Carnegie Mellon University
Software Engineering Institute

Fall 2017 Edition

The end of the year is fast approaching. We have had a busy year conducting research, publishing information, and updating our standards. We are still looking for talented candidates to join our team and work with us to define secure coding practices and help organizations adopt them. We presented some of our work at the recent IEEE SecDev conference and the SEI Research Review. Links to materials are below if you missed it.

We wish everyone good tidings and a challenging and rewarding new year.

Bob Schiela

News

Open Positions in the SEI CERT Secure Coding Team

Join our Secure Coding team. If you have the right qualifications and are interested in researching and developing improvements to the state of the art and practice in secure coding, secure development, and software assurance, please consider applying for any of the following positions:

VA | Bedford, MA)
- Compiler Researcher
- Software Security Engineer
- Associate Software Security Analyst

Request for Contributors: Ada Guidelines

We are developing draft guidelines for Ada. If you would like to contribute to these guidelines by providing content or reviewing material, please send us a note at info@sei.cmu.edu.

Mapping Refresh: CERT Guidelines to MITRE CWEs

We are collaborating with MITRE to refresh the mappings between CERT guidelines and MITRE CWEs. Both MITRE and the CERT Division will publish the revised mappings.

Publications

We published the following blog posts:

- Automated Detection of Information Leaks in Mobile Devices (by Lori Flynn and Will Klieber)
- Inference of Memory Bounds: Preventing the Next Heartbleed (by Will Klieber and Will Snavely)

Will Klieber presented an SEI Cyber Minute on Preventing the Next Heartbleed.

Recent Events

At the Opportunities for Undergraduate Research in Computer Science (OurCS) event (October 20-22, 2017), Lori Flynn, William Klieber, Nancy Mead, and Carol Woody led a three-day workshop called "Develop Secure Smartphone Applications." Participants in the workshop developed original or enhanced existing Android coding rules on the CERT Android Coding Standard wiki. The OurCS umbrella workshop is organized by CMU's School of Computer Science and Women@SCS.

In September, at the IEEE CyberSecurity Development (SecDev) Conference, Lori Flynn, David Svoboda, and William Snavely led a hands-on tutorial called "Auditing Static Analysis Alerts Using a Lexicon & Rules." The tutorial included hands-on participant use of a virtual machine and printouts, plus an accompanying presentation by the tutorial leaders. The presentation is

called Challenges and Solutions for Automated Repair of C Code.

The SEI Research Review took place in Pittsburgh on October 17-18, 2017. Researchers presented results of their work, including research in secure coding and software assurance. We published the following artifacts from the SEI Research Review:

- Rapid Expansion of Classification Models to Prioritize Static Analysis Alerts for C (presentation by Lori Flynn)
- Rapid Expansion of Classification Models to Prioritize Static Analysis Alerts for C (poster by Lori Flynn)
- Inference of Memory Bounds (presentation by Will Klieber)
- Inference of Memory Bounds (poster by Will Klieber)

Upcoming Events

The SEI is hosting FloCon, a forum for exploring large-scale, next-generation data analytics in support of security operations, in Tucson, AZ on January 8-11, 2018.

SEI CERT Secure Coding Standard Updates

CERT C Coding Standard

Editor: David Svoboda, CERT Division of the Software Engineering Institute

Download the latest stable version.

No C rules were added or removed.

Changed

- Both compliant solutions in FLP34-C. Ensure that floating-point conversions are within range of the new type now gracefully handle NaN inputs, as well as 0 inputs.
- Parameters were reordered in the second compliant solution of EXP47-C. Do not call va_arg with an argument of the incorrect type.
- DCL37-C. Do not declare or define a reserved identifier now uses "include guard" rather than "header guard." Likewise, PRE06-C. Enclose header files in an include guard uses "include guard" rather than "inclusion guard."

CERT C++ Secure Coding Standard

Download the latest stable version.

No C++ rules were added or removed.

Changed

- PRE06-CPP. Enclose header files in an include guard uses "include guard" rather than "inclusion guard."

CERT Oracle Secure Coding Standard for Java

Editor: David Svoboda, CERT Division of the Software Engineering Institute

Download the latest stable version.

No Java rules were added or removed.

Changed

- OBJ05-J. Do not return references to private mutable class members and OBJ13-J. Ensure that references to mutable objects are not exposed now cross-reference each other to explain why they are similar but distinct rules.

CERT Secure Coding Standard for Android

Editor: Lori Flynn, CERT Division of the Software Engineering Institute

We added an Introduction to the Android standard, similar to the format for the other CERT Secure Coding Standards. Also, we developed the Guidelines for Wiki Contributors section, including creating a template for new coding guidelines.

No Android rules were removed.

Added

The following Android rules were added or substantively added to, as part of the OurCS workshop. (See Recent Events above.)

- DRD22. Do not cache sensitive information
- DRD25. Use constant-time encryption
- DRD07-X. Protect exported services with strong permissions
- DRD26-J. For OAuth, use a secure Android method to deliver access tokens
- DRD12. Do not trust data that is world writable
- DRD23-J. Do not use loopback when handling sensitive data

tokens

CERT Perl Secure Coding Standard

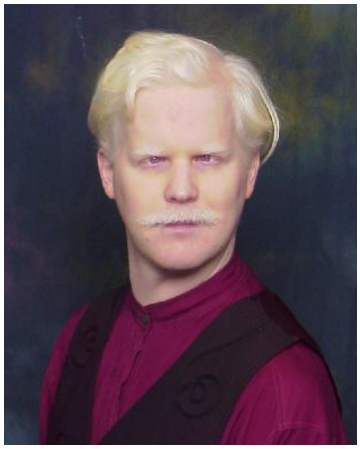
Editor: David Svoboda, CERT Division of the Software Engineering Institute

Download the latest stable version.

No Perl rules were added, changed, or removed.

Our People

In our eNewsletters, we highlight the staff members behind our secure coding research. In this issue, we feature David Svoboda.



David Svoboda is a Software Security Engineer at the CERT Division of the Software Engineering Institute. He co-authored or contributed to four books, including The CERT C Coding Standard and The CERT Oracle Secure Coding Standard for Java. He also maintains the CERT Coding Standard wikis and has taught Secure Coding in C and C++ all over the world to various groups in the military, government, and banking industries. David is also involved in several ISO standards groups: the JTC1/SC22/WG14 group for standardizing C and the JTC1/SC22/WG21 group for standardizing C++.

Since 1991, David has been the primary developer on a diverse set of software development projects at Carnegie Mellon University. His projects have ranged from hierarchical chip modeling and social organization simulation to automated machine translation (AMT). His KANTOO AMT software, developed in 1996, is still in production use at Caterpillar.