



SECURE CODING ANALYSIS OF AN AADL CODE GENERATOR'S RUNTIME SYSTEM

David Keaton

September, 2015

Abstract

Architecture Analysis and Design Language (AADL) is a foundation for creating model-based reliable systems. Its roots are in the safety community, specifically transportation engineering. The conditions for assuring safety and security often overlap, but they are not identical. As part of an investigation into using AADL for security applications, this paper describes a secure coding analysis of the PolyORB-HI-C runtime system used by C language code output from the Ocarina AADL code generator. The overall quality of the code is found to be high. However, several potential out-of-bounds stores were discovered, which opens up the possibility of buffer overflow attacks. The techniques for finding these situations are described, along with recommendations for their elimination and prevention.

1. Introduction

SAE International has standardized Architecture Analysis and Design Language (AADL) [SAE 2012] for the purpose of allowing engineers to create a model of a hardware/software system that can ensure all the requirements of the system are met before it is built. Typically, this has included safety requirements because SAE International specializes in the transportation industry. The current investigation looks at the implications of applying AADL to security requirements.

Safety means that a system will not harm its user, and security means that the user cannot harm the system. Consequently, there may be slightly differing requirements when security is considered. For example, vulnerabilities that potentially lead to abnormal termination of a program may be of higher importance for safety, while potential buffer overflows may be of higher importance for security.

To make AADL more useful by extending it deeper into the design and development phases of a project, several code generators have been developed. These code generators translate an AADL-described architecture into a program in a language such as C or Ada so that the software portion of the system's architecture will automatically match the model. It is therefore helpful to examine the interaction between code generation and security.

This paper examines one aspect of the implications of security for code generation. For an AADL application to be secure, its generated code must be secure, as must any runtime support software on

which the generated code depends. The current project used CERT's Source Code Analysis Laboratory (SCALe) to evaluate the secure coding robustness of the PolyORB-HI-C runtime support system for C language code generated by the Ocarina AADL code generator. The purpose is not to find all possible flaws in the runtime code, but to concentrate on those types of statically-analyzable coding practices and errors that in the past have led to security vulnerabilities in other software.

SCALe checks software against the set of rules in the CERT Secure Coding Standards, in this case *The CERT C Coding Standard, 2nd ed* [Seacord 2014]. Typically no one analysis tool will find all secure coding violations in a program. SCALe addresses this problem by using several tools and combining the results to provide better coverage.

2. AADL Code Generators

The following tools are able to generate C code from AADL.

Ocarina

Ocarina [ISAE 2015] is open source and can generate either Ada or C for several operating systems, including POSIX [ISO 9945 2011]. Its runtime systems are PolyORB-HI-Ada and PolyORB-HI-C.

RAMSES

RAMSES [TPT 2015] is open source and generates C code for ARINC653 and OSEK operating systems.

UCaG

UCaG [Gui 2008] is an academic project generating C code for Delta OS.

2.1 Runtime Code Overview

Ocarina was chosen for this study because it is readily available open-source code that can generate C for POSIX, making its output ideal for analysis using SCALe, which works on either Linux or Windows. Ocarina's C runtime system, PolyORB-HI-C, was analyzed.

The runtime system uses several header files that are generated by Ocarina for each translation of AADL into C. Consequently, PolyORB-HI-C must be recompiled separately for each such translation. To analyze the runtime code, the header files must be available just as if the code were being compiled, which means that analysis must be performed in the context of a sample AADL project that has been translated into C.

PolyORB-HI-C provides a set of example AADL projects for translation. These examples are designed to make use of all the general facilities of the runtime system, and therefore all of these can be analyzed.

Some target-specific features, low-level interfaces for the device drivers, cannot be analyzed because there are no examples that use them.

For this project, all the PolyORB-HI-C examples were compiled together with the runtime system on Mac OS X (to test the build process to ensure that the code could be analyzed) and on Linux (both to test the build process and to analyze the code with SCALE). On OS X, Ocarina's Makefile (the machine-readable instructions for building Ocarina) had to be modified to complete the final step of the build. On Linux, one example file, producer-consumer.c, had to be modified to include the time.h standard header, and one runtime file, po_hi_transport.c, had to be modified to include the stddef.h standard header before the software would build.

