

Secure Coding Plan

Background

The Secure Coding Plan is intended to be used as a government-provided document that is part of the acquisition's government reference library. It is written to be customized by the acquisition organization to meet individual program needs. The information and methodologies dealing with implementing secure coding practices are continually evolving to meet the changing threats which arise on a daily basis. The requirements to support secure coding in acquisitions need to account for this and provide the flexibility to enable developers and testers to use current, best practices to support their efforts.

Purpose

The Secure Coding Plan is to provide a consistent, complete set of requirements with references to implement secure coding practices for acquisition organizations. The plan is designed to be tailored by individual acquisition organizations for use in their specific programs. The plan will identify ties to program CDRLs and milestones so that a better understanding of the effort required to support the implementation of secure coding practices can be evaluated and planned for by the contractors and the acquisition organizations. The Secure Coding Acquisition Approach slide set further refines the plan.

References

- [1] The CERT Secure Coding Standards Wiki. <https://www.securecoding.cert.org/>
- [2] Seacord, Robert. *The CERT C Secure Coding Standard*. Boston, MA: Addison-Wesley, 2008.
- [3] The CERT C Secure Coding Standard Wiki. <https://www.securecoding.cert.org/confluence/x/HQE>
- [4] The CERT C++ Secure Coding Standard Wiki. <https://www.securecoding.cert.org/confluence/x/fQI>
- [5] The CERT Sun Microsystems Secure Coding Standard for Java Wiki. <https://www.securecoding.cert.org/confluence/x/Ux>
- [6] Seacord, Robert. *Secure Coding in C and C++*. Boston, MA: Addison-Wesley Professional, 2005.

Oversight

The CERT Secure Coding Standards Wiki [1] provides current, but not uniformly vetted, information secure coding guidelines. If custom software is being developed in C, then Version 1.0 of the CERT C Secure Coding Standard is to be used as the starting point for a secure coding standard. Information provided on the The CERT C Secure Coding Standard Wiki [3] should be considered for interpreting Version 1.0 of the CERT C Secure Coding Standard. If custom software is being developed in C++, then the CERT C++ Secure Coding Standard Wiki [4] is to be used as the starting point until the standard has been released. The acquisition organization will work with the contractor to develop the secure coding standard to be used on the program. If custom software is being development in Java, then The CERT

Sun Microsystems Secure Coding Standard for Java Wiki [5] is to be used as the starting point. The acquisition organization will work with the contractor to develop the secure coding standard to be used on the program. A mapping of secure coding guidelines to DISA's Security Technical Implementation Guide (STIG) Application Security and Development Checklist is available on the wiki for each language.

Compliance to a secure coding standard

The secure coding guidelines are classified as either rules or recommendations. Conformance to the secure coding rules defined in the standards are necessary (but not sufficient) to ensure the security of software systems developed. Conformance to secure coding recommendations is recommended, but not required.

Development of a Deviation Procedure for the Secure Coding Standards

Strict adherence to all rules is likely not possible for each program. Consequently, deviations associated with individual situations are permissible. Deviations may occur for a specific instance, typically in response to circumstances that arise during the development process or for a systematic use of a particular construct in a particular circumstance. Systematic deviations are usually agreed upon at the start of a project.

It is necessary that a formal procedure be used to authorize these deviations. The use of deviation must be justified on the basis of both necessity and security. Rules that have higher severity and/or a high likelihood require a more compelling argument for agreeing to a deviation than do rules with a low severity that are unlikely to result in a vulnerability.

Test and evaluation impacts

To ensure that the source code conforms to the secure coding standard, it is necessary to have measures in place that check for rules and selected recommendations violations. The most effective way to achieve this is through the use of automated analysis tools. In cases where the rule or selected recommendation can not be checked by an automated analysis tool, then manual review is required.

Impacts to possible program CDRLs

- Program Management Plan – training to support the secure coding standard for development and testing organizations; testing effort to show compliance to secure coding standard; effort needed to assess new rules and recommendations on a periodic basis to address new threats and mitigations and update the secure coding standard appropriately
- Software Development Plan – integration of the secure coding standard into the development process, use of source code analysis to demonstrate compliance with the secure coding standard, tying conformance to the secure coding standard with program milestones, impact on language and tool selection process, impact on code review process
- Software Test Plan – how compliance with the secure coding standard is determined (manual review, automated analysis tools)

- Training Plan – secure coding standard training, training for process used for secure coding standard compliance
- Configuration Management Plan - change control process to support the secure coding standard

Impacts to acquisition documents

- System Engineering Plan – identify that secure coding standards will be used in the software development and the impacts that will have on test and evaluation and security/information assurance
- Test and Evaluation Master Plan – impact of secure code standards on the testing process
- Acquisition Strategy and Plan – use of secure code standards will improve software quality and is a risk reduction technique

DRAFT