**T1** 2023

Coverity Scan Static Analysis Report

Hardhard Enterprises

Statement of Intent

Overview

This document aims to provide a record of static code analysis performed on a specific issue from the Coverity SAST scan for the NASA ION Open-Source code 4.1.1 project.

The primary purpose of this document is to validate the issue identified via the automated detection process to eliminate false positives.

Depending on findings, secondary purposes can include but are not limited to listing/providing recommended fixes alongside a list of attack vectors and potential exploits for consideration.

Reporting Best Practices

Please ensure best practices are kept when completing the document via regularly updating the Acronyms and Abbreviations table alongside any iterations made to the Document History table. This will allow other members to identify any updates and progress made across trimesters easily.

When using code snippets, please use screenshots that are clear and easy to read, alternatively, use words built-in code formatter found [here](https://appsource.microsoft.com/en-us/product/office/WA104382008?tab=Overview).

Document Naming Conventions

Naming conventions for this file are as follow; SAR\_{CID}. For example, when investigating issue 123456 the file name would be SAR\_123456.docx

Document History

|  |  |  |  |
| --- | --- | --- | --- |
| **Dates** | **Version** | **Author** | **Comments** |
| 12/05/2023 | V0.1 | Moe Khant Kyaw | Initial Document and analysis |
| 19/05/2023 | V0.2 | Moe Khant Kyaw | Finalizing Investigation and documentation |

Table of Content

Contents

[Introduction 3](#_Toc119848724)

[Objective 3](#_Toc119848725)

[Scope 3](#_Toc119848726)

[Acronyms and Abbreviations 3](#_Toc119848727)

[Code Review and Analysis 4](#_Toc119848728)

[Outcomes 4](#_Toc119848729)

[Observations 4](#_Toc119848730)

[Supporting Evidence 4](#_Toc119848731)

[Conclusions and Recommendations 4](#_Toc119848732)

[References 5](#_Toc119848733)

[Appendix 6](#_Toc119848734)

# Introduction

## Objective

The primary objective of this analysis is to determine whether the defects identified in the Coverity Report for the ION Open Source 4.1.1 project are:

* Indeed, defects.
* Potentially exploitable.

The secondary objective of this analysis, where applicable, is to provide the following:

* Recommendation(s) to fix.
* Any exploit for consideration.

## Scope

This static code analysis is limited to the ***High Impact Quality*** type defect identified in the following CIDs:  
***1520757***

# Acronyms and Abbreviations

Please keep an updated list of acronyms and abbreviations used throughout the report.

|  |  |
| --- | --- |
| **Acronym** | **Meaning** |
| DTN | Delay/Disruption Tolerant Network |
| ION | Interplanetary Overlay Network |

# Code Review and Analysis

## Overview

CID 1520757 has been marked as high impact issue, Out-of-bounds access type under Memory-Corruptions category by Coverity. This issue is marked as CWE-117: Improper restriction of Operations within the Bounds of a Memory Buffer, also known as buffer overflow, which happens when the program is trying to read or write outside of the limit of intended buffer.

## 

## Observations

This issue is found in “bsles\_clear\_event” function under “/bpv7/library/ext/bpsec/bpsec\_policy\_eventset.c” filepath. The function ‘bsles\_clear\_event’ is designed to remove a spectifc event from an event set. It starts by performing some checks to ensure the input parameters are valid. Then it clears the value in the event set’s mask, based on the event ID, is cleared and it iterates pver the list of events in the set. For each event, it checks if the event ID matches the specified ID. If there is a match, the event is removed by deallocating its memory, clearing the memory contents, and deleting it from the list. Finally, the function always returns 1, which indicates success, regardless of whether the event was actually removed or not. Overall, the purpose of the function “bsles\_clear\_event” is to remove a particular event from an event set by modifying the bitmask and remvoing it from the list.

