SafeStack

Secure Development Certifications

Recognize your security champions
Assess application security skills
Stand out as a secure development specialists

Contents

About SafeStack

At SafeStack, we believe that software should do more than just meet today's security needs. **It should be built to last and be safe for whatever challenges come next.**

Security is essential, but safety is what we care deeply about. Security is about protecting against known threats—locking gates, setting up cameras. But safety is bigger; it's about protecting against the unknown—like putting a sturdy fence around a pool to prevent accidents.

We make AppSec easy for teams by embedding it directly into your workflow, so security becomes second nature. Our approach is simple: structured education for your entire team, clear guidance to roll out a world-class AppSec program, and visibility into progress so you can confidently meet your compliance goals.

We understand that your team is busy, and we've built our platform to fit seamlessly into your existing resources—regardless of the size of your team or the complexity of your project.

"At SafeStack, we don't just help you build secure software. **We help you build software that's safe.** Because security should support innovation for software that lasts and does good for the people who use it."

Laura Bell Main
CO-FOUNDER AND CEO

Our History

Founded in 2020 by Laura Bell Main and Erica Anderson, SafeStack helps organizations of all sizes to be secure by design.

With its headquarters in New Zealand, SafeStack has grown to be a leader in secure coding training and application security. Supporting over 1500 organizations worldwide, SafeStack's unapologetically opinionated approach to bridging the gap between software development and security has made application security achievable to organizations of all sizes.

SafeStack is supported by Blackbird Ventures, Carthona Capital, Jelix Ventures, NAB Ventures and the New Zealand Growth Capital Fund.

Our social impact initiatives have provided free training and career development paths to military veterans (VetSec), female cyber security professionals (CyberSafe Foundation) and new computer science graduates across New Zealand and Australia.

Meet Our CEO

Laura Bell Main is recognized as a global leader in developing secure software.

With over twenty years of experience in software development and cyber security, she is the co-author of “Agile Application Security” (O’Reilly Media) and “Security for Everyone” (Holloway).

Her work has been featured in many international publications, including WIRED and MIT Tech Review.

She is an experienced keynote and conference speaker, presenting at BlackHat USA, RenderATL, and leading international software development and cyber security conferences.

Her latest book “Fundamentals of Product Security” (O’Reilly Media), is due for release in 2025.

The SafeStack Advantage

Built by Industry Experts

Our team of specialists are software engineers and application security specialists. Their expertise allows us to build platforms and approaches that are respectful and well matched to software development teams worldwide.

Whole Team, Cultural focus

Despite being a technical area, maturing your application security in a sustainable way means engaging your entire software team and changing your culture.

Our platforms and education ensure all roles are educated, supported and involved in secure development .

First Class Support

Our customer success and support team work hard to make sure every organization has the help they need to get the most from their investment. Whether its platform tips and tricks or solving common appsec cultural challenges - we are right by your side.

