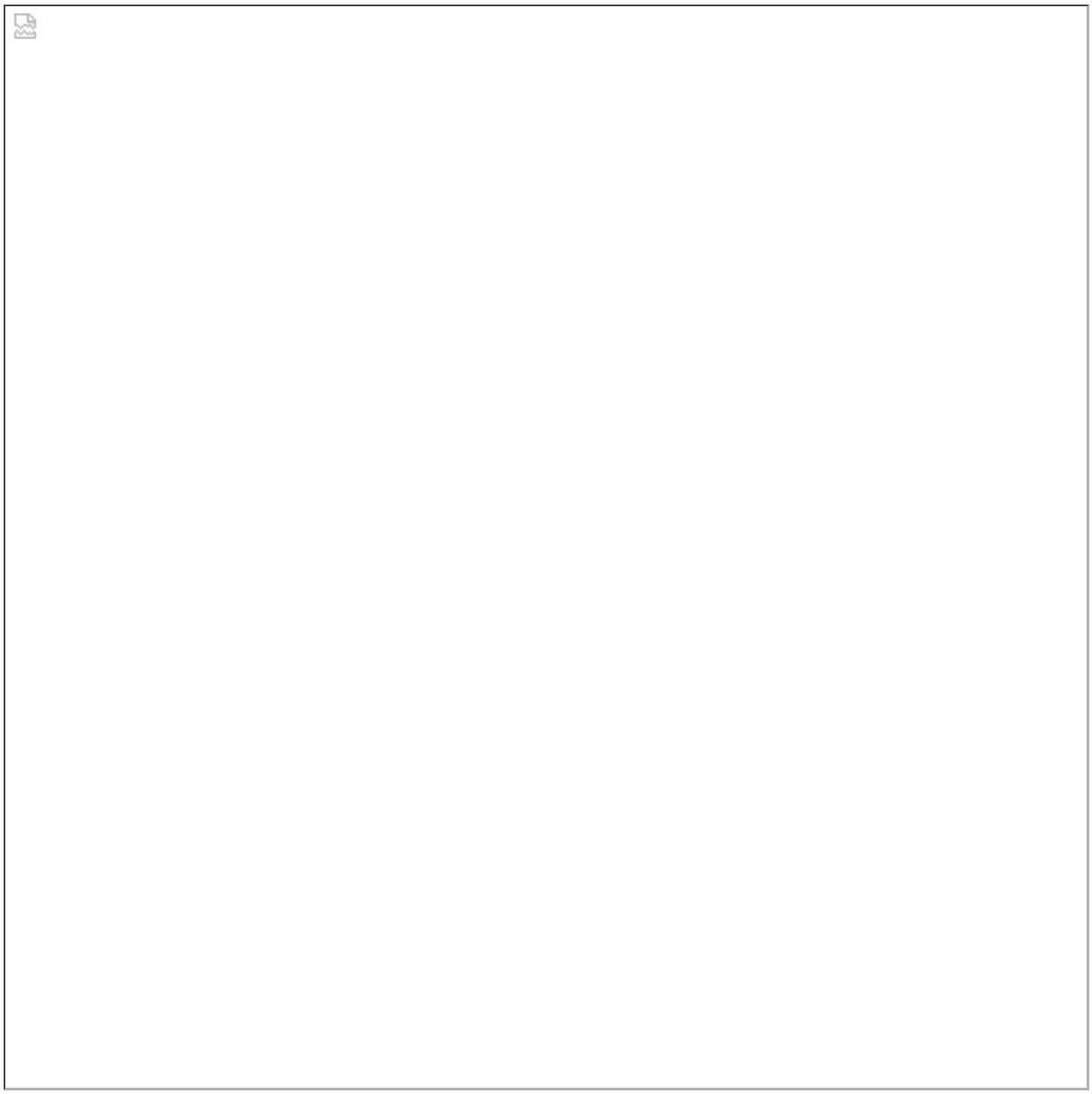

September 2013 Edition

News

This is the third monthly Secure Coding eNewsletter. 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. If you wish to unsubscribe, just use the SafeUnsubscribe link at the end of this newsletter.

The fall semester is upon us here at Carnegie Mellon University as students return to campus.

The new semester always brings new energy to the campus, even for staff members with no teaching responsibilities.

Our major push for September is to complete the CERT C Secure Coding Standard for C11. To do so, we need your help in reviewing the content and submitting comments on the wiki or by email. Please provide your comments by September 13, if possible, so that we will have time to incorporate them before publication.

A new book, *Java Coding Guidelines: 75 Recommendations for Reliable and Secure Programs*, is now available. This book was authored by a team of current and former CERT employees and visiting scientists, including Fred Long, Dhruv Mohindra, Robert C. Seacord, Dean F. Sutherland, and David Svoboda. This same group of authors published *The CERT Oracle Secure Coding Standard for Java* in September 2011. The CERT website offers a support site for the book.