This indicates that PolyORB-HI-C and its examples were modified since the last time they were tested on OS X or Linux, or that those environments have changed since the last time PolyORB-HI-C was tested on them. This is a typical issue when a large software project has multiple target environments.

Table 1 describes the size of the analyzed codebase and Table 2 explains the headers. The code for the examples includes both hand-written C code and Ocarina-generated C code.

Table 1: Code Size Metrics

Portion of Code	Files	LoC	sigLoC	Size
PolyORB-HI-C analyzed	69	15355	9325	471.8
Examples	287	29373	17911	602.2
Total analyzed	356	44728	27236	1074.0
PolyORB-HI-C unanalyzable	9	1404	940	44.9

Table 2: Code Size Metrics Headers

Heading	Definition
Files	Number of C files in each portion of the code
LoC	Lines of source code
sigLoC	Lines of significant source code (without blank lines and comments)

Size	Size, in kilobytes of C source code, ignoring non-code files
-------------	--

3. PolyORB-HI-C Findings

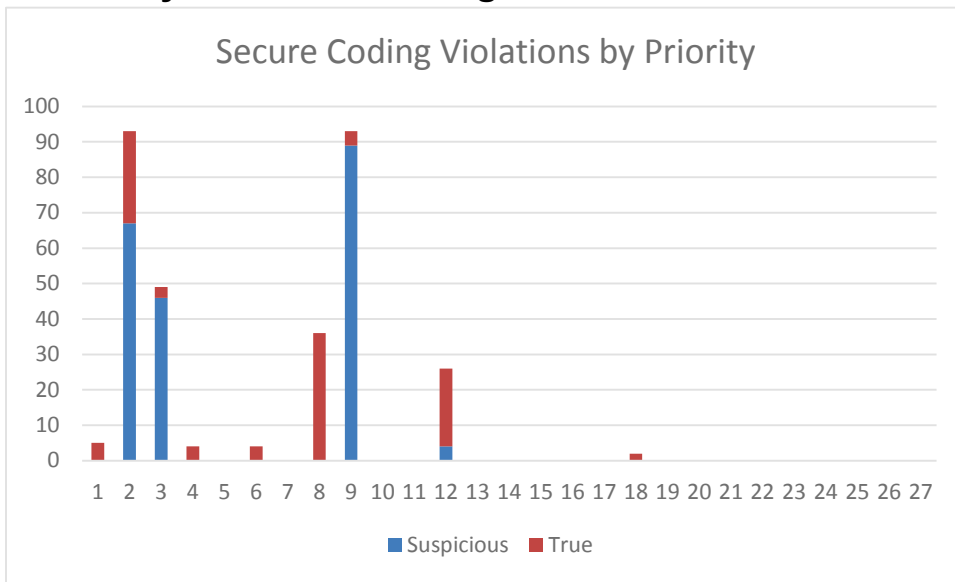


Figure 1: Higher priority is either urgent or inexpensive to fix. The maximum priority is 27.

Key finding: Most violations are lower on the priority scale.

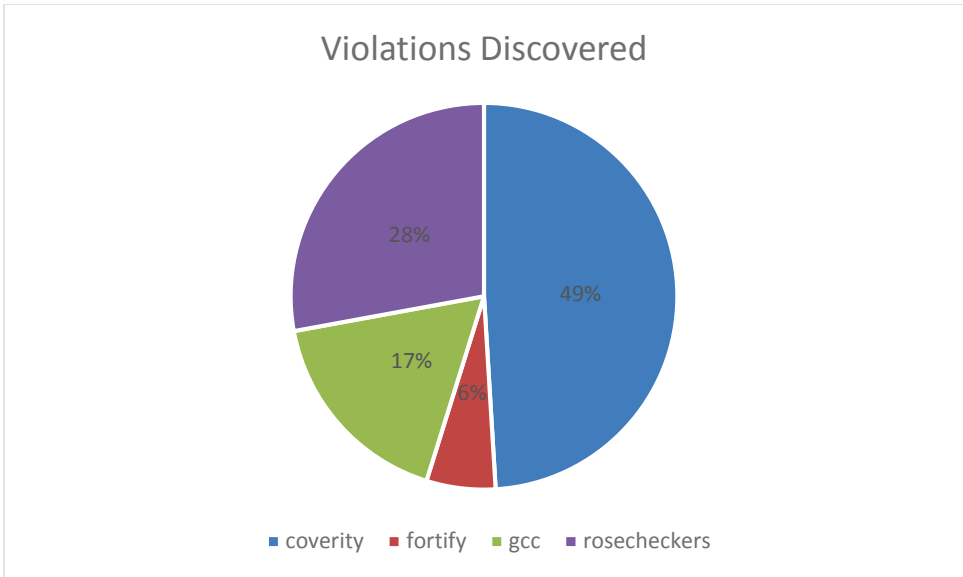


Figure 2: Secure Coding Violations Discovered by Tool

Key finding: All Linux-based tools available to SCALE were helpful in identifying violations.