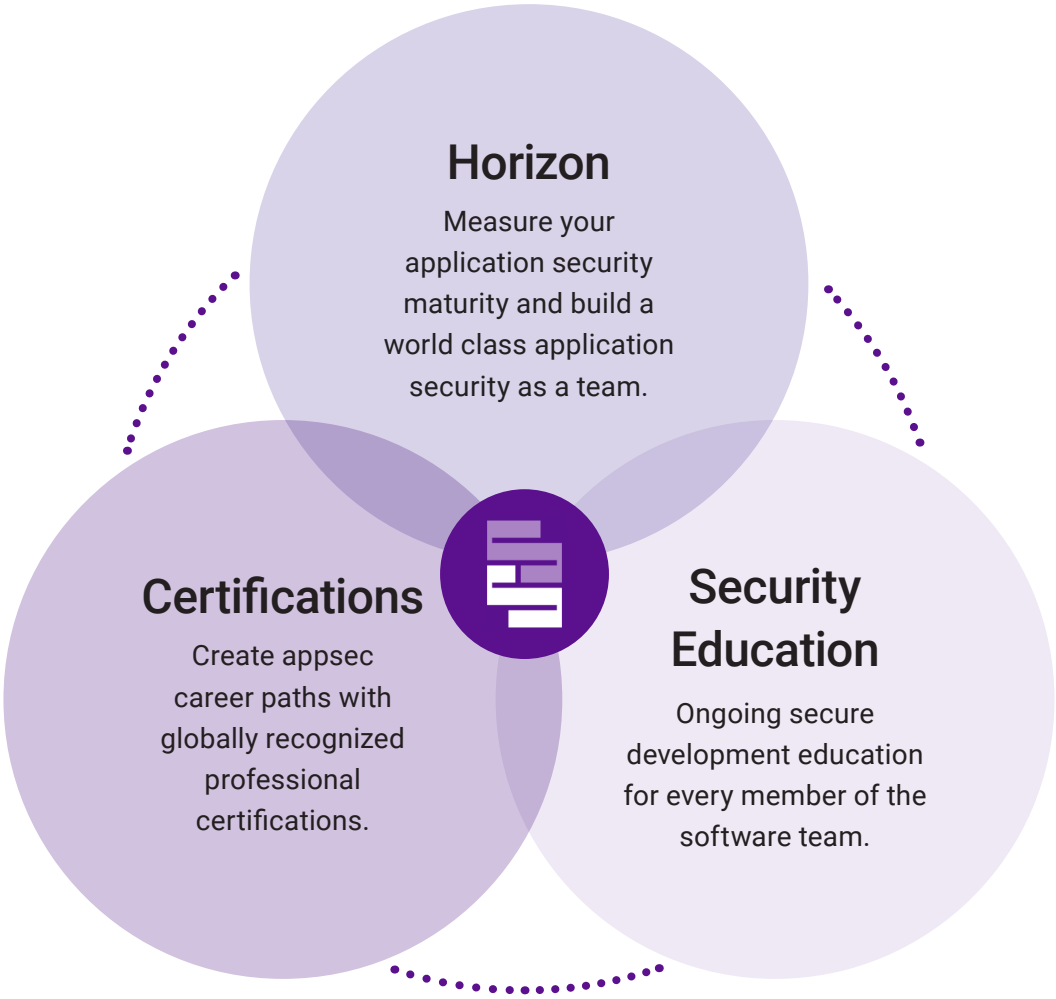
Global Partner Network

We know, sometimes you need a bit if extra help. That's why SafeStack partners with leading application security consultancies worldwide so you can build your own program, with a little help when you need it.

TL;DR:
Product Overview

We make AppSec Easy

SafeStack supports teams to mature their application security, build a secure development culture and meet compliance goals with ease.



Certifications

Professional Qualifications for Software Professionals

Globally recognized professional certifications for software development professionals and leaders

Professional Certifications

SafeStack offers two globally recognised professional certifications that recognize and reward secure development expertise at both professional and leadership levels. These certifications validate essential skills for building and leading secure software practices.

The CSDP and CSDL certifications help link secure development to career growth, performance recognition, and professional developments—empowering engineers and strengthening security culture.

Certified Secure Development Professional

Security is essential to software quality, and secure development skills are critical for modern software teams.

The CSDP certification equips professionals with the knowledge and techniques to integrate security into every stage of the development lifecycle. This industry-recognized qualification validates your expertise and ensures you're ready to build secure, resilient software.

Certified Secure Development Leader

Secure software doesn't happen by chance—it needs leadership.

The CSDL certification recognizes professionals who drive secure development initiatives, shaping both technology practices and security culture. This qualification provides a structured framework to assess your skills in designing, implementing, and managing application security programs across your organization.

Certified Secure Development Professional

On completing the CSDP exam, you will have proven skills for integrating security throughout your software development lifecycle, from design through development, deployment, and beyond.

This includes taking a security-by-design approach to identifying risk and security requirements in software before the code is written.

Learning Outcomes

Stand out as a software security champion for your team

Learn from industry leaders, authors, and experienced practitioners.

Demonstrate your skills in more than just secure coding

Take active steps to design and build secure software at every stage of your software development life cycle.

Take a security-by-design approach to secure software

Don't wait for the code to be written to identify security risks, design for security.

Career Pathways

The CSDP certification integrates security into software quality and will prepare engineering team members for more senior software roles.

