[Notification]

Detect Memory Corruption in Using Dynamic Memory Management Functions for Quality and Security Improvement!

R20TS0467EJ0100
Rev.1.00
Sep. 01, 2019

Renesas Compiler Professional Edition
Enhanced Security for Dynamic Memory Management Functions

Outline

We are introducing one of the features of Renesas compiler (CC-RL/CC-RX/CC-RH) professional edition; “Enhanced security for dynamic memory management functions”.

This feature allows users to check the heap area reserved by the dynamic memory management functions to detect memory corruption or invalid operations. It will help improve the quality and security of user programs.

1. Features

1.1 Improve Quality and Security of User Programs

By using the dynamic memory management functions, Renesas compilers (CC-RL/CC-RX/CC-RH) can dynamically reserve and release memory space (the heap area).

However, in using the heap area, a program may result in a runaway or malfunction due to invalid operations such as described below.

- Writing beyond the size of allocated memory in the heap
- Releasing the heap twice

This feature of “Enhanced security for dynamic memory management functions” can prevent a program runaway or malfunction by checking the ‘monitoring areas’ and ‘pointers’ and detecting invalid operations in the heap.

Below are process flows with and without the use of the “Enhanced security for dynamic memory management functions”.

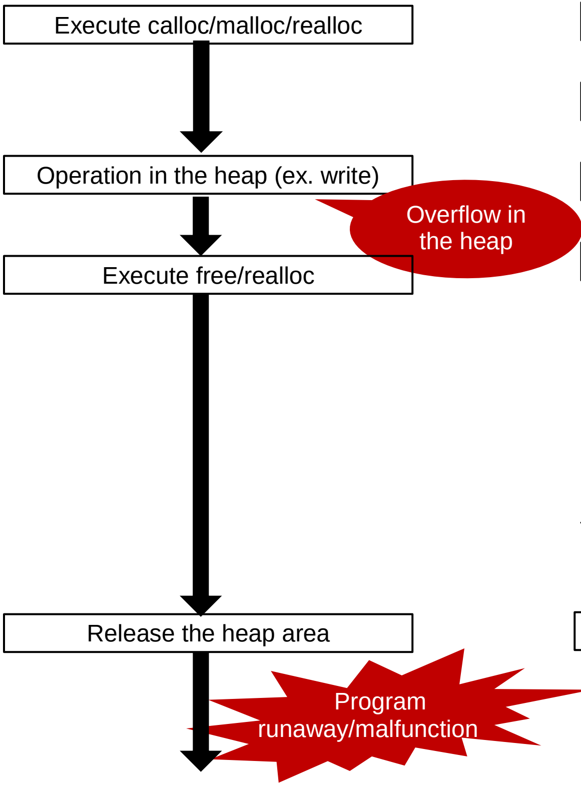
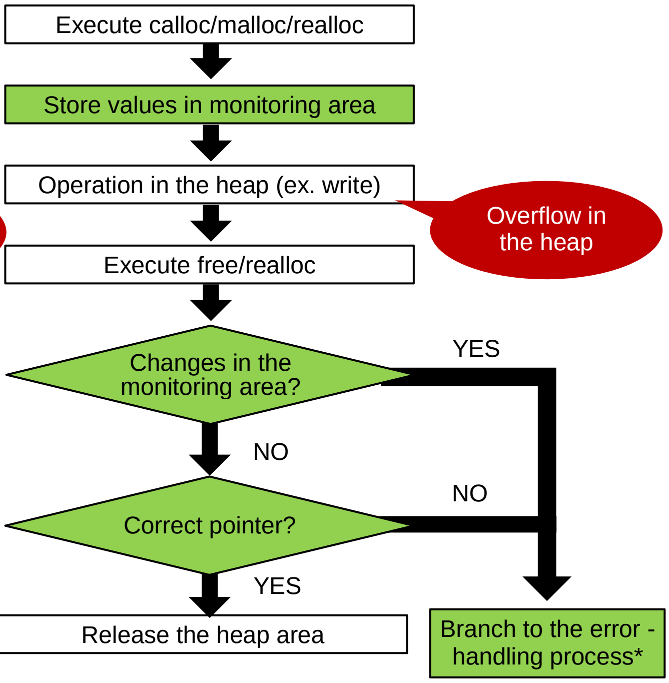

Execute calloc/malloc/realloc
Operation in the heap (ex. write)
Overflow in the heap
Execute free/realloc
Release the heap area
Program runaway/malfunction

Figure 1-1 Standard processing


Execute calloc/malloc/realloc
Store values in monitoring area
Operation in the heap (ex. write)
Overflow in the heap
Execute free/realloc
Changes in the monitoring area?
YES
NO
Correct pointer?
NO
YES
Release the heap area
Branch to the error - handling process*

*: The error-handling function can be defined by users.