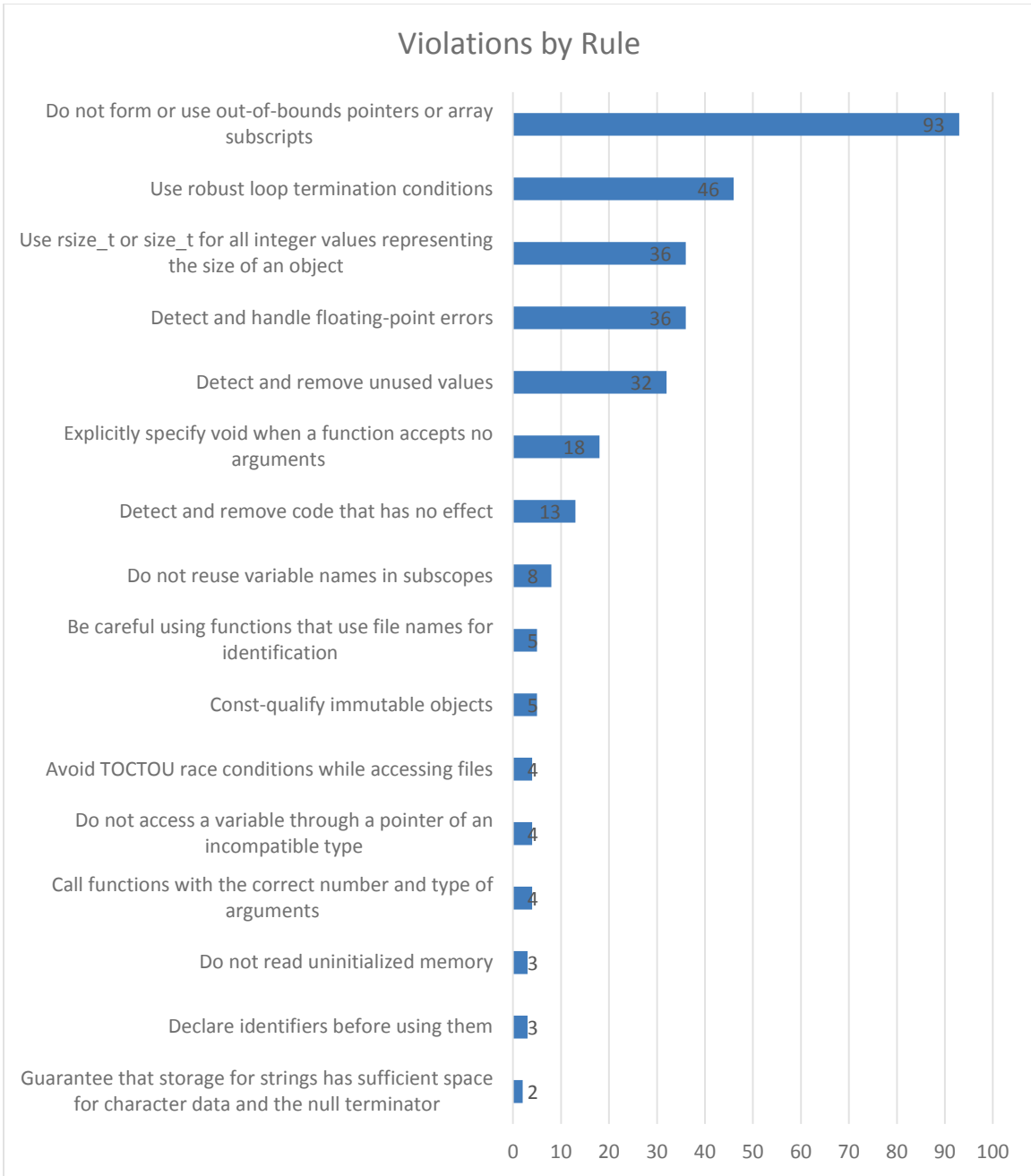


Figure 3: Secure Coding Violations by CERT Rule

Key finding: The largest category is out-of-bounds pointers.

