

# Package ‘sanitizers’

July 23, 2025

**Type** Package

**Title** C/C++ Source Code to Trigger Address and Undefined Behaviour Sanitizers

**Version** 0.1.1

**Date** 2023-06-11

**Author** Dirk Eddelbuettel

**Maintainer** Dirk Eddelbuettel <edd@debian.org>

**Description** Recent gcc and clang compiler versions provide functionality to test for memory violations and other undefined behaviour; this is often referred to as “Address Sanitizer” (or ‘ASAN’) and “Undefined Behaviour Sanitizer” (‘UBSAN’). The Writing R Extension manual describes this in some detail in Section 4.3 title “Checking Memory Access”.

This feature has to be enabled in the corresponding binary, eg in R, which is somewhat involved as it also required a current compiler toolchain which is not yet widely available, or in the case of Windows, not available at all (via the common Rtools mechanism).

As an alternative, pre-built Docker containers such as the Rocker container ‘r-devel-san’ or the multi-purpose container ‘r-debug’ can be used.

This package then provides a means of testing the compiler setup as the known code failures provides in the sample code here should be detected correctly, whereas a default build of R will let the package pass.

The code samples are based on the examples from the Address Sanitizer Wiki at <<https://github.com/google/sanitizers/wiki>>.

**License** GPL (>= 2)

**URL** <https://github.com/eddelbuettel/sanitizers>,  
<https://dirk.eddelbuettel.com/code/sanitizers.html>

**BugReports** <https://github.com/eddelbuettel/sanitizers/issues>

**NeedsCompilation** yes

**Repository** CRAN

**Date/Publication** 2023-06-11 16:40:02 UTC

## Contents

sanitizers-package . . . . .	2
------------------------------	---

<b>Index</b>	<b>3</b>
--------------	----------

---

sanitizers-package	<i>C/C++ Source Code to Trigger Address and Undefined Behaviour Sanitizers</i>
--------------------	--

---

## Description

Recent gcc and clang compiler versions provide functionality to test for memory violations and other undefined behaviour; this is often referred to as "Address Sanitizer" (or 'ASAN') and "Undefined Behaviour Sanitizer" ('UBSAN'). The Writing R Extension manual describes this in some detail in Section 4.3 title "Checking Memory Access".

This feature has to be enabled in the corresponding binary, eg in R, which is somewhat involved as it also required a current compiler toolchain which is not yet widely available, or in the case of Windows, not available at all (via the common Rtools mechanism).

As an alternative, pre-built Docker containers such as the Rocker container 'r-devel-san' or the multi-purpose container 'r-debug' can be used.

This package then provides a means of testing the compiler setup as the known code failures provides in the sample code here should be detected correctly, whereas a default build of R will let the package pass.

The code samples are based on the examples from the Address Sanitizer Wiki at <<https://github.com/google/sanitizers/wiki>>.

## Package Content

Index of help topics:

sanitizers-package	C/C++ Source Code to Trigger Address and Undefined Behaviour Sanitizers
--------------------	---

## Maintainer

Dirk Eddelbuettel <[edd@debian.org](mailto:edd@debian.org)>

## Author(s)

Dirk Eddelbuettel

# Index

## \* **package**

sanitizers-package, [2](#)

heapAddressSanitize

(sanitizers-package), [2](#)

intOverflowSanitize

(sanitizers-package), [2](#)

sanitizers (sanitizers-package), [2](#)

sanitizers-package, [2](#)

stackAddressSanitize

(sanitizers-package), [2](#)