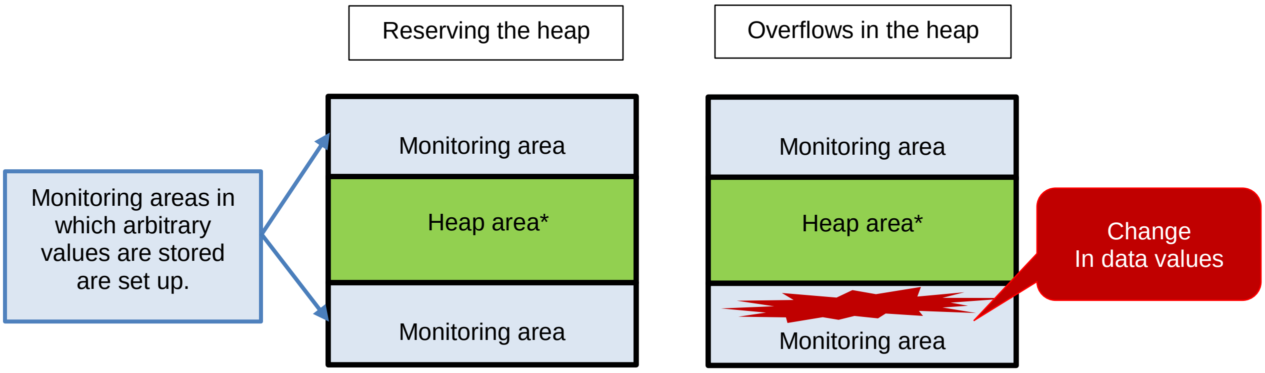
Figure 1-2 “Enhanced security for dynamic memory management functions” enabled

- Check monitoring areas

When reserving the heap area by using one of the following functions, monitoring areas in which arbitrary values are stored are set up before and after the given area of the heap. The values are checked when that part of the heap is released.

- calloc, malloc, or realloc

If, as a result of checking, there are changes in a monitoring area, the program will branch to the error - handling function (Note) instead of releasing the heap.



* : 2 bytes for CC-RL, 4 bytes for CC-RX and CC-RH

Figure 2 Monitoring area when reserving the heap and when the heap overflows

- Check the pointer to an area of the heap

When releasing the heap area, the following are checked regarding the pointers to that part of the heap. If a pointer is incorrect, the program will branch to the error-handling function (Note) instead of releasing the heap.

(a) Confirm that the pointer is to an area which was reserved by one of the following functions.

- calloc, malloc, or realloc

(b) Confirm that the pointer is not to an area that has already been released.

Note: The error-handling function can be defined by users.

1.2 Application Examples

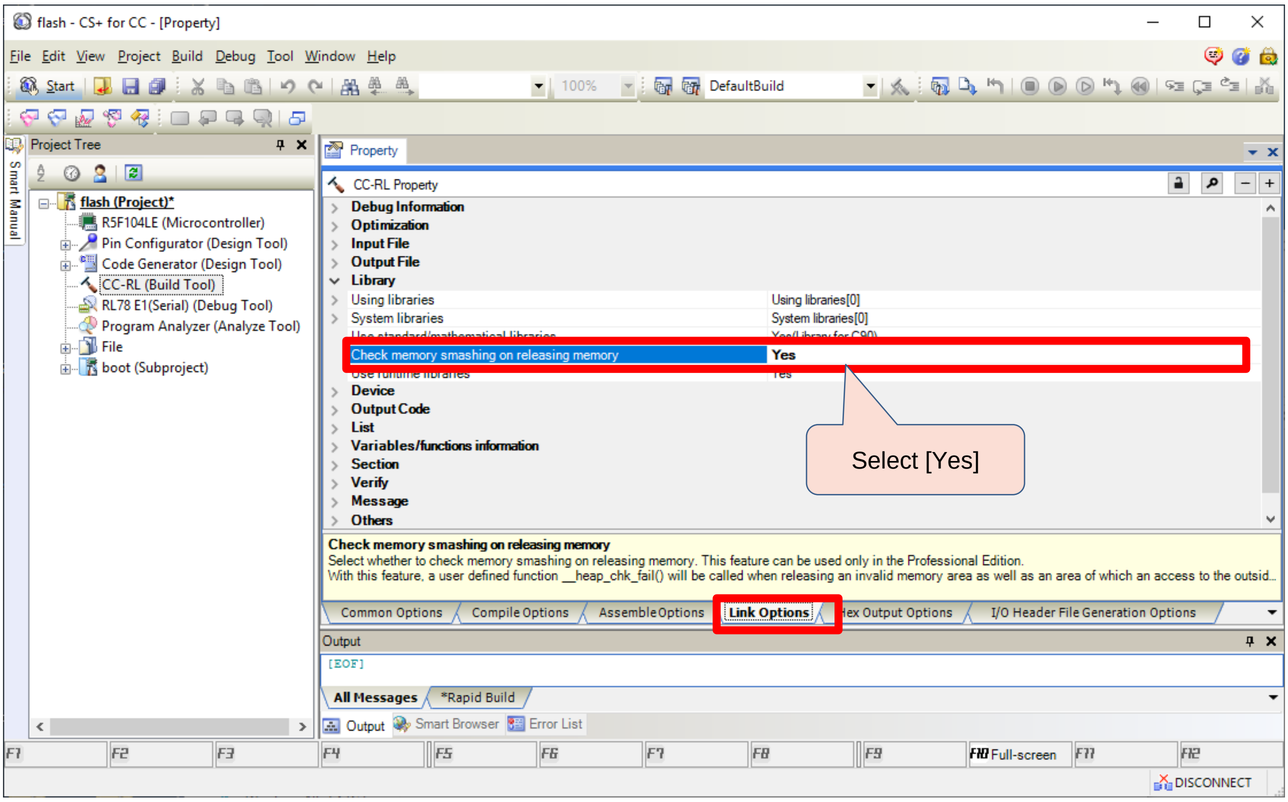
The “Enhanced security for dynamic memory management functions” can be easily enabled through GUI of integrated development environment (CS+ or e^2 studio).