Application Security Engineer

Helping software teams identify and address security issues in their systems and designs.

Security Tester

Identifying security vulnerabilities through automated and exploratory software security testing.

DevSecOps

Integrate security throughout your CI/CD pipelines and ensure your data, people, and systems remain safe.

Syllabus: Certified Secure Development Professional

Security is a key part of software quality, and secure development skills are essential to software teams worldwide. With this professional-level certificate, learners acquire skills and approaches to apply secure development practices at every stage of the software development lifecycle.

The **Certified Secure Development Professional (CSDP)** qualification provides a challenging way to reflect on what you have learned and how to apply these new skills.

SS-CSDP-RISK: Security fundamentals and risk for software development

MODULE	MODULE DESCRIPTION
Understanding and calculating security risk	▸ Define risk and vulnerability in the context of software. ▸ Understand the impact of security vulnerabilities. ▸ Calculate risk using CVSS. and other scoring systems. ▸ Use calculated risk to prioritize and respond to security issues and weaknesses.
Identifying the groups and individuals that threaten our security	▸ Understand what a threat actor is in the context of software security. ▸ Identify common threat actor personas and how they differ from each other. ▸ Analyze the motivations behind threats and their associated threat actors. ▸ Identify threat actors for a particular product or software system based on its operating model and context.

SS-CSDP-SDLC: Bringing security to your software development lifecycle

MODULE	MODULE DESCRIPTION
Defining the stages of the software development lifecycle (SDLC) and how security can be applied.	▸ Identify the stages of common software development lifecycles and their aims. ▸ Identify the roles involved in the software team at each stage of the software development lifecycle. ▸ Define objectives that each SDLC stage aims to achieve and how security relates to these aims. ▸ Understand common metrics used to measure the success of each SDLC stage.
Choosing processes, controls, and technologies.	▸ Identify typical software security tools that can be applied in each SDLC stage. ▸ Identify typical software security processes that can be applied to each stage of the SDLC. ▸ For each tool and process, define success criteria for its implementation and usage. ▸ Understand and define the likely impacts of introducing a new tool or process to an existing SDLC.

SS-CSDP-REQ: Identifying and capturing security requirements for software

MODULE	MODULE DESCRIPTION
Introduction to security requirements.	▸ Understand the role of security requirements alongside traditional software requirements. ▸ Identify the key owners for security requirements and how they relate to our roles and responsibilities in the software team.

SS-CSDP-REQ: Identifying and capturing security requirements for software (cont)

MODULE	MODULE DESCRIPTION
Identifying common security requirements.	▸ Identify common patterns in software that will require specific security requirements and considerations. ▸ Security requirements as functional requirements. ▸ Security requirements as non-functional requirements.
Using the OWASP Application Security Verification Standard (ASVS) to identify security requirements.	▸ Understand the OWASP Application Security Verification Standard and its purpose. ▸ Understand the domains and levels of the OWASP ASVS and how they are structured. ▸ Assess a project or application against ASVS levels. ▸ Apply the ASVS requirements to your projects. ▸ Link ASVS requirements to your documented system requirements.
Identifying, documenting and prioritizing security requirements.	▸ The relationship between business logic and security requirements. ▸ Capturing requirements across software components. ▸ Capturing requirements in supporting business processes. ▸ The importance of documenting security requirements. ▸ How to document security requirements. ▸ Common issues with documenting security requirements.

SS-CSDP-VULN: Finding and fixing common application security vulnerabilities

MODULE	MODULE DESCRIPTION
Understanding application security vulnerabilities and how to find them.	▸ Understand what a software security vulnerability is and why they are important to securing our systems. ▸ Identify sources of common vulnerabilities to use as references including OWASP Top 10, CWE and NIST NVD.

SS-CSDP-VULN: Finding and fixing common application security vulnerabilities (cont)

MODULE	MODULE DESCRIPTION
Applying techniques to reduce or remove common application security vulnerabilities.	▸ Understand and Identify the causes of common application vulnerability classes as featured in the OWASP Top 10. ▸ Identify common application vulnerabilities using code/peer review processes. ▸ Adapt software development approaches to avoid common application vulnerability classes as featured in the OWASP Top 10.
Choosing appropriate techniques with awareness of their impact on other software quality metrics.	▸ Understand common software quality metrics and how they are used. ▸ Analyze the impact of a suggested security control in relation to other software quality metrics.

