Abstract

Code Review

In this 4-day face-to-face or online course, Code Review walks students through the numerous cases of undefined and platform specific behavior in C. We'll look at every part of the C language, with numerous real-world examples of bugs found by the trainer. This course is partly focused on vulnerability research. Time will be spent on relating C memory corruption heap bugs to current attacks on the Linux Heap allocator. Moreover, we'll look for ways to automate bug discovery using fuzzing and static analysis. Finally, we will look at coding recommendations and ways to prevent, fix, and secure buggy C code.

Curriculums,

Day 1 (C Refresher)

Lectures

- Introduction to the Training
- History of C
- Developing in C
- Review of C Programming Basics
- Pointers, Strings, and Arrays
- Structures and Unions
- Dynamic Memory Management

Labs

- Review of C Programming Basics
- Pointers, Strings, and Arrays
- Dynamic Memory Management

Day 2 (Vulnerability Research)

Lectures

- Virtual Memory
- Debugging
- Compiler Construction
- Data Structures
- Linux Heap Allocator Internals
- Fuzz Testing
- Dynamic Memory Checkers
- SMT Solving
- Symbolic Execution

Labs

- ptmalloc Heap Metadata Corruption
- Fuzzing and AFL
- Dynamic Memory Checkers
- Static Program Analysis
- Coccinelle

Day 3 (C Bug Classes)

Lectures

- Bugs in Preprocessor
- Bugs in Declarations and Initialisation
- Bugs in Expressions
- Bugs in Floating Point
- Bugs in Arrays
- Bugs in Characters and Strings
- Bugs in Memory Management

● Bugs in Input Output
● Bugs in Environment
● Bugs in Signals
● Bugs in Error Handling
● Bugs in Miscellaneous
● Bugs in Posix
● Navigating the Linux Kernel
● Bugs in Unix Kernels
● Code Review Strategies

Labs
● Insecure Coding

Day 4 (Real Programs, Recommendations)
Lectures
● Fixes in Preprocessor
● Fixes in Declarations and Initialisations
● Fixes in Expressions
● Fixes in Integers
● Fixes in Floating Point
● Fixes in Arrays
● Fixes in Characters and Strings
● Fixes in Memory Management
● Fixes in Input Output
● Fixes in Environment
● Fixes in Signals
● Fixes in Error Handling
● Fixes in Miscellaneous
● Fixes in Posix
● Training Close
Labs
● Userspace Auditing
● Fixing and Securing Code

Introduction of the trainer and need-to-know etc (if you have not done yet)
Dr Silvio Cesare is the Managing Director at InfoSect. He has worked in technical roles and been involved in computer security for over 20 years. This period includes time in Silicon Valley in the USA, France, and Australia. He has worked commercially in both defensive and offensive roles within engineering. He has reported hundreds of software bugs and vulnerabilities in Operating Systems kernels. He was previously the Director for Education and Training at UNSW Canberra Cyber, ensuring quality content and delivery. In his early career, he was the scanner architect and a C developer at Qualys. He is also the co-founder of BSides Canberra - Australia's largest cyber security conference. He has a Ph.D. from Deakin University and has published within industry and academia, is a 4-time Black Hat speaker, gone through academic research commercialisation, and authored a book (Software Similarity and Classification, published by Springer).

Duration of the courses
4 Days, 9am - 5pm