- **Settings in IDE**

[For CS+]

(1) From the menu bar, click [Display] > [Project tree] and select [Build tools].

(2) In the [Property] tab > [Link Options] > [Library] category, select [Yes] in the property of [Check memory smashing on releasing memory].


flash - CS+ for CC - [Property]
File Edit View Project Build Debug Tool Window Help
Start
100%
DefaultBuild
Project Tree
flash (Project)*
R5F104LE (Microcontroller)
Pin Configurator (Design Tool)
Code Generator (Design Tool)
CC-RL (Build Tool)
RL78 E1(Serial) (Debug Tool)
Program Analyzer (Analyze Tool)
File
boot (Subproject)
Smart Manual
Property
CC-RL Property
Common Options
Compile Options
Assemble Options
Link Options
Hex Output Options
I/O Header File Generation Options
Output
[EOF]
All Messages
*Rapid Build
Output
Smart Browser
Error List
F1 F2 F3 F4 F5 F6 F7 F8 F9 F10 Full-screen F11 F12
DISCONNECT

Debug Information	
Optimization	
Input File	
Output File	
Library	
Using libraries	Using libraries[0]
System libraries	System libraries[0]
Use standard/mathematical libraries	Yes(Library for C90)
Check memory smashing on releasing memory	Yes
Use runtime libraries	Yes
Device	
Output Code	
List	
Variables/functions information	
Section	
Verify	
Message	
Others	

Check memory smashing on releasing memory
Select whether to check memory smashing on releasing memory. This feature can be used only in the Professional Edition.
With this feature, a user defined function __heap_chk_fail() will be called when releasing an invalid memory area as well as an area of which an access to the outsid...

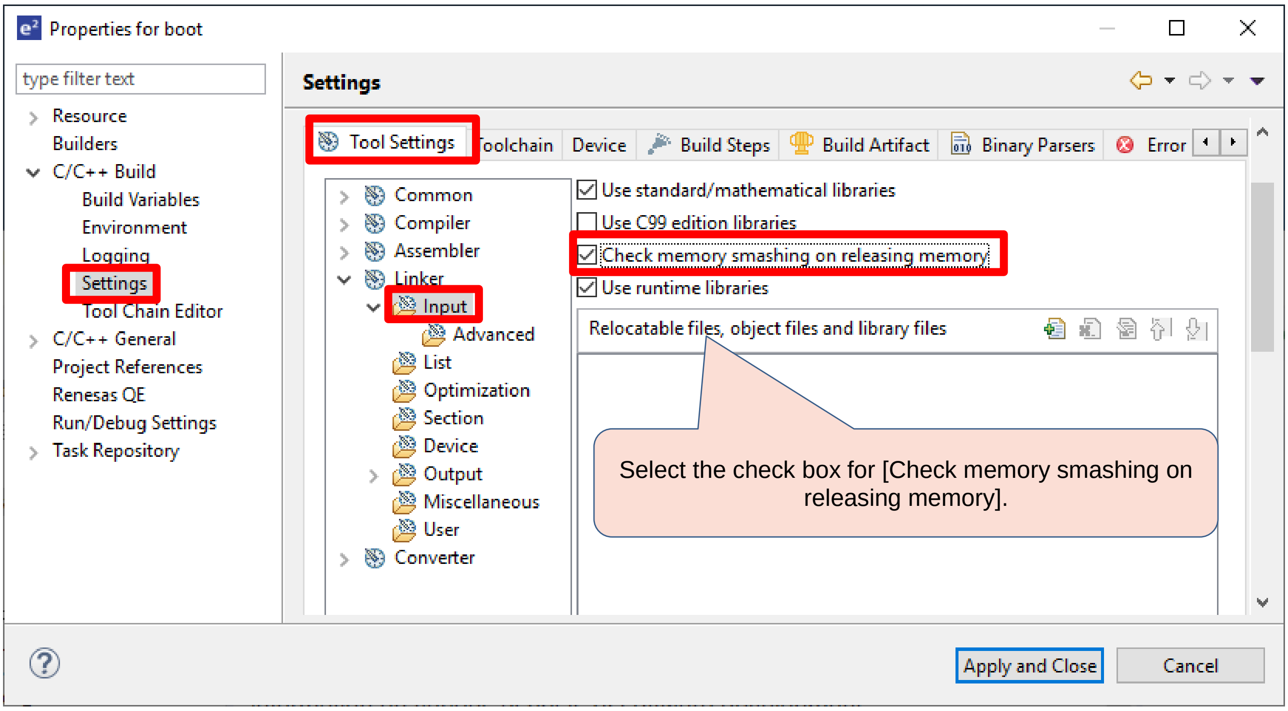
Select [Yes]

