
# MISRA C:2012 – Addendum 3

## Coverage of MISRA C:2012 (including Amendment 1) against CERT C 2016 Edition

January 2018


First published January 2018 by HORIBA MIRA Limited
Watling Street
Nuneaton
Warwickshire
CV10 0TU
UK

www.misra.org.uk



ISBN 978-1-906400-19-4 PDF

British Library Cataloguing in Publication Data
A catalogue record for this book is available from the British Library

# MISRA C:2012 – Addendum 3

## Coverage of MISRA C:2012 (including Amendment 1) against CERT C 2016 Edition

January 2018

# MISRA Mission Statement

We provide world-leading best practice guidelines for the safe and secure application of both embedded control systems and standalone software.

MISRA is a collaboration between manufacturers, component suppliers and engineering consultancies which seeks to promote best practice in developing safety- and security-related electronic systems and other software-intensive applications. To this end MISRA publishes documents that provide accessible information for engineers and management, and holds events to permit the exchange of experiences between practitioners.

# Foreword

The vision of MISRA C is set out in the opening paragraph of the Guidelines:

> "The MISRA C Guidelines define a subset of the C language in which the opportunity to make mistakes is either removed or reduced".

Many standards for the development of safety-related software require, or recommend, the use of a language subset, and this can also be used to develop any application with high integrity or high reliability requirements.

Unfortunately, many people focus on the safety-related software reference, and a perception exists that MISRA C is only *safety-related* and not *security related*.

In 2008, the Software Engineering Institute at Carnegie Mellon University published CERT C, as a "secure coding standard". A second edition was published in 2014, with a further update released in 2016 (PDF only).

This third Addendum to MISRA C:2012 sets out the coverage by MISRA C:2012 of the 2nd Edition of CERT C and justifies the viewpoint that MISRA C is equally as applicable in a *security related* environment as it is in a *safety-related* one – particularly relating to the development of *freestanding applications*. Ongoing developments of MISRA C will further address issues in the *hosted* domain.

Andrew Banks FBCS CITP
Chairman, MISRA C Working Group

# Acknowledgements

The MISRA consortium would like to thank the following individuals for their significant contribution to the writing of this document:

| | |
|---|---|
| Andrew Banks | Intuitive Consulting |
| Jill Britton | Programming Research Ltd |

The MISRA consortium also wishes to acknowledge contributions from the following members of the MISRA C Working Group during the development and review process:

| | |
|---|---|
| Fulvio Baccaglini | Programming Research Ltd |
| Dave Banham | Rolls-Royce plc |
| Paul Burden | Independent Consultant (Formerly Programming Research Ltd) |
| Mike Hennell | LDRA Ltd |
| Chris Hills | Phaedrus Systems Ltd |
| Chris Tapp | LDRA Ltd |
| Liz Whiting | LDRA Ltd |

The MISRA consortium also wishes to acknowledge contributions from the following individuals during the development and review process:

| | |
|---|---|
| David Ward | HORIBA MIRA Ltd |

# Contents

# 1 Introduction

## 1.1 Glossary

In this document:

| | |
|---|---|
| MISRA C | means the MISRA C:2012 Guidelines [1] |
| AMD1 | means Amendment 1 to MISRA C:2012 Guidelines [2] |
| CERT C | means SEI CERT C Coding Standard [3] |

## 1.2 Background

Throughout the development of MISRA C, the main focus has been to address vulnerabilities in the C language, particularly for use in embedded systems, and primarily targeted at safety-related applications. MISRA C particularly applies to freestanding applications, which use a sub-set of the C Standard Library.

One of the great successes of MISRA C has been its adoption across many industries, and in environments where safety-criticality is less of a concern, but where data-security is more of an issue.

There have been discussions as to the applicability of MISRA C for secure applications. The MISRA C Working Group have listened to those concerns, and have compiled this Addendum to document the coverage of MISRA C against CERT C.