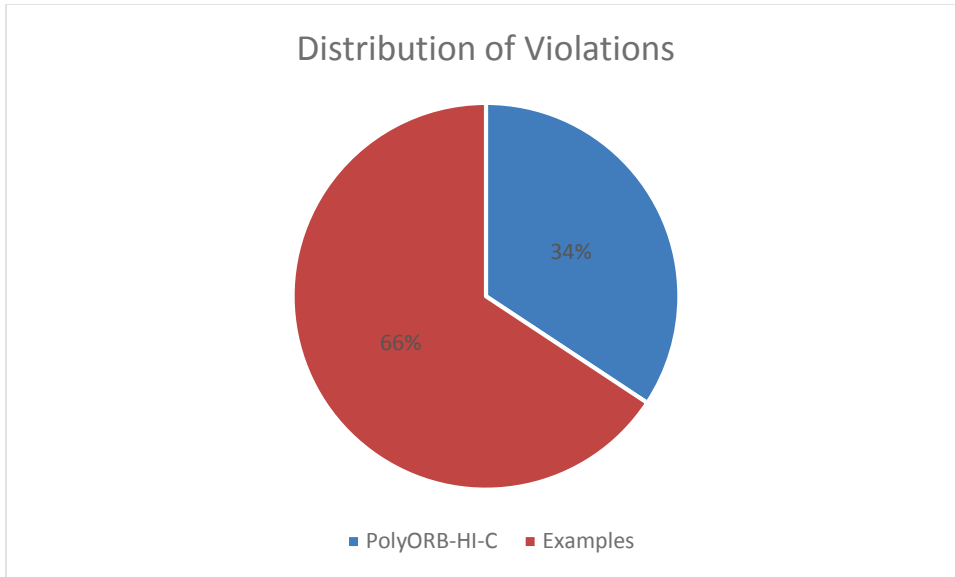


Figure 4: Secure Coding Violations by Portion of Code

Key finding: The runtime system itself comprises 34% of the significant lines of code and 34% of the secure coding violations. The violation density is uniform.

As noted in Section 6, the priority is the product of three metrics that measure the severity of the violation, the likelihood that the violation can be exploited, and the cost of remediating the violation. The maximum priority is theoretically 27, indicating a severe vulnerability that is most likely to be exploited and is least expensive to fix.

Table 3 lists some relevant summary metrics about this codebase in comparison with some other C and C++ codebases that have been audited using SCALE. The average and standard deviation numbers were originally presented in [Svoboda 2015]. Table 4 explains the summary statistics headers.

Table 3: Audit Summary Statistics

	Files	kLoC	ksigLoC	Rules	True	Susp	FileDens	LineDens
PolyORB-HI-C analyzable plus examples	356	44.7	27.2	16.0	106.0	206.0	0.9	11.5
Average	7606	4482.4	3237.1	19.3	99.0	6202.0	42.0	76.1
Std Dev	12516	7618.2	5497.5	8.1	54.7	9757.2	71.3	108.8

Table 4: Audit Summary Statistics Headers

Heading	Definition
kLoC	Lines of source code /1000
ksigLoC	Lines of significant source code /1000 (without blank lines and comments)
Rules	Number of CERT rules that were violated
True	Number of true violations
Susp	Number of suspicious violations
FileDens	Ratio of violations per file: number of diagnostics/number of files
LineDens	Ratio of violations per code size: diagnostics/ksigLoC

Key finding: This codebase has a much lower defect density than average. The code quality is significantly above average.

3.1 Future Work

The spreadsheet in the appendix presents the known true violations first, followed by suspicious diagnostics, and then is sorted from highest priority to lowest priority. The diagnostics that occur earlier are more urgent and easier to fix than the diagnostics that occur later. Therefore, attending to the diagnostics in the order in which they appear within each category (true or suspicious) is recommended.

Suspicious diagnostics are those that appear to be true violations, but a deeper knowledge of the code is required in order to be certain. Suspicious diagnostics that turn out not to be actual violations may be ignored.

