**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** |
| 28/03/2023 | 1.0 | Dean Scanlon | Initial document. |
| 15/05/2023 | 1.1 | Dean Scanlon | Update |
|  |  |  |  |

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 access*** type defect identified in the following CIDs:  
***1520633***

# 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

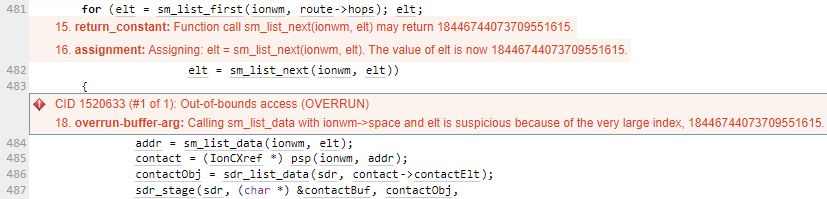
## Overview

When scanning ipnfw.c within the /bpv7/ipn directory of Bundle Protocol 7, Coverity was able to identify a **High Impact Quality** vulnerability. In this case, the issue related to Out-of-Bounds access as documented in CWE-119 (CWE – CWE-119: Improper Restriction of Operations within the Bounds of a Memory Buffer).

Within the **enqueueToEntryNode** function, a value is passed without being validated and as such is too large to be handled by an element index, thereby overflowing the allocated memory buffer.

## Observations

The **enqueueToEntryNode** function is part of the IPN forwarding module. It is responsible for enqueuing budles to the forwarding entry node’s output queue. The element index **elt** is erroneously passed the value 18446744073709551615 when the **sm\_list\_next** function calls **(ionwm, elt)**. The size of this element index is too large and as such encroaches beyond the allocated memory in the memory buffer. There is no validation when the value is passed and as such there is a possibility of a negative parameter being passed to the standard memory copy or allocation functions and being implicitly cast a large unsigned value.



*Fig.1. Coverity Static Analysis tool flag*



*Fig.2. Coverity flag with reasoning*

Issues that can arise from Out-of-bounds access errors

Out-of-bounds access errors can be considered a vulnerability in that they can result in system crashes or unauthorized access to private information as documented in CWE-119 (CWE – CWE-119: Improper Restriction of Operations within the Bounds of a Memory Buffer). In a worst-case scenario, attackers could potentially exploit buffer overflow vulnerabilities such as the one present in ION DTN BPv7 to change execution paths and execute code that can be tailored to damage applications, crash systems or expose private information. A best-case scenario for such code errors is a crash of the system which could negatively affect NASA projects that use the protocol.

## Supporting Evidence

Please provide any supporting evidence, and feel free to make references to documents in the appendix.

# 

# Conclusions and Recommendations

The C language does not automatically validate the locations to which memory buffers are assigned and lacks a ‘garbage collection’ type memory management scheme as in Java or Python. As such, efforts need to be made to validate the values being passed to the elt variable. A function should be included in the code which validates the elt function returns 1 or no greater than the number of elements in the linked list **sm\_list\_data**. This would avoid elt taking a negative value and as a result being interpreted as an unsigned integer.

An example, albeit out of context, of a function that could be used to validate the values of elt is shown in Fig.3.

Text

Description automatically generated

*Fig.3. Generic function for code validation*

Alternatively, consideration could be made for the NASA DTN protocol to be written in another programming language such as Python or Java.

References  
CWE-119: Improper Restriction of Operations within the Bounds of a Memory Buffer (2023 January) Common Weakness Enumeration <https://cwe.mitre.org/data/definitions/119.html>

Appendix

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