SS-CSDP-TEST: Testing for software security vulnerabilities

MODULE	MODULE DESCRIPTION
Understanding common types of specialist security testing.	▸ Understand the purpose of penetration testing, vulnerability assessment, and red teaming in the context of software testing. ▸ Understand typical results from external security testing. ▸ Communicate external testing results to your project and prioritize any actions required. ▸ Understand common misconceptions about external security testing.

SS-CSDP-TEST: Testing for software security vulnerabilities (cont)

MODULE	MODULE DESCRIPTION
Designing security tests and test cases.	▸ Understanding the difference between technical security tests (control validation) and business logic security tests. ▸ Designing test cases for technical security tests (control validation). ▸ Designing test cases for business logic security risks. ▸ Testing security controls as part of manual testing. ▸ Writing automated security tests.
Managing the security testing process.	▸ Scheduling security testing as part of testing processes. ▸ Reusing technical security tests and creating common test libraries. ▸ Managing and prioritizing test results.

SS-CSDP-3RD: Safely using third-party software components

MODULE	MODULE DESCRIPTION
Understanding the software supply chain.	▸ Understand what a supply chain is and how this concept relates to software. ▸ Understand how supply chain attacks work and the risks they pose to software applications. ▸ Identify examples of software supply chain attacks and how they impacted systems, data, and people.
Identifying and documenting 3rd party components in your systems.	▸ Identify the common types of components in your software supply chain. ▸ Understand options for recording third-party components. ▸ Understand the function of a “Software Bill of Materials” and how to create one.


SafeStack

Certified Secure Development Leader

The CSDL certification equips software leaders to drive software security change within their teams and build secure-by-design products.

There is a critical shortage of secure development specialists. The CSDL provides a structured program for becoming an application security leader.

Learning Outcomes

Develop a culture of application security within your organization

This course and resources enable you to reach certification standards while building an application security program in your current organization.

Take a security-by-design approach to secure software

Reduce your team's and organization's risk by implementing security from the start of your SDLC and weave it as a continuous theme throughout your software's life.

Create certainty and direction, and measure progress

Implement processes and metrics that allow you to move your team's maturity measurably, in alignment with globally recognized standards such as OWASP SAMM, NIST SSDF, BSIMM, and ISO 27001.

Career Pathways

There has never been more need for secure software built maturely and measurably. Chart your course as a leader in this field and explore pathways to becoming:

Application Security Manager

Leading application security teams to enable secure development practices through large, complex environments.

Product Security Leader

Leading the design of software products that integrate security throughout. Keep your organization's data, people, and systems safe.

Senior Application Security Engineer

Leading the security efforts in your team or organization in a hands-on role, coaching others, and setting direction.

Syllabus:
Certified Secure Development Leader

Secure software doesn't just happen; no magic tool can bring it to your team, organization, or software development lifecycle. Secure development needs a leader and champion within an organization to craft a change program that improves our technology practices and security culture.

The **Certified Secure Development Leader (CSDL)** exam provides a recognized framework to assess your skills as a secure development leader and your ability to design, implement, measure, and manage an application security program across your organization.

SS-CSDL-ROLE: Understanding the role of the secure development leader

MODULE	MODULE DESCRIPTION
Defining the need for leadership in secure development.	▸ Explain the role leadership has to play in the creation of secure software. ▸ Define the role and responsibilities of a secure development leader. ▸ Understand the impact of poor/missing leadership on software security outcomes.
Understanding the impact of security on software.	▸ Understand why software may be vulnerable and how these vulnerabilities are found. ▸ Understand and explain the impacts of security vulnerabilities in terms of harm to an organization's operations, reputation, or success. ▸ Understand and explain the impacts of security vulnerabilities on developer toil, workflow disruption, and innovation. ▸ Define a secure development leader's role in minimizing or removing these impacts.