Furthermore, some diagnostics indicate code that may or may not be vulnerable due to external circumstances. For example, many concurrency diagnostics would not apply to code that is never run in a multithreaded environment. Likewise, some diagnostics apply to code only when it is run on certain platforms (such as 64-bit Linux). These diagnostics may be ignored if the code is only to be run on platforms where the code is not vulnerable.

4. Analysis of Findings

This section provides an in-depth analysis of some of the confirmed diagnostics listed in the previous section. The following discussions explain why the code in question violates the rule, but the discussions do not attempt to explain the rules themselves because they are meant to be self-contained, and each rule provides ample rationale for its purpose. Every rule in the CERT coding standard has a page devoted to it on the CERT wiki, and at the bottom of each page is a section where the public can post comments related to the rule. Issues about the validity of any rule should be posted to the rule's Comments section. The CERT Division welcomes feedback about the rules and about the validity of each diagnostic.

4.1 Violation: Out of Bounds Array Subscript

`src/po_hi_transport.c` has the following code:

```
43 __po_hi_transport_sending_func
__po_hi_transport_devices_sending_funcs[__PO_HI_NB_DEVICES];
...
229 __po_hi_transport_sending_func __po_hi_transport_get_sending_func (const
__po_hi_device_id device)
230 {
231     if (device > __PO_HI_NB_DEVICES)
232     {
233         return NULL;
234     }
235
236     return __po_hi_transport_devices_sending_funcs[device];
237 }
```

On line 43, the array `__po_hi_transport_sending_funcs` is declared with a bound of `__PO_HI_NB_DEVICES`, which means the elements of the array are indexed from zero to `__PO_HI_NB_DEVICES - 1`. However, the test on line 231 checks that the array index is greater than the bound. This allows line 236 to read one element past the end of the array.

This code violates CERT rule *ARR30-C Do not form or use out-of-bounds pointers or array subscripts*.

Solution: Adjust the Input Validation Test

The off-by-one error can be eliminated by changing test on line 231 from `>` to `>=`.

4.2 Violation: TOCTOU Race Condition

The function `__po_hi_storage_file_create` in `src/po_hi_storage.c` contains the following code:

```
115     if (stat (file->filename, &ss) == 0)
116     {
117         __DEBUGMSG ("[STORAGE] __po_hi_storage_file_create: file %s already
exists\n", file->filename);
118         return __PO_HI_ERROR_EXISTS;
119     }
120
121     /*
122     * We assume the file is not open previously by a call to open().
123     * Otherwise, we assume this is an error.
124     */
125     if (file->fd != -1)
126     {
127         __DEBUGMSG ("[STORAGE] __po_hi_storage_file_create: file already
opened (%d)\n", file->fd);
128     }
129
130     fd = open (file->filename, O_RDWR | O_CREAT | O_SYNC, S_IRWXU |
S_IRGRP | S_IROTH);
131
132     if (fd == -1)
133     {
134         __DEBUGMSG ("[STORAGE] Warning, cannot open file %s with create
attributes\n");
135         return __PO_HI_INVALID;
136     }
```

The test on line 115 uses `stat` to test for the presence of a file. The same filename is used to open a file on line 130. In between lines 115 and 130, there is a Time-Of-Check/Time-Of-Use (TOCTOU) race condition, a window of vulnerability where a symbolic link could be created by another user (an attacker) using the same filename. This could cause the application to write to an unintended file, which could cause an information leak or modify an existing file to which the application has access.

This code violates CERT recommendation *FIO01-C Be careful using functions that use file names for identification*.

Solution: Open with `O_CREAT|O_EXCL`

The call to `open` on line 130 already includes the `O_CREAT` flag. Adding the `O_EXCL` flag would cause the open to fail if the file already exists, which is what the call to `stat` was trying to achieve. Lines 115-119 could then be deleted, and the race condition would be eliminated.

