**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** |
| 9/09 | V0.1 | John-Eddie Cubis | Draft |
| 10/09 | V0.2 | John-Eddie Cubis | Final |
|  |  |  |  |

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 ***Untrusted value as argument*** type defect identified in the following CIDs:

[***15209641***](https://scan7.scan.coverity.com/reports.htm#v45706/p15622/fileInstanceId=124898899&defectInstanceId=18495916&mergedDefectId=1520641&eventId=18495916-32)

# 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

Analyised is segment issuance function, this function validates segments being received during transmission over large interplanetary networks. The bug/vulnerability is found in the ‘**serialiseCancelSegment**’ and is responsible for constructing the serialized representation of a cancel segment, considering the origin (sender or receiver) and the associated header information before appending the reason code to the buffer.

## Observations

Typically networking requires error checking in order to protect from outside attackers coming into the network. Tainted values can come from untrusted sources may potentially lead to security or functional issues if not handled correctly.

‘**segment->pdu.segTypeCode’**, ‘**segment->remoteEngineId’**, and ‘**segment->pdu.reasonCode’** could all be influenced by external inputs or data from untrusted sources.



If any of the tainted data (e.g., incorrect segTypeCode, malicious remoteEngineId, or tampered reasonCode) is not properly validated, sanitized, or checked for validity before being used in this function, it can propagate to the serializeHeader function, leading to potentially incorrect or insecure serialization.

## 

## Supporting Evidence

[CWE-20: Improper Input Validation](https://cwe.mitre.org/data/definitions/20.html)

[CWE-116: Improper Encoding or Escaping of Output](https://cwe.mitre.org/data/definitions/116.html)

# 

# Conclusions and Recommendations

To mitigate risk of tainted data we can implement security measures in code:

Validate inputs: Ensure all input values, ‘segTypeCode’,’remoteEngineID’, and ‘reasonCode’ are properly validated and sanitised

Range and validity checks: Ensuring all values are within the range of expected validity criteria.

Security awareness: ensure secure coding practices.

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.