SS-CSDL-MEAS: Measuring security maturity on an application and lifecycle basis

MODULE	MODULE DESCRIPTION
Defining the stages of the software development lifecycle (SDLC) and how security can be applied.	▸ Identify the stages of common software development lifecycles and their aims. ▸ Identify the roles involved in the software team at each stage of the software development lifecycle. ▸ Define objectives that each SDLC stage aims to achieve and how security relates to these aims. ▸ Understand common metrics used to measure the success of each SDLC stage.
Measuring the maturity of a software application or product.	▸ Understand why we measure the maturity of a specific application and how that measurement is carried out. ▸ Understand the parts of an application that can be measured and assessed. ▸ Identify common frameworks such as OWASP ASVS and NIST 800-53 to measure software application maturity. ▸ Understand the challenges and limitations of measuring maturity for a specific software or application.
Measuring the security maturity of a software development lifecycle.	▸ Understand why we measure the maturity of security within a specific software development lifecycle and how this measurement is carried out. ▸ Identify common frameworks such as OWASP SAMM and NIST SSDF to measure the maturity of a software development lifecycle or process. ▸ Understand the challenges and limitations of measuring maturity for a software development lifecycle.

SS-CSDL-MEAS: Measuring security maturity on an application and lifecycle basis (cont)

MODULE	MODULE DESCRIPTION
Measuring your current security maturity.	▸ Identify factors that will guide measuring software security, including culture, budget, operating cadence, and rate of technical change. ▸ Communicate your chosen approach and create buy-in and engagement. ▸ Choose an appropriate maturity approach for your organization's size, culture, and context. ▸ Implement your measurement program and metrics and schedule regular reviews.

SS-CSDL-PROG: Designing an application security program

MODULE	MODULE DESCRIPTION
Understanding the purpose and structure of an application security program.	▸ Explain what an application security program is and what it aims to achieve. ▸ Identify typical elements of an application security program. ▸ Understand the common motivations for designing and implementing an application security program.
Setting program aims and objectives.	▸ Define the aims for your application security program in terms of what you hope to achieve. ▸ Identify concrete or numerical metrics against which these aims can be assessed.

SS-CSDL-PROG: Designing an application security program (cont)

MODULE	MODULE DESCRIPTION
Creating a program of work and prioritizing efforts.	▸ Create an outline application security program that defines the initiatives you intend to put in place against the objectives you aim to achieve. ▸ Assess each initiative to identify each item's likely cost (people, technology, time). ▸ Use this assessment data to prioritize program initiatives, including your current budget, team size, and likely impact on overall application security goals/objectives. ▸ Arrange your prioritized initiatives into a 1-3 year program of work that can be communicated to your team, leadership, and other relevant stakeholders.
Measuring and reviewing progress.	▸ Define goals and metrics for each program initiative and ensure relevant data is available to measure them. ▸ Ensure data sources used for metrics are consistent across teams and systems for direct comparison. ▸ Define a reporting cadence for the program, a report format, and a channel in which they should be shared. ▸ Create a feedback mechanism to invite software team members to honestly and actively comment on the effectiveness and challenges of new initiatives. ▸ Schedule regular review sessions to evaluate the program, its progress against objectives, and any feedback.

SS-CSDL-CHAM: Identifying and enabling security advocates and champions

MODULE	MODULE DESCRIPTION
Understanding the role of the security champion or advocate in application security.	▸ Understand why the role of security champion/advocate exists and the need it attempts to solve. ▸ Explain why security champions provide an alternative to growing a specialized application security team. ▸ List the benefits of establishing security champions within an organization and their impact on application security from a technology, people, and process perspective.
Defining the Security Champion role.	▸ Identify the attributes that make an effective security champion. ▸ Define a set of responsibilities for this role and map how this relates to existing roles/responsibilities in the software team. ▸ Write a role definition for a security champion that can be used to communicate their function and part of internal recruitment.
Setting objectives and monitoring progress.	▸ Establish a backlog or task recording process that will capture software security tasks and issues in a central, visible location. ▸ Define a schedule of activities for your champions that includes knowledge sharing and working on specific software security backlog items. ▸ Define metrics for the success of your security champions on an individual and program level. ▸ Set up mechanisms to monitor these metrics and schedule regular reviews.
Incentivizing action and responding to issues.	▸ Understand the mechanisms available to reward or recognize security champions for their actions. ▸ Define and document your expectations for security champions and how rewards and incentives will operate. ▸ Create feedback mechanisms for your security champions to raise issues and for your wider team to raise the problems about the champions and their surrounding program.

