastructure platforms used by these different stakeholders has similarly exploded.

That's creating a lot of challenges for organizations still stuck in the progressive, linear training model.

Back in the day, software security training could be keyed to a pretty homogenous audience of developers, all using a handful of industry standard tools and coding in a few major languages. Worst case scenario, they might need to conduct some additional targeted training for QA or pen testers in the organization.

According to ESG and ISSA, the most acute skills shortages include:

- Cloud computing security (33%)
- Application security (32%)
- Security analysis (30%)

Now in this DevOps-driven world, literally everyone in IT is a software stakeholder, and the systems and code they touch are highly interconnected. Developers have to know about deployment environments. And operators have to understand code at some level, because so much of infrastructure today is run as code.

Meantime, the CI/CD pipeline and DevOps tool chain is mushrooming the amount of new technologies in use. Quickly evolving technologies like containers, container orchestration platforms, cloud services, open source libraries, and more all need to be securely configured, and they're often spun up by developers.

So security trainers are challenged to present some pretty advanced concepts—such as protecting mobile data in transit—to a mixed audience. Learners may not necessarily be cryptographers or developers...but they are, in fact, making crucial decisions about features that will ultimately impact the security posture of the application.

IN THE FAST-PACED WORLD OF MODERN SOFTWARE DELIVERY, NO ONE HAS THE TIME TO WASTE ON A LENGTHY SECURITY COURSE THAT'S ONLY 20% RELEVANT TO THEM. THEY NEED TARGETED INFORMATION, IN BITE -SIZED LEARNING EXPERIENCES. AND THEY NEED IT TO KEEP UP WITH THE PACE OF CHANGE IN THE TECHNOLOGY STACK.

Software security learners are highly specialized by role and by technology, and they often need deep understanding of just a few technical security mechanics specifically relevant to what they do. The trouble is that they've got to wade through hours of irrelevant course work to find the advanced content they need.

Not only is progressively linear traditional training rigid, but it's also difficult to keep up to date. Every time a huge, interconnected curriculum framework and path is created for software security, it's typically obsolete very quickly.

Clearly, something needs to change.

WHY MODULARIZATION IS NEEDED

Shifting to a modularized software security curriculum is just the change that modern development shops are seeking to solve their training woes. Modularized curriculum breaks down the security building blocks so they can be mixed and matched to individual specialties and roles.

In today's modern software environment different security stakeholders really need to understand the security principles in context with how they're specifically using relevant technology in their daily work. Every role needs the tools and training that allows them to make decisions or guide them to manage people toward appropriate security objectives.

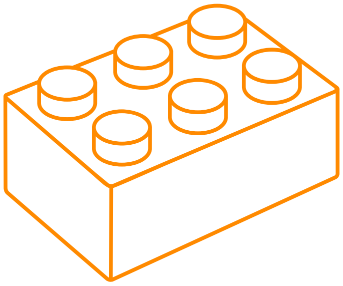
For example, take the concepts and general ideas behind how modern cryptography works. Each person contributing to the implementation and use of cryptography needs to specifically know how the decisions they make impact deployment of encryption features.

- A **developer** needs to know which cryptographic standards to use and which to avoid, and that homegrown cryptography is always a bad idea.
- An **operator** may need to know the details about appropriate key management and why they should never put keys on publicly accessible cloud stores.
- A **product manager** may just need basics in how cryptography works and for which data and risk environments encryption is necessary or mandated by law.

That's a lot of different dimensions to know about a single topic—each with varying degrees of knowledge overlap. In order to efficiently meet this variety of needs, organizations need curriculum that keeps core concepts as untethered as possible. This way an ops person who needs to know some core concept about cloud computing doesn't need to wade through two hours of irrelevant developer training material to get there.

MODULARIZED TRAINING MAKES UNTETHERING POSSIBLE BY BREAKING DOWN SECURITY CONCEPTS INTO INTERCHANGEABLE CORE COMPONENTS.

Modularized training rethinks security training design to be less of a learning path and more of a blueprint for a building made of Lego blocks. With the right pile of component blocks, an organization can design highly relevant curriculum blueprints for numerous audiences with minimal effort or rework.