4.3 Violation: Insufficient Storage Space for Data

In `src/drivers/po_hi_driver_sockets.c`, the following code appears inside the function `__po_hi_driver_sockets_init`:

```
443     struct sockaddr_in     sa;
...
587         hostinfo = gethostbyname ((char*)ipconf->address);
588
589         if (hostinfo == NULL )
590         {
591             __DEBUGMSG ("[DRIVER SOCKETS] Error while getting host
informations for device %d\n", dev);
592         }
593
594         sa.sin_port = htons (ip_port);
595         sa.sin_family = AF_INET;
596
...
607         tmp = (char*) &(sa.sin_addr);
608         for (i=0 ; i<hostinfo->h_length ; i++)
609         {
610             tmp[i] = hostinfo->h_addr[i];
611         }
```

On line 587, the call to `gethostbyname` may return either IPv4 or IPv6 addresses. However, lines 607-611 unconditionally copy the result into a `sockaddr_in` structure, which only holds IPv4 addresses. Specifically, the `sin_addr` field is four bytes wide, which is too small to hold an IPv6 address. In the case of IPv6, `hostinfo->h_length` will be 16 bytes and an overflow of `sa.sin_addr` will occur.

This code was flagged as violating CERT rule *STR31-C Guarantee that storage for strings has sufficient space for character data and the null terminator*. A null terminator does not apply in this case, but there is insufficient space for the data when an IPv6 address is returned. This code is also a violation of CERT rule *ARR30-C Do not form or use out-of-bounds pointers or array subscripts*.

Solution: Test the Address Length

A thorough solution providing full support for IPv6 may require significant changes to large portions of the code. However, at a minimum, the result of `gethostbyname` must be tested to see if its address length is greater than four bytes. A temporary fix could reject any addresses that are larger than IPv4 so they would not overflow the buffer.

5. Diagnostic Findings

The analysis results are provided in the spreadsheet in the appendix. The “true” violations indicate flagged nonconformities that were verified to be violations of the *CERT C Coding Standard*. The “suspicious” diagnostics indicate flagged nonconformities that appear to be violations but must be verified by a deeper study of the source code. This section documents the contents of the spreadsheet.

5.1 Diagnostics

The spreadsheet contains the columns listed in Table 5:

Header	Definition
Verdict	Indicates whether a diagnostic violation is a known true violation or suspicious
Path	Path name to the source file
Line	Line number where violation occurs
Message	Diagnostic message describing the violation
Rule	ID of the CERT guideline that is violated

Some violations may have two or more diagnostic messages. For instance, a checker that warns of the use of an uninitialized variable might provide two messages. The first would be at the location where the variable is declared, and the second would indicate the location where the variable is read (while never being initialized).

6. Procedure

C can be analyzed by an extensive number of *static analysis* (SA) tools. CERT’s experience with C static analysis tools has led us to the conclusion that each SA tool has its own strengths and weaknesses, and every tool can detect faults undetectable by other tools. Consequently, running only one SA tool is likely to miss many faults that other tools can detect.

CERT therefore employs a coverage analysis technique, where we use several SA tools to detect vulnerabilities and merge their results. This technique has several advantages, the biggest one being that we minimize the risk of overlooking critical vulnerabilities (that is, false negatives). Because of the different strengths of different tools, we can also gain new perspectives on vulnerabilities identified by multiple analyzers.

Many tools rely on the assumption that it is more prudent for an SA tool, when encountering some questionable code, to report it as a potential vulnerability than to ignore it. This assumption also enables a security analyst to manually inspect the code and confirm the vulnerability or eliminate it. It minimizes the possibility of *false negatives*, that is, uncaught vulnerabilities. However, it does increase the number of false positives; that is, code constructs that might be vulnerable, but turn out to be perfectly legitimate when taken in their total context.

Several tools yield many false positives. Validating each of these diagnostics requires an inspection of the code in question, but sometimes it is necessary to inspect the entire function or file containing the code, or all functions that invoke the function containing the questionable code. Consequently, an auditor has no hope of thoroughly validating each and every diagnostic that may be generated by an automated SA tool.

6.1 CERT Secure Coding Rules

An essential element of secure coding in any programming language is well-documented and enforceable coding standards. Coding standards encourage programmers to follow a uniform set of rules and guidelines determined by the requirements of the project and organization, rather than by the programmer's familiarity or preference. Once established, these standards can be used as a metric to evaluate source code (using manual or automated processes).