SS-CSDL-CHAM: Identifying and enabling security advocates and champions

MODULE	MODULE DESCRIPTION
Recruiting champions to your security program.	▸ Promote your champions program to your software teams, including the roles of champions and your vision for the program. ▸ Assess applications to ensure they have the right mix of skills, experience, and availability for the role. ▸ Promote your selected champions across your teams to ensure they are easy to identify, and the scope of their role is understood.

SS-CSDL-DIV: Managing software security across diverse team and process maturities

MODULE	MODULE DESCRIPTION
Documenting your software security landscape.	▸ Understand the importance of mapping your software security landscape and its benefits. ▸ Map your software environment from an infrastructure perspective. ▸ Map your product architecture from an application perspective. ▸ Map your business process architecture, encompassing software components and related business and operational processes. ▸ Identify and overcome common challenges with mapping your software landscape.

SS-CSDL-DIV: Managing software security across diverse team and process maturities (cont)

MODULE	MODULE DESCRIPTION
Responding to change.	▸ Identify sources of change for your organization, including operational, technological, regulatory, and commercial. ▸ Analyze how these changes can impact your organization and the software it produces. ▸ Analyze and identify changes that would impact the security requirements of your software systems. ▸ Analyze and identify changes impacting the secure operations of software supported business processes. ▸ Define a process to regularly review changes and respond to any resulting security impacts.
Understanding and securing the software supply chain.	▸ Understand what a supply chain is and how this concept relates to software. ▸ Understand how supply chain attacks work and the risks they pose to software applications. ▸ Identify examples of software supply chain attacks and how they impacted systems, data, and people. ▸ Define processes to support the secure selection of third-party technologies and software components. ▸ Define processes to support the ongoing monitoring and management of third-party technologies. ▸ Develop processes for where third-party components cannot be updated or patched. ▸ Implement processes and standards for documenting your organization's software supply chain and maintaining this document over time.
Applying a security program to self-governing teams.	▸ Understand the concept of self-governing teams in software and how that changes their behaviors and software development lifecycle. ▸ Identify and define strategies to overcome the challenges of self-governing teams to ensure equitable outcomes across them.

SS-CSDL-DIV: Managing software security across diverse team and process maturities (cont)

MODULE	MODULE DESCRIPTION
Securing legacy projects and systems.	▸ Understand the place of legacy systems in our software architectures and how they impact our security approaches. ▸ Identify the challenges with applying modern security approaches to legacy components. ▸ Understand and define compensating controls and approaches where modern approaches cannot be applied.

SS-CSDL-COMM: Communicating with software security stakeholders

MODULE	MODULE DESCRIPTION
Identifying stakeholders in your software teams (and associated software security efforts).	▸ Identify common stakeholders with an interest in software delivery for your organization. ▸ Identify each stakeholder group's role in the organization's operation, motivations, measures of success, and primary concerns.
Putting your application security program in the context of wider organization goals.	▸ Identify the primary goals and metrics for your wider organization, how they are measured, and the impact meeting or missing these goals has on the organization. ▸ Identify how these metrics apply to the software team, their metrics, and their expectations. ▸ Understand the relationship between your application security program (and its initiatives) and the metrics for your software team. ▸ Analyze the impact of causing key software team metrics to decrease or underperform. ▸ Identify how to support team and operational goals with your application security program. ▸ Communicate your application security program in the context of these combined metrics and goals, the program's impact, and how this work will support them.