In a way it is very much akin to the kind of efficient repurposing that developers do with open source components. Like an open source library, modularized training components can be reused and recombined in a way that's completely tailored to the role or technical specialization. Different modules fit together in different ways and, depending on the combination, the completed 'building' presents learners security knowledge in a professional context that makes sense to them.

HOW TO GET THE MOST OUT OF MODULARIZED TRAINING CONTENT

Security and IT leaders need to lay out a detailed plan for how they're going to develop security skills relevant to every function. While each DevOps role needs to understand the fundamentals of secure development, privacy protection, and cloud infrastructure, additional job specific training is needed.

For example:

- **Product Owners** need to understand how mitigate risk as it relates to supporting the application, the data classification level of data processed by the application, and how to safeguard the data based on the cloud infrastructure and operating environment.
- **Cloud Architects** need to understand how to securely design applications deployed on cloud infrastructure. This includes analyzing implementation options for core features as well as configuration options for architectural frameworks, API gateways, microservices, and data sharing.
- **Code Release Managers** need to understand how to detect and mitigate insecure interfaces and APIs and prevent common code-level vulnerabilities. They also need to recognize insufficient identity controls and insecure or out-of-date dependencies.

- **Site Reliability Engineers (SRE)** need to understand how adverse events like Denial of Service (DoS) attacks could negatively affect system availability; and how to implement application whitelisting and apply least privileged access to sensitive data.
- **Automation Engineers** need to understand how to leverage automation without compromising security, and how to conduct vulnerability.

5 STEPS FOR SUCCESS IN MODULARIZED SECURITY TRAINING

1. Assessment: Assess the organization's software security stakeholders—their roles and the technologies they touch—to come up with training requirements/objectives. Either decide to use presets determined by Security Innovation or tweak them based on the assessment results.

2. Pilot Program: Start small with a pilot program of a few learners representing the different learning audiences delineated by the assessment.

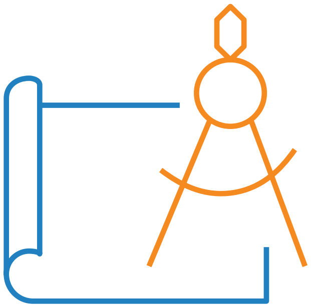
3. Rapid Prototyping: Engage in rapid prototyping from the pilot results to tweak modules included within certain learning plans; for example, the organization may find a product manager needs to understand how embedded tech and IoT works so they may need to add two modules to a plan that doesn't currently include those.

4. Feedback Loop: Once learning plans are rolled out across the organization, utilize learning management tools to look at feedback on courses and trainee success to further tweak paths

5. Concept Reinforcement: Reinforce training with blended learning opportunities. Security Innovation's Cyber Range offers one example of the kinds of real-world opportunities learners should have to see how security concepts and vulnerabilities look in actual software.

One mistake organizations often make is taking a Do-it-Yourself (DIY) approach. Our experience shows that security stakeholders need guidance, as they often 'don't know what they don't know' about security or job functions. Either decide to use presets determined by your training partner or customize them based on the assessment results.

NEXT GENERATION TRAINING
THE CMD+CTRL APPROACH



When our training experts at Security Innovation saw the writing on the wall with regard to instructional design for this industry, we made the shift to put modularized curriculum at the heart of how we train IT teams about software security.

In order to do that, our team took our traditional, linearly tracked curriculum library and atomized it. We assessed our old collection of over 100 software security courses that ranged from 30 minutes to 2 hours and broke down the core concepts into approximately 1,400 topics. Then our team of subject matter and instructional design experts took that raw knowledge library, updated many concepts to account for new technology and development methodology, and began creating our new learning modules.

The goal was to redesign our catalogue into smaller, bite-size courses. Not only is this crucial to support the mix-and-match capability of modularized training, but it's also in sync with the principles of modern instructional design, which favors shorter, more succinct training with frequent reinforcement. We leveraged the National Initiative for Cybersecurity Education (NICE) Framework to help align training concepts and taxonomy to modern IT roles. They also folded in the latest concepts and requirements from other common frameworks and regulations like CWE, NIST, GDPR, and HIPAA.

In the first year of development, this atomization of content helped us double the number of courses we made available through the program we call our CMD-CTRL Defend course library.