Figure 3 Settings for CS+ (CC-RL)

[For e^2 studio]

(1) From the menu bar, select [Project] > [Properties] to open the property dialog box.

(2) Select [C/C++ build] > [Settings] and then on the [Tool Settings] tab, click [Linker] > [Input] and select the check box for the [Check memory smashing on releasing memory].



☑ Use standard/mathematical libraries
□ Use C99 edition libraries
☑ Check memory smashing on releasing memory
☑ Use runtime libraries

Select the check box for [Check memory smashing on releasing memory].

Figure 4 Setting for e^2 studio (CC-RL)

■ Define the error-processing function

Define the __heap_chk_fail function which will be called for error processing when an invalid operation is detected. The processing of the __heap_chk_fail function can be defined by users.

Below is an example of a halt instruction in the error processing.

```
void __heap_chk_fail(void) {
// Describe content of error processing
  __halt();
}
```

2. More Features of Professional Edition

The professional edition offers more additional features as below.

See tool news and leaflet to learn more about each feature.

Checking against MISRA-C Rules
https://www.renesas.com/search/keyword-search.html#genre=document&q=r20ts0342 [Notification] Perform MISRA-C Rule Check During Compilation to Reduce Man-hours and Improve Quality for Program Development! Introducing MISRA-C Rule Checking Feature of Renesas Compiler Professional Edition
Synchronization Features in the Updating of Control Registers
https://www.renesas.com/search/keyword-search.html#genre=document&q=r20ts0347 [Notification] Automatic Insertion of Synchronization Processing to Reduce Man-hours for Development of RH850 Family! Synchronization Features in the Updating of Control Registers of Renesas Compiler Professional Edition
Detection of Stack Smashing
https://www.renesas.com/search/keyword-search.html#genre=document&q=r20ts0378 [Notification] Dynamic Checking for Corruption in Stack Area for Quality and Security Enhancement! Introducing Detection of Stack Smashing Feature of Renesas Compiler Professional Edition
Detection of illicit indirect function calls
https://www.renesas.com/search/keyword-search.html#genre=document&q=r20ts0438 [Notification] Prevent Illicit Indirect Function Calls and Improve the Quality of Your Program! Renesas Compiler Professional Edition Detection of Illicit Indirect Function Calls
Other Useful Features
https://www.renesas.com/search/keyword-search.html#genre=document&q=r20pf0024 Renesas compiler professional edition CC-RH professional edition also supports the following feature. ・ Half-precision floating point

For the usage of each feature, see the application note below.
The C source examples are also available for you to try simply by copy-pasting.

https://www.renesas.com/search/keyword-search.html#genre=document&q=r20ut4026
Renesas compiler Professional edition

3. Purchasing the Product

To order the product, contact your local Renesas Electronics sales office or distributor.

Users who already have the standard edition node-locked license can upgrade from the standard edition to professional edition by additionally purchasing the “upgrade (edition) license”. For details, see the following web pages for the compiler packages.

CC-RL: https://www.renesas.com/rl78_c

CC-RX : https://www.renesas.com/rx_c

CC-RH : https://www.renesas.com/rh850_c

Revision History

Rev.	Date	Description	
		Page	Summary
1.00	Sep.01.19	-	First edition issued

Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.

The past news contents have been based on information at the time of publication. Now changed or invalid information may be included.

URLs in Tool News also may be subject to change or become invalid without prior notice.

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu, Koto-ku, Tokyo 135-0061 Japan
www.renesas.com

Contact information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit:
www.renesas.com/contact/

TS Colophon 4.0