CERT | Software Engineering Institute
Carnegie Mellon University

# CERT Secure Coding

## Develop and Deploy Error-Free Software

The most effective way for developers to improve software security is to eliminate vulnerabilities during development. The CERT Secure Coding Team devises programming techniques that help developers increase the security of their code and reduce its vulnerability to attack.

### Secure Coding Standards

We coordinate developing secure coding standards by working with researchers and software developers in the Secure Coding Wiki. More than 1,900 contributors and reviewers develop rules and recommendations for C, C++, Java, and Perl. We are also developing a standard for the Android platform. Visit securecoding.cert.org to explore the wiki.

We publish standards resulting from our work on our wiki. The 2016 editions of the *SEI CERT C Coding Standard* and *SEI CERT C++ Coding Standard* identify the root causes of today's most widespread software vulnerabilities, show how they can be exploited, review the potential consequences, and present secure alternatives. Both standards, available for free download, can help you develop more secure software systems written in C and C++. Download the 2016 editions at resources.sei.cmu.edu/library/asset-view.cfm?assetID=494934.

### What We Do

- **Research flaws in code and develop solutions to identify and prevent them**
- **Work with software developers to establish coding standards**
- **Use SCALe audits to evaluate codebases for their conformance to coding standards**
- **Provide training that enables software developers to build secure and reliable products**
- **Contribute to the development of international coding standards**
- **Help organizations set up and/or optimize their code analysis framework**

## SCALe

The Source Code Analysis Laboratory (SCALe) is a conformance testing process used to examine codebases to identify known security-related coding errors. You can submit your code for a SCALe audit to identify common errors in the codebase and get advice for addressing them. Conforming systems can promote their software using the CERT SCALe seal to indicate that the codebase analyzed conforms to applicable secure coding standards. We published sample SCALe audit results and SCALe videos that demonstrate how SCALe works. For more information, see cert.org/secure-coding/products-services/scale.cfm.

## API Usability and Security

As part of our collaboration with the Human-Computer Interaction Institute at Carnegie Mellon University, we study the interactions between making an application program interface (API) more usable and making it more secure. In this research, we are gathering empirical evidence about the security impacts of API design.

## Consulting Services for Optimizing Code Analysis Frameworks

We provide advice on how to optimize your system for analyzing code, using multiple analyzers to discover more code flaws, and using intelligent methods to optimize human efforts working with diagnostics. These consulting services can help your organization evaluate and improve its software security system. Contact us to request this consulting service.

## Our Certificate Program

Our **Secure Coding Professional Certificates** for C, C++, and Java help you increase the security of your software and reduce vulnerabilities in the programs you develop. Earning one of these certificates demonstrates to your managers and customers that you're disciplined and serious about your coding practices. Learn more at cert.org/go/secure-coding/.

## International Standards Development

We participate in the development of international standards for programming languages to improve the overall safety and security of these languages. We send our technical experts to ISO/IEC working group meetings for C, C++, and programming language vulnerabilities. Learn more at cert.org/secure-coding/standards/.

## Secure Coding Tools

The tools we develop are used in SCALe auditing but can also help software developers reduce the number of vulnerabilities in their code. All of our tools are available for free download:

**Clang Thread Safety Analysis**, collaboratively developed with Google, uses annotations to declare and enforce thread safety policies in C and C++ programs.

**DidFail** uses static analysis to detect potential leaks of sensitive information in Android app sets.

**Rosecheckers** performs static analysis on C/C++ source files to enforce the rules in the CERT C Coding Standard.

**Compiler-Enforced Buffer Overflow Elimination** prevents buffer overflows in multithreaded code.

## Our Work in Secure Coding

We are constantly searching for ways to address commonly made coding errors. Such research results in approaches and tools that help software developers produce secure products.

We share our secure coding research findings in our blog posts, webinars, newsletter, and other publications in our digital library.

We provide up-to-date training in secure coding practices, including a four-day Secure Coding in C and C++ course and an onsite Java Workshop.

Learn more at cert.org/secure-coding/research/.

## About Us

For nearly 30 years, the CERT Division of the SEI at Carnegie Mellon University has been a leader in cybersecurity. Originally focused on incident response, we have expanded into cybersecurity areas such as network situational awareness, malicious code analysis, secure coding, resilience management, insider threats, digital investigations and intelligence, workforce development, DevOps, forensics, software assurance, vulnerability discovery and analysis, and risk management.



## Contact Us

Software Engineering Institute
Carnegie Mellon University
4500 Fifth Avenue
Pittsburgh, PA 15213-2612

**Phone:** 412-268-5800
**Toll Free:** 1-888-201-4479
**Email:** info@sei.cmu.edu
**Web:** www.sei.cmu.edu