システムグループ

組込みソフトウェア開発向けコーディング作法ガイド（ESCR）の改訂について

SEC 調査役
十山　圭介

SEC 調査役
三原　幸博

1 組込みソフトウェア開発向けコーディング作法ガイド（ESCR）の改訂状況

IPA/SEC では組込みソフトウェアのソースコード品質をより良いものとすることを目的に、C 言語と C++ 言語において、コーディングの際に注意すべき事柄やノウハウを「コーディング作法」という形で整備し、それらを取りまとめたガイドラインとして「組込みソフトウェア開発向けコーディング作法ガイド（ESCR）」（以下 ESCR）を公開している。ESCR は、コーディングの際の基本的な考え方（作法）と作法を具体化した守るべき個々の事項（ルール）をソフトウェア品質特性の観点で整理しており、組織内でコーディングルールを決める際や実際のコーディング時の参考、またプログラミング学習のため、書籍や PDF 版などこれまで３万部を超えて多くの方々に利用いただいている。

新たな機能を導入するなど言語の標準規格は定常的に改訂されており、ESCR についても多くのユーザが使用する規格に準拠して内容の更新の有無を検討し、必要な改版を実施しなければならない。ESCR［C 言語版］は 2006 年に Ver. 1.0、2007 年に一部修正を行った Ver. 1.1 を発行している。2014 年３月には JIS 最新の規格である C99 に準拠し、更に近年 C99 に対応して大幅に改訂された欧州組込み業界標準規格である MISRA C:2012 との整合性も確保した。

また、近年広く使用されるようになってきている C++ 言語向けにも 2003 年版の言語規格に準拠した ESCR［C++ 言語版］Ver. 1.0 を 2010 年に発行している。こちらについても、C++ 言語の新規格 C++11 及び C++14 に準拠し、また先に改訂した ESCR［C 言語版］との整合性を確保するべく、2014 年度より ESCR［C++ 言語版］の改訂作業を開始している。

2 ESCR のセキュアコーディングへの対応

ソフトウェアは常に攻撃の脅威に曝されており、ソフトウェアの欠陥による「脆弱性」が攻撃されることでセキュリティ被害がもたらされる。この脆弱性を作り込まない / 軽減させるよう、正しく動作するプログラムを書くことがセキュアコーディングである。

これまで、JIS によるソフトウェア品質規格では「セキュリティ」は「機能性」（設計段階の特性）に含まれる副特性であり、実装段階を支援する ESCR では積極的には取り上