SS-CSDL-COMM: Communicating with software security stakeholders (cont)

MODULE	MODULE DESCRIPTION
Communicating with senior leadership and executive teams.	▸ Identify the communications channels senior leadership, executive teams, and stakeholders use. ▸ Identify the preferred communication style of each group and their data/information needs. ▸ Document the reporting and information needs for each group, including the preferred channel, format, and any details or linguistic needs.
Choosing when and how to escalate software security issues.	▸ Map your organization's data and escalation pathway - how does important information flow upwards to the executive and senior stakeholders? ▸ Identify and document the profile of issues and incidents that need to be escalated. Relate this to your organization's software/security vulnerability classification system. ▸ Once notified, understand the impact of escalating security incidents on the board and senior executives and their obligations. ▸ Create an incident escalation plan that defines how and when to escalate software security issues and incidents.
Negotiating for budget and resources for application security initiatives.	▸ Estimating budget for application security initiatives, including the costs of technology, people, and processes. ▸ Document your budget against your program goals. ▸ Prioritize spending across initiatives when funding for all initiatives is not available.

SS-CSDL-COMM: Communicating with software security stakeholders (cont)

MODULE	MODULE DESCRIPTION
Communicating progress and telling the story of your application security program.	▸ Understand the importance of storytelling in application security. ▸ Use a narrative focus to create a shared vision and increase collaboration. ▸ Explain your progress numerically and in ways to suit different audiences. ▸ Celebrate application security success stories as a tool for engagement. ▸ Thoughtfully share secure development challenges, incidents and issues as a tool for grounding actions and linking them to consequences.

SS-CSDL-INCI: Planning for and responding to software security incidents

MODULE	MODULE DESCRIPTION
Understanding the incident response process.	▸ Understand the common stages of incident response processes, including identification, verification, containment, and remediation. ▸ Understand the aim of each stage and what activities happen within them. ▸ Identify the types of security incidents that might occur and which may be relevant (or include/require) the software team. ▸ Identify how the software team supports the incident response process and the importance of this role.

SS-CSDL-INCI: Planning for and responding to software security incidents (cont)

MODULE	MODULE DESCRIPTION
Creating a software incident response plan.	▸ Understand the typical structure and contents of an incident response plan. ▸ Identify and understand any whole organization's incident response plans if they exist. ▸ Create a software-specific incident response plan that outlines the high-level or non-incident-specific processes the software team should follow when investigating and responding to security incidents.
Planning for common software security incidents.	▸ Identify common software attack patterns and security incident types that may require incident response. ▸ Build a playbook outlining the scenario-specific steps to investigate and respond to each incident type.
Testing your software incident response plan.	▸ Identify the people and roles that should be included in incident response planning and testing. ▸ Schedule regular incident response plan tests and ensure the correct people and roles attend. ▸ Conduct a plan walkthrough for a specific scenario during each testing session to ensure the associated plans and playbooks are accurate and useful. ▸ Feedback and lessons learned from incident response plan testing are applied to the relevant systems, plans, and playbooks.
Conducting post-incident reviews.	▸ Understand the importance and function of post-incident reviews. ▸ Define a structure for post-incident reviews that focuses on capturing lessons learned and improvements to be made. ▸ Understand and adopt a culture of "blameless" post-incident reviews and how this impacts the review and overall security culture. ▸ Document findings and actions from post-incident reviews and prioritize them so that they are addressed promptly.

Customer Testimonials

nab

"We were able to establish security as ongoing practice and rise awareness within the entire development team. It's easy for developers to engage with and they don't need to spend much time on it, just 15 min is enough."

annalise.ai

"We have given our dev teams easy to digest training,and safe coding skills, which has given our exec team confidence that we can protect our business and customer data.

In turn, this has given our customers confidence that our software will protect them."

prospa

"As well as the knowledge and concepts, we're training developers how to think like attackers. If developers are learning how to break things, they'll subsequently learn how they can fix things."

fastly

"With so many of our company's customers expecting a high standard of security and compliance, SafeStack's platform ensures my team are trained in best practice at every level from junior through to senior software contributors."

Southern Cross Health Insurance

"We're attempting to upskill a lot of developers to develop secure microservices, so it's a godsend being able to send people to the training to explain why we ask for some of the things that we do. "

Auror.

"Sometimes security training is a matter of ticking a box, but with Safestack you genuinely feel like you're levelling up your teams skills, which puts you in the best place to defend from real attacks, no matter what's "on paper"

For more information or to book a demo:

www.safestack.io

hello@safestack.io