# 2 Coverage

## 2.1 Coverage classification

The coverage of each Cert C rule against MISRA C is classified as follows:

| Status | Interpretation |
|---|---|
| Explicit | The behaviour addressed by the CERT C rule is EXPLICITLY covered by one or more MISRA C guidelines, which directly addresses the undesired behaviour. |
| Implicit | The behaviour addressed by the CERT C rule is IMPLICITLY covered by one or more MISRA C guidelines, although the behaviour is not explicitly referenced. |
| Restrictive | The behaviour addressed by the CERT C rule is covered by one or more MISRA C guidelines that prohibit a language feature in a RESTRICTIVE manner. For example:<br>• Rule 21.3 – `<stdlib.h>` (memory allocation/deallocation)<br>• Rule 21.5 – `<signal.h>` (all)<br>• Rule 21.6 – `<stdio.h>` (input/output functions)<br>• Rule 21.8 – `<stdlib.h>` (getenv()) |
| Partial/Restrictive | Some aspects of the behaviour addressed by the CERT C rule are covered in a RESTRICTIVE manner.<br>However, some aspects of the behaviour are not covered by any MISRA C guidelines. |
| Out of Scope | Aspects of behaviour are out of scope for C99 and are related to C11. |
| None | The behaviour addressed by the CERT C rule is not covered by any MISRA C guidelines. |

## 2.2 Coverage strength

The strength of the coverage of each CERT C rule against MISRA C is classified as follows:

| Status | Interpretation |
|---|---|
| Strong | The behaviour addressed by the CERT C rule is covered by one or more targeted MISRA C rules. |
| Weak | The behaviour addressed by the CERT C rule is only covered by one or more MISRA C directives, or by Rule 1.3. |
| None | The behaviour addressed by the CERT C rule is not covered by any MISRA C guidelines. |

*Note:* For CERT C rules with "partial" coverage, a combination of strength coverages is shown.

# 3 CERT C cross reference

## 3.1 Guideline by guideline

| CERT C Rule | MISRA C:2012 | | | Comments |
|---|---|---|---|---|
| | Guidelines | Coverage | | |
| PRE30-C | Rule 1.3 | Implicit | Weak | |
| PRE31-C | Rule 13.2 | Explicit | Strong | |
| PRE32-C | Rule 1.3, 20.6 | Explicit | Strong | |
| DCL30-C | Rule 18.6 | Explicit | Strong | |
| DCL31-C | Rule 8.1, 17.3 | Explicit | Strong | |
| DCL36-C | Rule 8.2, 8.4, 8.8, 17.3 | Explicit | Strong | |
| DCL37-C | Rule 1.3, 21.1, 21.2 | Explicit | Strong | |
| DCL38-C | Rule 1.1, 1.3, 21.3 | Restrictive | Strong | Dynamic memory allocation is not permitted by MISRA C |
| DCL39-C | | Out of Scope | None | |
| DCL40-C | Rule 1.3, 5.1, 5.2, 8.4, 8.5 | Implicit | Weak | |
| DCL41-C | Rule 16.1 | Explicit | Weak | |
| EXP30-C | Rule 13.2 | Explicit | Strong | |
| EXP32-C | Rule 1.3, 11.8 | Explicit | Strong | |
| EXP33-C | Dir 4.2<br>Rule 9.1, 21.3 | Explicit | Strong | |
| EXP34-C | Dir 4.1, 4.14<br>Rule 1.3 | Implicit | Weak | |
| EXP35-C | | Out of Scope | None | |
| EXP36-C | Rule 11.1, 11.2, 11.3, 11.4, 11.5, 11.6, 11.7 | Explicit | Strong | |
| EXP37-C | Rule 8.2, 17.3 | Explicit | Strong | |
| EXP39-C | Rule 1.3, 11.1, 11.2, 11.3, 11.7 | Explicit | Strong | |

