Thomas Plum

Subject: WG14/N1412: draft proposed TR

Reply to: Thomas Plum, tplum@plumhall.com

-----Original Message-----
From: owner-sc22wg14@open-std.org On Behalf Of Thomas Plum
Sent: Saturday, August 15, 2009 5:43 AM
To: sc22wg14@open-std.org
Subject: (SC22WG14.11790) draft proposed TR

Robert Seacord posted a draft proposed TR on the wiki; see
http://wiki.dinkumware.com/twiki/pub/WG14/SecurityTRProposal/ISO-IEC_TR_E_C_Secure_Coding_Guidelines_TR.pdf

This draft has been assigned doc number WG14/N1393. We have requested agenda time at Santa Cruz to consider initiating an NP for this TR.

During our discussions with members of the committee, we received requests for a one-page "executive summary". Here is a draft of such a summary (and please send your questions and comments, to the authors or to the reflector):

C Secure Coding Guidelines - Executive Summary

This proposed Technical Report specifies guidelines for secure coding in the C programming language and non-compliant code examples. This Technical Report (TR) does not specify the mechanism by which these guidelines are enforced or any particular coding style to be enforced.
The proposed project editor is Robert C. Seacord, Secure Coding Team Lead at CERT, located at Carnegie Mellon University's Software Engineering Institute.

Some projects follow rigorous standards for safety-critical software (such as IEC 61508 or DO-178B); refer to TR 24772 for definitions and discussion. Other projects follow locally-developed processes that reflect the demands of intensely competitive commercial marketplaces.
Within this wide range of development processes there may be one common factor: the project has chosen to use a C compiler that conforms to ISO/IEC 9899. If the project has chosen to make use of the full range of features of 9899, this TR does not restrict that choice unless absolutely necessary. A guideline that simply says "don't use language feature X" may be easy to draft, or easy to test with a tool, but is unsuitable for a set of guidelines that address the full range of features of ISO/IEC C language.

Various software defects ("bugs") may create serious problems (according to the project's chosen development criteria) while not creating a security vulnerability. This TR does not attempt to address these software defects, leaving them to the project's chosen development criteria, so that this TR can focus only on the specific issues that create security vulnerabilities. There are large databases such as the US-CERT Vulnerability Notes Database, NIST's National Vulnerability Database (NVD), and MITRE's Common Vulnerability Enumeration (CVE) that catalog causes of empirically-observed security problems; eliminating or reducing these software vulnerabilities is the goal of this TR.

Some projects may require a high degree of portability, to compile and execute on any system with an ISO/IEC C compiler. Other projects may have sound reasons to restrict the set of target compilers or systems.
It is not a goal of these guidelines to require any specific level of portability. In other words, software portability is not intrinsically required to avoid security

vulnerabilities. One reason why these two system qualities are sometimes confounded is that various situations are categorized in ISO/IEC C as "undefined behavior" only because the behavior is non-portable, not because it introduces any security vulnerability. These guidelines have benefitted from the recent work within WG14 which distinguishes critical undefined behaviors from the other (less serious) bounded undefined behaviors.

An analyzer, as defined by this TR, is the mechanism that diagnoses coding flaws in software programs. This may include static analysis tools or tools within a compiler suite. This TR is drafted such that a conforming analyzer is required to diagnose all violations of coding guidelines specified herein. These guidelines may be extended in an implementation-dependent manner. This TR assumes that, if diagnostics are generated, the programmer can make corresponding revisions to the original source code; therefore, these guidelines are primarily intended for new code.

The various people who have authored and proposed this TR look forward to detailed scrutiny from the technical experts in SC22/WG14 (ISO/IEC C).

NOTE: The draft was prepared from an ISO doc template for a Technical Report; there are several fields that cannot be filled in until the project is underway, so please ignore the "Error! Reference source not found" messages.

Thomas Plum, Plum Hall Inc, 3 Waihona Box 44610, Kamuela HI 96743 USA

tplum@plumhall.com ..... TEL +1-808-882-1255 FAX +1-808-882-1556

http://www.PlumHall.com ..... TOLLFREE +1-800-PLUM-HALL (800-758-6425)