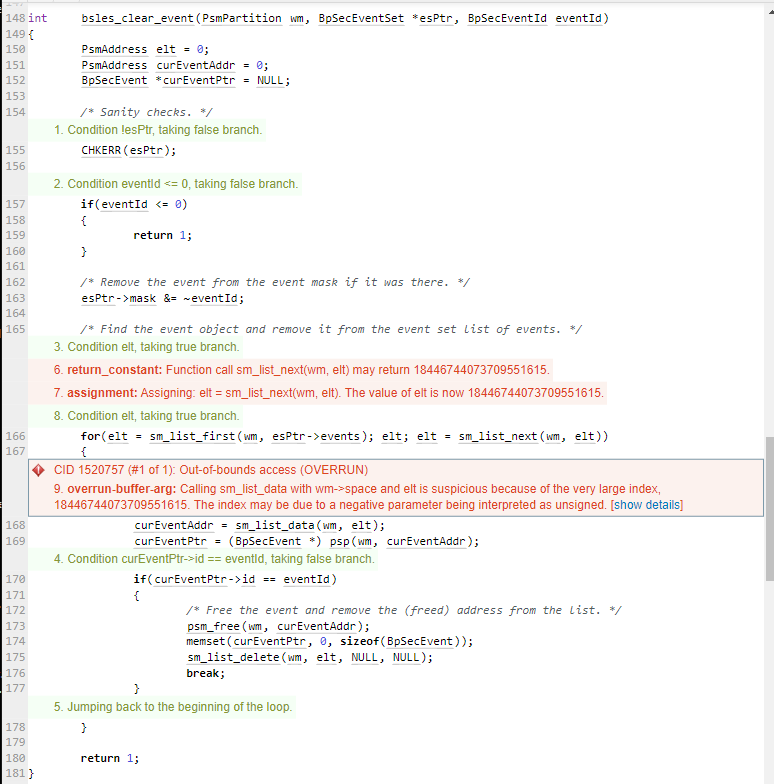


Figure 1: bsles\_clear\_event function

Regarding the issue highlighted by the static analyzer, it pointed out that the line “curEventAddr = sm\_list\_data(wm, let)” could lead to an out-of-bounds access due to a large index value, specially 18446744073709551615. This issue typically occurs when a negative parameter is mistakenly interpreted as an unsigned value, resulting in an unexpected index. To resolve this problem, it is crucial to carefully review the code and ensure that the initialized variables ‘wm’ and ‘elt’ are correctly defined and are within valid range.

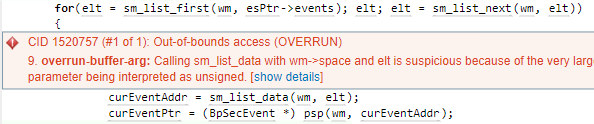


Figure 2: out-of-bound access error

# Conclusions and Recommendations

This issue can expose to risks such as memory corruption, data integrity issues, program crashes and security vulnerabilities. Attackers can exploit this to gain unauthorized access and perform malicious activities.

To prevent this issue from happening, we can do the followings:

* The validality of the ‘eventID’ value should be checked. Currently, the function onlychecks if it is less than or equal to 0. If ‘eventID’ should always be positive, a suitable error handling method should be implemented instead of returning 1. We could also consider returning an error code or taking required actions based on the needs of the function.
* The list iteration should also be validated. The code iterates over a list of events with the use of ‘sm\_list\_first’ and ‘sm\_list\_next’, however, there is a potential out-of-bounds access warning. The list operations should be reviewd and it is important to ensure that the variables ‘wm’ and ‘elt’ are correctly defined and initialized.
* The memory should also be managed properly. It is crucial to make sure the operations ‘psm\_free(wm, curEventAddr)’ used to clear the memory are suitable for freeing and clearing the event’s memory.

References

*Common weakness enumeration* (no date) *CWE*. Available at: https://cwe.mitre.org/data/definitions/119.html (Accessed: May 4, 2023).

Appendix

Include additional information/documentation here to help the readers understand complex information.