# 3: CERT C cross reference

| CERT C Rule | MISRA C:2012 | | | Comments |
|---|---|---|---|---|
| | Guidelines | Coverage | | |
| EXP40-C | Rule 1.3, 7.4, 11.4, 11.8, 19.2 | Implicit | Weak | |
| EXP42-C | Rule 21.16 | Explicit | Strong | |
| EXP43-C | Rule 1.3, 8.14, 19.1 | Restrictive | Strong | |
| EXP44-C | Rule 13.6 | Explicit | Strong | |
| EXP45-C | Rule 13.4 | Explicit | Strong | |
| EXP46-C | | Out of Scope | None | |
| INT30-C | Rule 12.4 | Explicit | Strong | |
| INT31-C | Rule 10.1,10.3, 10.4, 10.5, 10.6, 10.7, 10.8 | Explicit | Strong | |
| INT32-C | Dir 4.1<br>Rule 1.3 | Implicit | Weak | |
| INT33-C | Dir 4.1<br>Rule 1.3 | Implicit | Weak | |
| INT34-C | Rule 10.1, 12.2 | Explicit | Strong | |
| INT35-C | | None | None | |
| INT36-C | Rule 11.1, 11.2, 11.4, 11.6, 11.7 | Explicit | Strong | |
| FLP30-C | Rule 14.1 | Explicit | Strong | |
| FLP32-C | Dir 4.11 | Explicit | Weak | |
| FLP34-C | Rule 10.1, 10.3, 10.4, 10.5, 10.8 | Explicit | Strong | |
| FLP36-C | Rule 10.3, 10.4, 10.5, 10.8 | Implicit | Weak | |
| FLP37-C | Rule 21.16 | Explicit | Strong | |
| ARR30-C | Rule 1.3, 18.1, 21.17, 21.18 | Implicit | Weak | |
| ARR32-C | Rule 18.8 | Restrictive | Strong | |
| ARR36-C | Rule 18.2, 18.3 | Explicit | Strong | |
| ARR37-C | Rule 18.1, 18.4 | Explicit | Strong | |
| ARR38-C | Rule 1.3, 21.17, 21.18 | Implicit | Weak | |

| CERT C Rule | MISRA C:2012 | | | Comments |
|---|---|---|---|---|
| | Guidelines | Coverage | | |
| ARR39-C | Rule 1.3, 18.4 | Explicit | Strong | |
| STR30-C | Rule 7.4 | Explicit | Weak | |
| STR31-C | Dir 4.1<br>Rule 1.3, 18.1, 21.6, 21.17, 21.18 | Implicit | Weak | |
| STR32-C | Rule 21.16 | Explicit | Weak | |
| STR34-C | Rule 10.1, 10.3 10.4 | Explicit | Strong | |
| STR37-C | Rule 10.3, 21.13 | Explicit | Strong | |
| STR38-C | Rule 1.3, 10.3 | Explicit | Strong | |
| MEM30-C | Dir 4.12<br>Rule 1.3, 21.3, 22.2 | Explicit | Strong | |
| MEM31-C | Rule 22.1 | Explicit | Strong | |
| MEM33-C | Rule 1.3, 18.7 | Restrictive | Strong | |
| MEM34-C | Rule 22.2 | Explicit | Strong | |
| MEM35-C | Dir 4.1, 4.12<br>Rule 1.3, 21.3 | Restrictive | Strong | |
| MEM36-C | Rule 21.3 | Restrictive | Strong | |
| FIO30-C | Dir 4.14 | Implicit | Weak | |
| FIO31-C | Rule 22.3 | Partial | Weak | This rule was removed from the 2016 edition of the CERT C standard and is not included in the coverage summary |
| FIO32-C | Rule 21.6 | Restrictive | Strong | Rule 21.6 bans all functions in `<stdio.h>` |
| FIO34-C | | None | None | |
| FIO37-C | Rule 21.6 | Restrictive | Strong | Rule 21.6 bans all functions in `<stdio.h>` |
| FIO38-C | Rule 22.5 | Implicit | Strong | |
| FIO39-C | Dir 4.13<br>Rule 1.3, 21.6 | Restrictive | Strong | Rule 21.6 bans all functions in `<stdio.h>` |
| FIO40-C | Rule 21.6 | Restrictive | Strong | Rule 21.6 bans all functions in `<stdio.h>` |