Security Innovation now offers over 200 software security training modules, most of which are brief 10-20 minutes in duration. Even the longest are no more than 40 minutes in duration.

Using that taxonomy we created 45 different learning blueprints tuned to default roles and technology paths most commonly found within organizations today. This makes it easy for most teams to easily select learning plans for their most common audiences. At the same time the taxonomy allows for a great deal of flexible customizability so that security teams can personalize their course material for any number of roles or knowledge needs.

We also made sure that all modules are filterable by four dimensions:

- **Role** - with hundreds of varying software security duties, we curated learning paths applicable to 50+ software development and IT job functions
- **Platform** - cloud, mobile, web, IoT and other deployment environments each have unique threats and defensive techniques that learners need to know
- **Standard** - training modules that relate directly to various industry regulations, best practice frameworks, and certifications
- **Technology** - security education on any one of dozens of different technologies used to build and operate software systems

Just like CI/CD focuses on iterative improvement to software, Security Innovation reimagined its curriculum to allow for continuous improvement and updates to courses as technologies and roles change. Through that process our Customer Success Managers have figured out some of the most effective ways to keep software stakeholders current with security knowledge.

"Role-based training looks at the security functions that individuals perform, not their job title. FISMA and other requirements mandate that we do role-based training for staff with significant security responsibilities."

PATRICIA TOTH, NIST COMPUTER SCIENTIST

THE BENEFITS OF THE CMD+CTRL TRAINING PLATFORM

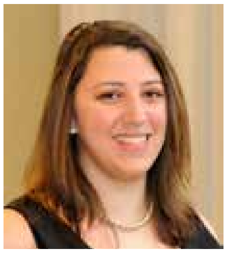
Modularized training has already shown tremendous benefits to Security Innovation customers. Our unique approach to develop microlearning modules helps teams that recognize that security is a shared responsibility to personalize training for all of the roles within their organization.

And they can do so without any heavy lifting. Security Innovation did all of the hard work of mapping security fundamentals to modern development roles and technical specialties. As a result, CMD+CTRL stands as a systematic, well-integrated library where each modularized component can stand on its own or serve as context for adjacent modules.

To learn more about CMD+CTRL, please contact Security Innovation today!

- **150 Micro-courses**
- **20 minutes average**
- **4 dimensions of personalization: role, platform, standard, technology**
- **45 pre-set learning paths**
- **Endless security training Possibilities**

ABOUT THE AUTHOR



Lisa Parcella is the VP of Marketing & Product Management at Security Innovation. With a background in security awareness, product management, marketing communications, and academia, Lisa leverages her vast experience to design and deliver comprehensive security-focused products and educational solutions for the company's diverse client base.

Lisa's primary role at Security Innovation is to work with customers, prospects and industry experts to ensure we are creating innovative and holistic products and programs that address the various needs of today's global workforce. Lisa spearheaded the company's Security Awareness 365 program, an innovative mix of computer-based training, multimedia assets and programmatic tools that helped place Security Innovation on Gartner's Magic Quadrant for Security Awareness since its inception. In addition to managing internal training programs and product lines, Lisa provides strategic counsel and support to the company's clients, helping them optimize their program methods, metrics gathering, messaging and execution.

Before joining Security Innovation, Lisa served as Vice President of Educational Services at Safelight Security. Lisa was responsible for managing subject matter experts and instructors in the creation of Safelight courseware. In this role, Lisa worked with internal teams and a global customer base to create dynamic, interactive learning solutions. Lisa also led the marketing team, working to promote and perpetuate the Safelight brand worldwide.

ABOUT SECURITY INNOVATION

Security Innovation is a pioneer in software security and trusted advisor to its clients. Since 2002, organizations have relied on our assessment and training solutions to make the use of software systems safer in the most challenging environments – whether in Web applications, IoT devices, or the cloud. The company's flagship product, CMD+CTRL Cyber Range, is the industry's only authentic environment to build the skills teams need to protect the enterprise where it is most vulnerable – at the software layer. Security Innovation is privately held and headquartered in Wilmington, MA USA. For more information, visit **www.securityinnovation.com** or connect with us on **LinkedIn** or **Twitter**.