**T2** 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** |
| 11/09/2023 | V1.0 | Damon Willmott | Initial document |
| 12/09/2023 | V1.1 | Damon Willmott | Finalisation |
|  |  |  |  |

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 ***Structurally Dead Code*** type defect identified in the following CIDs:  
***CID 1520726***

# 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 |
| CID | Coverity Issue Identification number |
| CWE | Common Weakness Enumeration |
| BCB | Block Confidentialy Block |

# Code Review and Analysis

## Overview

Static Analysis by Coverity categorieses CID-1520726 as a **Medium** impactvulnerability, and depicts it as a **Structurally Dead code** error. This issue is described in CWE-561 relates to ‘dead code’, which is code that is impossible to execute within a running program as a result of the surrounding code, meaning the that code is flagged as irrelevant to the proper functioning of the program as a whole. The section of code analysed for this particular issue has the purpose of receiving the number of failed Tx BCB (Block Confidentiality Block) bytes from SRC within the Bundle Protocol.

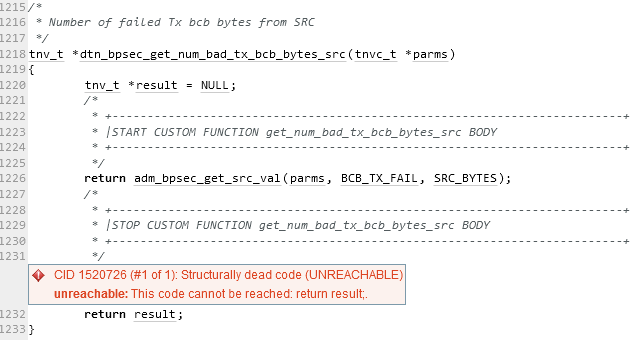
## Observations

This issue can be observed in various functions in the file, which is located within the /bpv7/adm/adm\_bpsec\_impl.c file directory. The role of this file is to monitor various statistics within the Bundle Protocol, such as monitoring security policies and bundle performance, as well as implement and retrieve functions from the ‘BPsec’ Administration API, which can retrieve statistics such as the number of blocks that have successfully or failed to transmit.

The occurrence of this issue appears in several sections of code in the file, and the majority of them originate from the same cause, that being the ‘return result’ statements located at the end of most of the functions that display this error. This particular issue identified by Coverity can be found in the dtn\_bpsec\_get\_num\_bad\_tx\_bcb\_bytes\_src function, on line 1232. This function is designed to provide information on how many bytes had failed to transmit within the BCB during communication (potentially due to being corrupted or invalid).

The error occurs because an unnecessary return statement has been included at the end of the function, which is not being executed because the function has already been ceased by a previous return statement, hence Coverity is highlighting the second return statement as structurally dead code.

## Supporting Evidence

The following highlights Coverity flagging the problematic code. Here, the second return statement is considered structurally dead because the previous return statement has already ceased the functions operation, and thus the second returen statement cannot be executed, making it redundant in the function.

# Conclusions and Recommendations

This issue is simply a bug in the code, and does not pose an immediate threat to the security of the program.

The recommendation for this error would be to simply remove the second return statement to ensure that all redundant and unnecessary code is removed to ensure good coding practises.

References  
Common Weakness Enumeration (n.d.) *CWE-561:* *Unchecked Return Value*, <https://cwe.mitre.org/data/definitions/561.html>

Appendix

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