ISC2 CSSLP

OUR LEARNING EXCLUSIVE

- Custom exam prep software and materials
- Exam delivery in the classroom, with a **98% success rate**
- Course-specific thinQtank® Learning publications to keep the learning engaging
- Extended hours of training, including immersive hands-on exercises
- **We do not teach the test.** We always deliver valuable hands-on experience
- All reading material and study guides sent to you when you register
- Every course taught by certified instructors

TRAINING CAMP FORMAT

thinQtank® Learning delivers this course as an **instructor-led training camp**, with the exam sat in the classroom as part of the camp. Camps are scheduled by cohort rather than to a fixed calendar, and the balance of lecture and lab work is set to suit the group and the delivery method. Contact us for the schedule covering your dates.

PREREQUISITES

- **Four years** of cumulative, full-time experience in one or more of the eight domains — the highest bar of any course we run
- **A degree takes a year off it.** A post-secondary degree in computer science, IT or a related field satisfies one of the four
- **Part-time and internships count.** 20-34 hours a week accrues; 1,040 hours is six months of full-time credit. Internships count with documentation
- **And you can sit the exam without any of it.** Pass, and you are an Associate of ISC2 with five years to earn the four

EXAM

ISC2 CSSLP — Certified Secure Software Lifecycle Professional

Four years — but read the whole rule. A relevant degree covers one of them, part-time work and documented internships accrue, and you can sit the exam with none of it: pass and you are an Associate of ISC2 with five years to earn the four. Far more people qualify than the headline suggests.

Eight domains. Passing score **700 on a scale of 1000**. ISC2 does not publish an item count, an exam duration or domain weightings for CSSLP, so we do not quote any.

Accredited to ISO/IEC 17024 by ANAB, and approved by the US Department of Defense under DoDM 8140.03.

DELIVERY METHODS

- Instructor-led training
- Immersive live-online training
- On-site and custom delivery

COURSE OVERVIEW

thinQtank® Learning offers an industry-unique training camp in which students earn the **ISC2 CSSLP** — Certified Secure Software Lifecycle Professional. As with all our ISC2 training experiences, the exam is delivered in the classroom.

CSSLP is the credential for building security in rather than bolting it on. Every phase of the software lifecycle gets its own domain — concepts, lifecycle management, requirements, architecture and design, implementation, testing, deployment and operations, and the supply chain the whole thing now depends on.

It is the rare security certification aimed at the people who write and ship the software, not the people who audit it afterward. The **voucher is available** at $599 USD.

COURSE OBJECTIVES

Primary course objectives:

- Apply core secure software concepts: confidentiality, integrity, availability and least privilege
- Apply secure design principles and recognize where each one fails in practice
- Manage a secure software lifecycle, and embed security into the methodology in use
- Define security and privacy requirements, including abuse and misuse cases
- Perform threat modeling, and turn its output into requirements rather than a document
- Design a secure architecture, and select controls that match the threat model
- Model and validate a design against the security requirements it must meet
- Implement securely: input validation, output encoding, error handling and cryptography
- Recognize and avoid common vulnerabilities in code you write and code you review
- Perform secure code review, and use static and dynamic analysis deliberately
- Plan and run security testing, including negative and abuse-case tests
- Verify and validate the software before it ships
- Secure deployment: configuration, secrets management, and hardening the release
- Operate, monitor and maintain software securely, including patch and incident handling
- Manage the software supply chain: third-party components, provenance and risk
- Handle end of life and decommissioning without leaving data or access behind

WHAT YOU WILL BE ABLE TO DO

- Write security into requirements, where it is cheapest to fix
- Threat model a design and defend the decisions that come out of it
- Review code and tests for the failures that actually get exploited
- Answer for the third-party components your product ships with

TARGET AUDIENCE

- Software architects, engineers and developers who own security in their code
- Application security specialists and secure-code reviewers
- QA testers and penetration testers working against their own products
- DevSecOps engineers putting security into a pipeline rather than after it
- Security managers and IT directors accountable for what ships

EXCLUSIVE LEARNING PACKAGE

- Threat modeling and abuse-case workshops against real designs
- Secure code review exercises against the stack your team actually ships
- Pipeline exercises: putting security testing where it will be run
- CSSLP video course
- CSSLP reading plan and review guide
- CSSLP certification practice exam

COURSE MODULES

01 Secure Software Concepts
- Topic A: Core concepts — confidentiality, integrity, availability
- Topic B: Authentication, authorization and accountability
- Topic C: Secure design principles, and how each one fails
- Topic D: Risk management for software
- Topic E: Regulations, privacy and standards that bind the product

02 Secure Software Lifecycle Management
- Topic A: Security in the methodology you actually use
- Topic B: Roles, responsibilities and security champions
- Topic C: Security metrics, gates and decision points
- Topic D: Configuration and version management
- Topic E: Policy, standards and the documentation that survives an audit

03 Secure Software Requirements
- Topic A: Eliciting security and privacy requirements
- Topic B: Abuse cases, misuse cases and how to write them
- Topic C: Data classification and its consequences for the design
- Topic D: Requirements traceability
- Topic E: Turning compliance obligations into testable requirements

04 Secure Software Architecture and Design
- Topic A: Threat modeling, and what to do with the output
- Topic B: Architectural risk assessment
- Topic C: Selecting controls that match the threat model
- Topic D: Secure interface, API and data-flow design
- Topic E: Design review and validation

05 Secure Software Implementation
- Topic A: Input validation, output encoding and error handling
- Topic B: Using cryptography correctly, and the ways it is misused
- Topic C: Common vulnerabilities, and the code patterns that cause them
- Topic D: Secure code review, static and dynamic analysis
- Topic E: Secure build, and securing the pipeline itself

06 Secure Software Testing
- Topic A: Designing a security test strategy
- Topic B: Functional, negative and abuse-case testing
- Topic C: Penetration testing and fuzzing in context
- Topic D: Interpreting results, triage and remediation tracking
- Topic E: Verification and validation before release

07 Secure Deployment, Operations and Maintenance
- Topic A: Secure configuration and hardening at release
- Topic B: Secrets, keys and credential management
- Topic C: Monitoring, logging and detection for running software
- Topic D: Patch, vulnerability and incident management
- Topic E: End of life, decommissioning and data disposal

08 Secure Software Supply Chain
- Topic A: Third-party and open-source component risk
- Topic B: Software composition analysis and the bill of materials
- Topic C: Supplier assessment and contractual security requirements
- Topic D: Provenance, integrity and signing
- Topic E: Managing risk you inherit rather than create