# 3: CERT C cross reference

| CERT C Rule | MISRA C:2012 | | | Comments |
|---|---|---|---|---|
| | Guidelines | Coverage | | |
| FIO41-C | Rule 13.1, 13.2, 13.3, 13.4, 13.5, 13.6, 21.6 | Restrictive | Strong | |
| FIO42-C | Rule 22.1 | Explicit | Strong | |
| FIO44-C | Rule 1.3, 21.6 | Restrictive | Strong | |
| FIO45-C | | Out of Scope | None | |
| FIO46-C | Rule 22.6 | Explicit | Strong | |
| FIO47-C | Rule 1.3, 21.6 | Restrictive | Strong | |
| ENV30-C | Rule 21.19 | Explicit | Strong | |
| ENV31-C | Rule 1.3 | Implicit | Weak | |
| ENV32-C | Rule 21.4, 21.8 | Restrictive | Strong | |
| ENV33-C | Rule 21.8 | Explicit | Strong | |
| ENV34-C | Rule 21.20 | Explicit | Strong | |
| SIG30-C | Rule 1.3, 21.5 | Restrictive | Strong | Rule 21.5 bans all functions in `<signal.h>` |
| SIG31-C | Rule 1.3, 21.5 | Restrictive | Strong | Rule 21.5 bans all functions in `<signal.h>` |
| SIG34-C | Rule 1.3, 21.5 | Restrictive | Strong | Rule 21.5 bans all functions in `<signal.h>` |
| SIG35-C | Rule 1.3, 21.5 | Restrictive | Strong | Rule 21.5 bans all functions in `<signal.h>` |
| ERR30-C | Rule 22.8, 22.9, 22.10 | Explicit | Strong | |
| ERR32-C | Rule 1.3, 21.5 | Explicit | Strong | |
| ERR33-C | Dir 4.7 | Explicit | Weak | |
| CON30-C | | Out of Scope | None | |
| CON31-C | | Out of Scope | None | |
| CON32-C | | Out of Scope | None | |
| CON33-C | | Out of Scope | None | |
| CON34-C | | Out of Scope | None | |
| CON35-C | | Out of Scope | None | |
| CON36-C | | Out of Scope | None | |

| CERT C Rule | MISRA C:2012 | | | Comments |
|---|---|---|---|---|
| | Guidelines | Coverage | | |
| CON37-C | Rule 21.5 | Restrictive | Strong | Rule 21.5 bans all functions in `<signal.h>` |
| CON38-C | | Out of Scope | None | |
| CON39-C | | Out of Scope | None | |
| CON40-C | | Out of Scope | None | |
| CON41-C | | Out of Scope | None | |
| MSC30-C | | None | None | |
| MSC32-C | | None | None | |
| MSC33-C | Rule 21.10 | Restrictive | Strong | |
| MSC37-C | Rule 17.4 | Explicit | Strong | |
| MSC38-C | Rule 1.1, 1.3 | Implicit | Weak | |
| MSC39-C | Rule 17.1 | Restrictive | Strong | |
| MSC40-C | Rule 1.1 | Restrictive | Strong | |

## 3.2 Coverage summary

In summary, the coverage of MISRA C:2012 against CERT C is as follows:

| Classification | Strength | Number |
|---|---|---|
| Explicit | Strong | 39 |
| | Weak | 5 |
| Implicit | Strong | 1 |
| | Weak | 13 |
| Restrictive | Strong | 22 |
| | Weak | 0 |
| Partial | Strong/Weak/None | 0 |
| Out of Scope | None | 15 |
| None | None | 4 |
| | Total | 99 |

# 4 References


[1] MISRA C:2012, *Guidelines for the use of the C language in critical systems*, ISBN 978-1-906400-10-1, MIRA Limited, Nuneaton, March 2013

[2] MISRA C:2012, A*mendment 1: Additional security guidelines for MISRA C:2012*, ISBN 978-1-906400-16-3, HORIBA MIRA Limited, Nuneaton, April 2016

[3] *SEI CERT C Coding Standard – Rules for Developing Safe, Reliable, and Secure Systems* - 2016 Edition