Secure Code Review Summary

Revision

Version 5
4/11/25 7:09 PM

SME

Charles Wilson

Abstract

This document describes the process used to produce a secure code review summary.

Group / Owner

Security / Secure Software Assessor

Motivation

This document is motivated by the need to provide security-related feedback during the development of software for use within safety-critical, cyber-physical systems for certification of compliance to standards such as **ISO/SAE 21434** and **ISO 26262**.

License

Overview

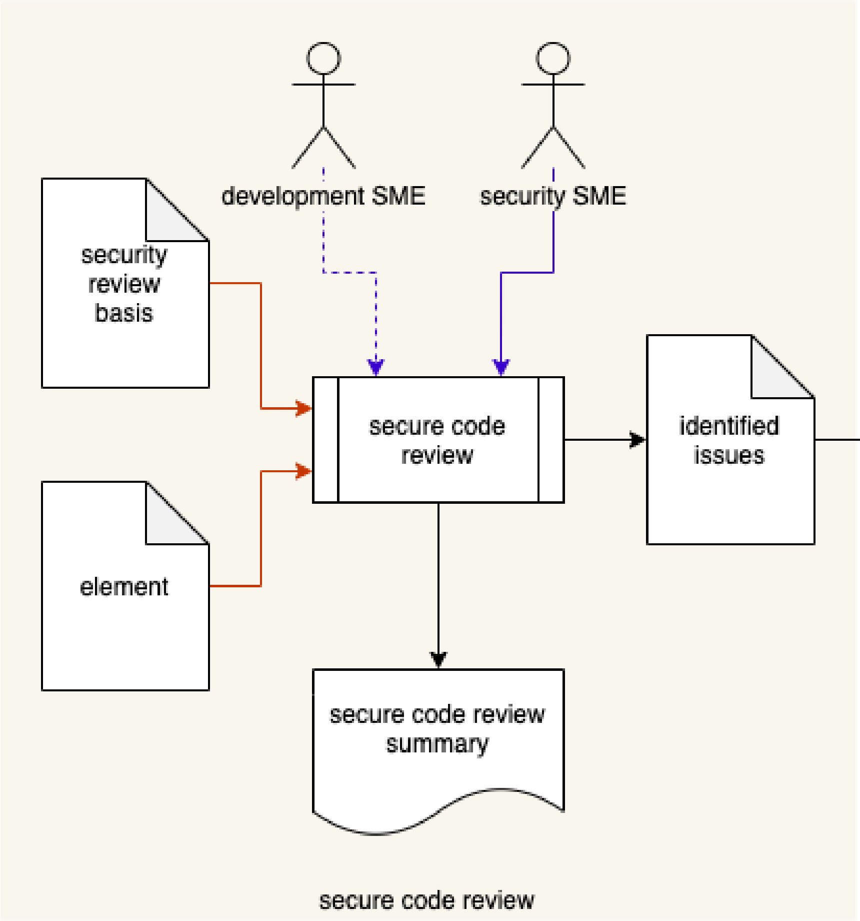
Although the quality of feedback from the compiler and static analysis tools have become much better over time with regard to security-related issues, there are many situations which are not considered by them. Secure code reviews provide an opportunity for development to take advantage of the experience of security SMEs to spot problematic usage and provide suggestions as to secure coding best practices.

Note: Although the static analysis [5], dynamic analysis [6], and fuzz testing [7] activities may be performed in parallel, it is recommended that these activities take place prior to conducting a secure code review.

Process

Secure Code Review

Inputs	Element
Outputs	Identified issues Secure code review summary
Participants	Security SME Development SME



The Security SME, possibly with input from the Development SME, review the **Element** with respect to how it implements its security-relevant aspects (**Security Review Basis**). The composition of the **Security Review Basis** is described in the **Secure Development** AVCDL secondary document. Issues identified during the review are tagged in the code review system and entered into the issue tracking system. A **Secure Code Review Summary** is generated.

References

1. **Design Showing Security Considerations** (AVCDL secondary document)
2. **Element-level Security Requirements** (AVCDL secondary document)
3. **Secure Settings Document** (AVCDL secondary document)
4. **Currently Used Deprecated Functions** (AVCDL secondary document)
5. **Static Analysis Report** (AVCDL secondary document)
6. **Dynamic Analysis Report** (AVCDL secondary document)
7. **Fuzz Testing Report** (AVCDL secondary document)
8. **Secure Design** (AVCDL secondary document)
9. **SEI CERT Coding Standards**
   https://wiki.sei.cmu.edu/confluence/display/seccode/SEI+CERT+Coding+Standards
10. **MISRA**
    https://www.misra.org.uk
11. **C++ Core Guidelines**
    https://isocpp.github.io/CppCoreGuidelines/CppCoreGuidelines
12. **Secure Coding Guidelines for Developers**
    https://docs.oracle.com/cd/E26502_01/html/E29016/scode-1.html
13. **NIST SP 800-78-4 Cryptographic Algorithms and Key Sizes for Personal Identify Verification**
    https://csrc.nist.gov/publications/detail/sp/800-78/4/final
14. **NIST SP 800-131A Transitioning the Use of Cryptographic Algorithms and Key Lengths**
    https://csrc.nist.gov/publications/detail/sp/800-131a/rev-2/final
15. **AUTOSAR Guidelines for the use of the C++14 language in critical and safety-related systems**
    https://www.autosar.org/fileadmin/user_upload/standards/adaptive/18-10/AUTOSAR_RS_CPP14Guidelines.pdf
16. **Writing Secure Code, 2ed**
    https://www.amazon.com/dp/0735617228
17. **BSIMM – Code Review**
    https://www.bsimm.com/framework/software-security-development-lifecycle/code-review.html