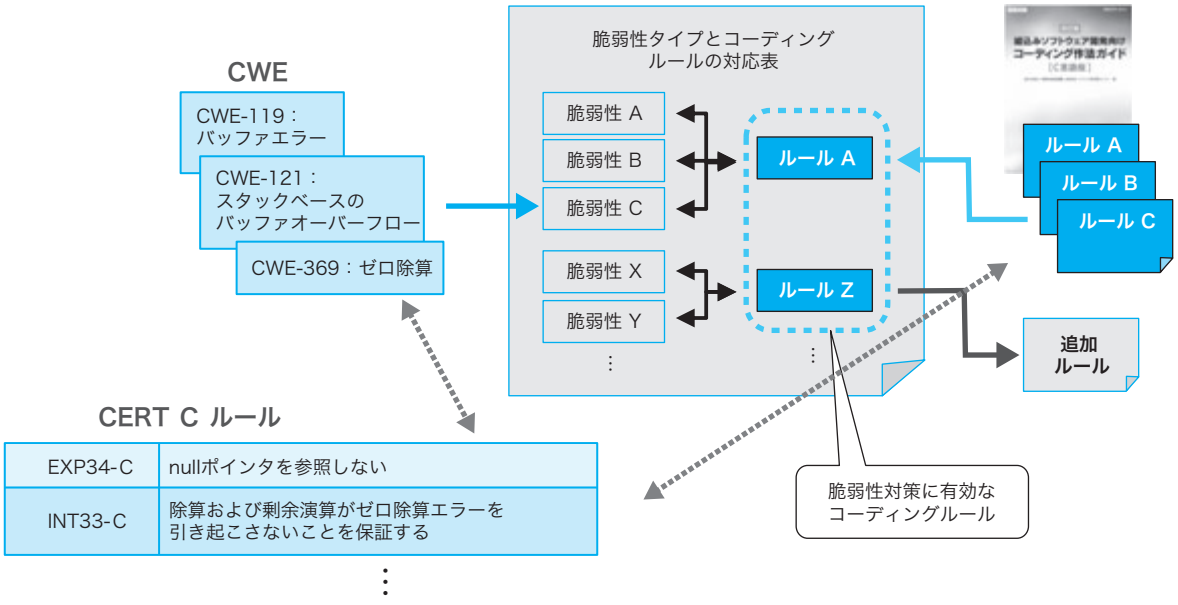

図１　ESCR ルールと CERT C ルール、CWE との関連

表1　ESCR ルールと CERT C ルール、CWE の対応表（抜粋）

作法詳細	ルール		MISRA ルールとの関係 C:2004	MISRA ルールとの関係 C:2012	CERT C	CWE
［信頼性 1］R1 領域は初期化し、大きさに気を付けて使用する。						
R1.1 領域は、初期化してから使用する。	R1.1.1	自動変数は宣言時に初期化する。または値を使用する直前に初期値を代入する。	9.1	R9.1	EXP33-C	CWE-456
	R1.1.2	const 型変数は、宣言時に初期化する。			EXP40-C	CWE-456
R1.2 初期化は過不足無いことが分かるように記述する。	R1.2.1	要素数を指定した配列の初期化では、初期値の数は、指定した要素数と一致させる。			ARR02-C STR11-C	
	R1.2.2	列挙型（enum 型）のメンバの初期化は、定数を全く指定しない、すべて指定する、または最初のメンバだけを指定する、のいずれかとする。	9.3	R9.4	INT09-C	CWE-665
R1.3 ポインタの指す範囲に気を付ける。	R1.3.1	（1）ポインタへの整数の加減算（++、-- も含む）は使用せず、確保した領域への参照・代入には [] を用いる配列形式で行う。 （2）ポインタへの整数の加減算（++、-- も含む）は、ポインタが配列を指している場合だけとし、結果は配列の範囲内を指すようにする。	17.1 17.4	R17.1 R17.4	ARR30-C ARR37-C	CWE-468 CWE-788 CWE-823
	R1.3.2	ポインタ同士の減算は、同じ配列の要素を指すポインタにだけ使用する。	17.2	R17.2	ARR36-C	CWE-469
	R1.3.3	ポインタ同士の比較は、同じ配列の要素、または同じ構造体のメンバを指すポインタにだけ使用する。	17.3	R17.3	ARR36-C	CWE-188
	R1.3.4	restrict 型修飾子は使用しない。【MISRA C:2012 R8.14】		R8.14	EXP43-C	
［信頼性 2］R2 データは、範囲、大きさ、内部表現に気を付けて使用する。						
R2.1 内部表現に依存しない比較を行う。	R2.1.1	浮動小数点式は、等価または非等価の比較をしない。	13.3	D1.1	FLP00-C	
	R2.1.2	浮動小数点型変数はループカウンタとして使用しない。	13.4	R13.1	FLP00-C FLP30-C	
	R2.1.3	構造体や共用体の比較に memcmp を使用しない。			EXP42-C	CWE-188
R2.2 論理値などが区間として定義されている場合、その中の一点（代表的な実装値）と等しいかどうかで判定を行ってはならない。	R2.2.1	真偽を求める式の中で、真として定義した値と比較しない。				
R2.3 データ型をそろえた演算や比較を行う。	R2.3.1	符号なし整数定数式は、結果の型で表現できる範囲内で記述する。	12.11	R12.10	INT30-C	CWE-190
	R2.3.2	条件演算子（?: 演算子）では、論理式は括弧で囲み、戻り値は 2 つとも同じ型にする。			INT02-C	
	R2.3.3	ループカウンタとループ継続条件の比較に使用する変数は、同じ型にする			INT02-C	

げていなかった。後継規格である JIS X 25010 で「セキュリティ」が特性として位置付けられたが、ESCR［C 言語版］Ver.2.0 ではセキュリティの観点での整理は行わず、バッファオーバーフローを避けるなどセキュリティに影響するコーディングもあるとして「CERT C コーディングスタンダード※1」を参照する旨を記載している。

一方、ESCR の個々のコーディングルールの中には、セキュリティの観点から見て重要なものが含まれている。それらと脆弱性との対応関係を示してセキュリティ面からも体系化することで、ソフトウェア品質の一層の向上に寄与できると見込める。

以上の背景から、SEC では IPA セキュリティセンターと連携して以下のように ESCR のルールとセキュアコーディングとの関連付けを実施した。

- 国際的に活用される脆弱性タイプ CWE（共通脆弱性タイプ一覧）※2 の中から、重要な 14 種を選定し、それを軽減しうる ESCR ルールとの対応を明らかにする
- ESCR ルールに対して、それと同等の意味を持つ CERT C コーディングスタンダードのルールを選定し、対応付けを行う

なお、この対応表については、ESCR の PDF 版の付録として公開した※3。

図 1 はこれらの対応関係の概要を図示したもので、表 1 は ESCR ルールと CERT C ルール、CWE の対応関係の表（抜粋）である。

【脚注】

※ 1　脆弱性につながる恐れのある危険なコーディングや未定義の動作を削減することを目的に定められたコーディング規約。CERT C Coding Standard の日本語版。

※ 2　Common Weakness Enumeration：ソフトウェアにおけるセキュリティ上の弱点（脆弱性）の種類を識別するための共通の基準。

※ 3　https://www.ipa.go.jp/sec/publish/tn13-001.html