Another area experiencing increased activity is the development of additional rules for the Android (DRD) appendix to *The CERT Oracle Secure Coding Standard for Java*. This appendix includes rules that are specific to developing Java apps for the Android platform. This is a good place to give a shout out to our colleagues at JPCERT who have led the way in the development of these rules, including Yozo TODA and Masaki Kubo.

You have probably noticed that we have upgraded the wiki software to Confluence v5.1. We plan to upgrade the look and feel of the wiki, but in the meantime, if you are wondering where the dashboard went, you'll find it here: https://securecoding.cert.org/confluence/dashboard.action.

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

Language Standards Updates

CERT C Secure Coding Standard

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

Standards Added

WIN03-C. Understand HANDLE inheritance
Discusses `HANDLE` inheritance on Windows when creating child processes
CON41-C. Do not join or detach a thread that was previously joined or detached
Points out undefined behavior when a detached thread is joined

Standards Changed

WIN guidelines renumbered to properly match rules versus recommendations conventions
FIO32-C. Do not perform operations on devices that are only appropriate for files
MEM06-C. Ensure that sensitive data is not written out to disk
MEM36-C. Check for alignment of memory space before calling realloc() function
CON00-C. Avoid race conditions with multiple threads
CON02-C. Do not use volatile as a synchronization primitive

CON34-C. Declare objects shared between threads with appropriate storage durations
Updated to have Windows-specific compliant solutions
Removed spurious calls to `thrd_exit()` from Concurrency (CON) code examples
FLP00-C. Understand the limitations of floating-point numbers
Updated to contain further information about inexact floating-point operations
DCL04-C. Do not declare more than one variable per declaration
Added an exception for multiple, simple declarations without initializers
MSC23-C. Beware of vendor-specific library and language differences was moved from rule to recommendation and renumbered accordingly

Standards Deprecated & Removed
FIO07-C. Prefer fseek() to rewind()
FIO12-C. Prefer setvbuf() to setbuf()
Deprecated and moved to the VOID because it is covered by ERR07-C. Prefer functions that support error checking over equivalent functions that don't
FLP06-C. Understand that floating-point arithmetic in C is inexact
Deprecated and moved to the VOID because it is covered by FLP00-C. Understand the limitations of floating-point numbers
FIO16-C was moved to the POS section and is now POS05-C. Limit access to files by creating a jail
STR33-C was deprecated moved to the VOID because it is covered by STR38-C. Do not use wide-char functions on narrow-char strings and vice versa and MEM35-C. Allocate sufficient memory for an object

Resolved Issues
After a short teleconference, we determined that EXP35-C is not obsolete with respect to changes to C11. C11 introduced the term *temporary lifetime*, which actually increases the lifetime of a temporary object. We will be updating EXP35-C to use the new language and also to explain any differences in behavior between C11 and C99.

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)

Standards Added
DRD03-J. Do not broadcast sensitive information using an implicit intent
Three other Android rules are under development

No Java rules were removed, deprecated or substantively changed last month.

CERT Perl Secure Coding Standard
Editor: David Svoboda (SEI)

Added `DATA` to the set of barewords that may be used as filehandles in FIO00-PL. Do not use bareword file handles

Modifed the examples in IDS31-PL. Do not use the two-argument form of open() to comply with

FIO00-PL

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

Several rules contain comments that have not yet been addressed. (It's been a busy month!) We will address them next month.

Upcoming Events and 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 presented by Dean Sutherland and Robert Seacord
Session ID: CON3122 Anatomy of a Java Zero-Day Exploit presented by David Svoboda
Session ID: TUT5599 The Java Security Architecture: How and Why presented by David Svoboda

Training Through CERT's
Secure Coding Initiative

Mini-Track Announcement | Hawaii International Conference on Systems Sciences
Software Security for Mobile Platforms
January 6-9, 2014
Hilton Waikoloa, Big Island, Hawaii

Our People

Each month we highlight staff members behind our secure coding research. This month we feature David Keaton.

David Keaton is the chairman of the ANSI C Committee, the U.S. segment of the international committee that standardizes the C programming language. He has been a voting member of the committee since 1990.

David has written compilers for everything from embedded systems to supercomputers. He has two patents related to compiler-assisted security mechanisms.

Secure Coding Resources

Watch Robert Seacord's presentation of Secure Coding

Listen to Robert Seacord and Julia Allen discuss Mainstreaming Secure Coding Practices

Read David Keaton's Blog post: Helping Developers Address Security with the CERT C Secure Coding Standard

Subscribe to Our eNewsletter

Join the SEI CERT Secure Coding Community