

July 2013 Edition

[News](#)

[Language Standard Updates](#)

[Upcoming Events / Training](#)

[Our People](#)

[Secure Coding Resources](#)

News

Welcome to the inaugural monthly newsletter for the CERT secure coding standards. Our goal is to provide you with timely information concerning updates to the CERT secure coding standards and to make you aware of other interesting news and events related to secure coding. Because this is the first edition of the newsletter, it will probably be a bit longer than most. If you wish to unsubscribe, just use the [SafeUnsubscribe](#) link at the end of this newsletter.

The CERT Division's Secure Coding Initiative has been developing secure coding standards since 2006. InformIT recently published an article called [C Secure Coding Rules: Past, Present, and Future](#), which focuses primarily on the history and future of the [CERT C Secure Coding Standard](#).

Industry is widely adopting the CERT secure coding standards. Representatives of many organizations have visited our wiki. At October 2011's annual SecCon conference, Cisco Systems Inc. announced its adoption of the CERT C Secure Coding Standard as a baseline programming standard in its product development. Recently, Oracle has integrated all of the CERT secure coding standards into their existing Secure Coding Standards. This adoption is the most recent step of a long collaboration: the CERT Division and Oracle previously collaborated on [The CERT Oracle Secure Coding Standard for Java](#).

Given the fluid nature of languages and coding standards, guidelines can change between publication cycles. We are codifying the deprecation rules and are looking for feedback from users and vendors to ensure the deprecation process is as painless as possible. Please provide feedback on the [Deprecations](#) page.

In addition to working on secure coding standards for C, C++, Java, and Perl on the public wiki, we have also begun efforts to create secure coding standards for various other languages, including Ada, C#, Fortran, Python, JavaScript, and SPARK. We will share these with the public once they move beyond the embarrassing stage. If you would like to contribute to the development of these standards, please contact us at secure-coding@cert.org.

New Edition of Secure Coding in C and C++ Addresses Code Changes and New Threats



To address advances and changes in the C and C++ coding languages, and to address new threats faced by programmers working in these languages, Software Engineering Institute (SEI) researcher Robert C. Seacord has authored *Secure Coding in C and C++, Second Edition*. Seacord, a senior member of the SEI technical staff and technical manager of the CERT Secure Coding Initiative, also authored the original 2005 edition. The book has been published by Addison-Wesley Professional as part of its SEI Series.

SAVE 35%+ FREE Shipping within the U.S., just enter the discount code **SEACORD** during checkout.

Language Standard Updates

CERT C Secure Coding Standard

Editors: Martin Sebor (Cisco Systems), Aaron Ballman (SEI)

John Benito from Blue Pilot has been updating the CERT C Secure Coding Standard for C11, starting by changing references to TR 24731-1 to C11 Annex K.

Added

[EXP42-C. Do not compare padding data](#)

[API01-C: Avoid laying out strings in memory directly before sensitive data](#)

Changed

[DCL05-C. Use typedefs of non-pointer types only](#) Updated the function pointer type definition

code example to be a function type definition instead; added more examples.

[MEM02-C. Immediately cast the result of a memory allocation function call into a pointer to the allocated type](#) Added an exception to the recommendation for compiling in C90 on 64-bit platforms due to implicit function declarations possibly causing pointer truncation/sign extension bugs.

Removed

None

Deprecated

None

Unresolved Issues

There is a conflict between [PRE03-C. Prefer typedefs to defines for encoding types](#) and [DCL05-C. Use typedefs of non-pointer types only](#) that needs to be resolved. PRE03-C has a compelling code example that demonstrates the dangers of using macros to define types, but the solution involves using a typedef for a pointer type, which contradicts the recommendation of DCL05-C. Comments, suggestions, or possible solutions are welcome!

CERT C++ Secure Coding Standard

Editors: Martin Sebor (Cisco Systems), Aaron Ballman (SEI)

No C++ rules were added, removed, deprecated, or substantively changed last month.

CERT Oracle Secure Coding Standard for Java

Editors: Adam O'Brien (Oracle), David Svoboda (SEI)

Added

The following rules were added as part of a new appendix of rules that are specific to the development of Java Apps for the Android platform:

[DRD00-J. Do not store sensitive information on external storage \(SD card\)](#)

[DRD01-J. Limit the accessibility to your sensitive content provider](#)

[DRD02-J. The WebView class allows local resource access through file scheme](#)

[DRD04-J. Do not send sensitive information to log output](#)

Changed

[DCL00-J. Prevent class initialization cycles](#) A new pair of code examples were given. They show code that seems to work properly but still contains an initialization cycle, so it doesn't guarantee proper initialization. The accompanying compliant example avoids the initialization cycle.

Removed

None

Deprecated

None

CERT Perl Secure Coding Standard

Editor: David Svoboda (SEI)

No Perl rules were added, removed, deprecated, or changed last month.

Upcoming Events / Training

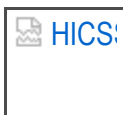


Members of the Secure Coding Initiative will be giving three presentations at the [JavaOne 2013 conference](#)

Session ID: [CON6396](#) Don't Be Pwned: A Very Short Course on Secure Programming in Java - Dean Sutherland and Robert Seacord

Session ID: [CON3122](#) Anatomy of a Java Zero-Day Exploit - David Svoboda

Session ID: [TUT5599](#) The Java Security Architecture: How and Why - David Svoboda



Mini-Track Announcement | Hawaii International Conference on Systems Sciences

Software Security for Mobile Platforms

January 6-9, 2014

Hilton Waikoloa, Big Island, Hawaii


Secure Coding in C and C++ Course

This four-day course provides a detailed explanation of common programming errors in C and C++ and describes how these errors can lead to code that is vulnerable to exploitation. The course concentrates on security issues intrinsic to the C and C++ programming languages and associated libraries. This course may be offered by special arrangement at customer sites.

Our People

Each month we will highlight staff members behind our secure coding research. This month we feature Robert Seacord and Martin Sebor.



Robert C. Seacord is a senior vulnerability analyst in the CERT  Program at the Software Engineering Institute (SEI) in Pittsburgh, PA where he leads the Secure Coding Initiative. Robert is the author of *The CERT C Secure Coding Standard* (Addison-Wesley, 2008) and *Secure Coding in C and C++* (Addison-Wesley, 2002) as well as co-author of two other books. Robert is an adjunct professor at Carnegie Mellon University and a technical expert for

ISO/IEC JTC1/SC22/WG14, the international standardization working group for the programming language C.



Martin Sebor is a technical leader in the C and C++ compiler tool chain group in the Network Operating Systems Group at Cisco Systems, Inc., where he works on compilers and related development tools as well as the Cisco networking operating system IOS. Among Martin's responsibilities is leading the



development and deployment of Cisco Secure Coding Standards. Martin's expertise includes the C and C++ languages and development tools, and the POSIX standard. Martin is Cisco's representative to the C and C international standards committees (PL22.11 and PL22.16 subgroups of the INCITS technical committee for Programming Languages, PL22).

Secure Coding Resources



Watch Robert Seacord discuss avoiding future security incidents from the recent [CERT Virtual Event-- A Discussion with CERT Experts: Constructing a Secure Cyber Future](#)



Listen to Martin Sebor and Julia Allen discuss [Cisco's Adoption of CERT Secure Coding Standards](#)



Read "Silent Elimination of Bounds Checks" by Robert Seacord on the [InformIT website](#).

Join the SEI CERT Secure Coding Community

