**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** |
| 14/05/23 | V0.1 | Nicholas Mag |  |
|  |  |  |  |
|  |  |  |  |

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 ***Out of Bounds Error*** type defect identified in the following CIDs:  
***CID-1520755***

***CID-1520753***

***CID-1520749***

# 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

This analysis deals with analysis 3 separate Out of Bound exception errors. These errors are all linked to eachother and it seems like the validation is the primary reason for this. The findings here may be used to broaden the net and encompass all of the out of bound errors that have been discovered.

## Observations

Within the validation of the code there is a value called ELT which refers to the PSMAddress. PSM is the personal space memory within the code, so ELT refers to the specific memory addressed. There is also a value called eltbuffer which is what validates whether the value is within the bounds of the allowasble array.

Graphical user interface, text, application, email

Description automatically generated

This is what gives us the error. The value being checked within the bounds is SM\_List\_Next here, and because of the value being outside the bounds it throws the error.

This is the case within all 3 CIDs as well as previous Out of Bounds CIDs that have been looked at.

Furthermore the error value that has been given in all the CIDs looked at are exactly the same. This is why the net can be thrown and encompass all CIDs regarding this error. The validation error is thrown because this one specific value is referring to the next PSMaddress which would then be out of the bounds of buffer.

## Supporting Evidence

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

A picture containing chart

Description automatically generated

Graphical user interface, text, application

Description automatically generated

# Conclusions and Recommendations

The recommendation of this report is to ultimely extend the bounds of the buffer allowing for correct validation of the value so that the error would no longer be thrown. It may be possible to look at dynamic memory allocation to help allocate the sizeable memory and not throw the error. It is possible that an Out of Bounds error can be used by a malicious party, but this one seems more like a bug or quirk within the code. And if the code is fixed then ultimately many of the out of bounds errors present would seese to be.

References  
Please keep an updated references list in APA7; The Deakin referencing guide can be found [here](https://www.deakin.edu.au/__data/assets/pdf_file/0009/2236752/Deakin-guide-to-APA7.pdf).

Appendix

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