The CERT Division has published *The CERT C Coding Standard, 2nd ed.* This book provides rules and recommendations for secure coding in the C programming language. The goal of these rules and recommendations is to eliminate insecure coding practices. The application of the secure coding standard will lead to higher quality systems that are robust and more resistant to attack. This coding standard affects the wide range of products coded in C, such as PCs, game players, mobile phones, home appliances, and automotive electronics. It is designed specifically for code conforming to the C standard [ISO 9899 2011], with some support for POSIX. The CERT Division provides certification for code that is conformant with *The CERT C Coding Standard*, which is available at the following web address:

<https://www.securecoding.cert.org/confluence/x/HQE>

This standard consists of nearly 300 rules and recommendations. Coding practices are defined to be rules when the following conditions are met:

1. Violation of the coding practice is likely to result in a security flaw that may result in an exploitable vulnerability.
2. Conformance to the coding practice can be determined through automated analysis, formal methods, or manual inspection techniques.

Implementation of the secure coding rules defined in this standard are necessary (but not sufficient) to ensure the security of software systems developed in the C programming language.

Recommendations are guidelines or suggestions. Coding practices are defined to be recommendations when all of the following conditions are met:

1. Application of the coding practice is likely to improve system security.
2. One or more of the requirements necessary for a coding practice to be considered a rule cannot be met.

The set of recommendations that a particular development effort adopts depends on the security requirements of the final software product. Projects with high-security requirements can dedicate more resources to security and are consequently likely to adopt a larger set of recommendations.

To ensure that the source code conforms to this secure coding standard, it is necessary to have measures in place that check for rule violations. The most effective means of achieving this conformance is to use one or more static analysis tools. Where a rule cannot be checked by a tool, then a manual review is required.

6.1.1 Risk Assessment

Each guideline has an assigned priority. Priorities are assigned using a metric based on Failure Mode, Effects, and Criticality Analysis (FMECA) [IEC 60812 2006]. Three values are assigned for each guideline on a scale of 1 to 3 for

severity – how serious are the consequences of the guideline being ignored

1 = low (denial of service attack, abnormal termination)

2 = medium (data integrity violation, unintentional information disclosure)

3 = high (run arbitrary code, privilege escalation)

likelihood – how likely is it that a flaw introduced by ignoring the guideline could lead to an exploitable vulnerability

1 = unlikely

2 = probable

3 = likely

remediation cost – how expensive it is to comply with the guideline

1 = high (manual detection and correction)

2 = medium (automatic detection and manual correction)

3 = low (automatic detection and correction)

The three values are then multiplied together for each guideline. This product provides a measure that can be used in prioritizing the application of the guidelines. These products range from 1 to 27. Guidelines with a priority in the range of 1-4 are level 3 guidelines, 6-9 are level 2, and 12-27 are level 1. As a result, it is possible to claim level 1, level 2, or complete compliance (level 3) with a standard by implementing all guidelines in a level.

6.2 Diagnostic Categorization

Fortunately, many vulnerabilities rely on a relatively small number of errors in coding technique, and many SA tools rely on a handful of heuristics to identify vulnerabilities. SA tools typically provide their own categorization of diagnostics and often assign a unique identifier for each diagnostic category. Furthermore, the diagnostics produced by SA tools can be associated easily with CERT secure coding guidelines, where a valid diagnostic indicates a violation of the associated CERT guideline. While our SA tools produced many diagnostics, these diagnostics could be classified into violations of a few secure coding guidelines.

Therefore, our approach involves collecting all diagnostics produced by all of the SA tools at our disposal and classifying them by the secure coding guideline with which they can be associated. Within each secure coding guideline, normally a representative sample of diagnostics is examined. However, in studying PolyORB-HI-C, all of the diagnostics were examined. Any diagnostic that turns out to be a *true positive*, that is, indicates a true vulnerability in the code, is added to a table of confirmed vulnerabilities.

Some diagnostics are labeled as suspicious. In the evaluation of PolyORB-HI-C, this label signifies either

1. that a diagnostic is believed to indicate a true violation of a guideline, but a definite determination requires a more comprehensive knowledge of the code than the auditor possesses, or
2. that a diagnostic indicates a true violation of a guideline but this has no effect because the questionable function is a stub, and the code will need to be reexamined when the function's implementation is completed.

In the former case, the code might possibly be safe but difficult to analyze. In the latter case, the code is currently safe but might not remain so once it is completed.

In either case, the code merits attention, and should probably be modified. It is likely that the code may be passed to a maintainer who fails to understand the code and makes incorrect assumptions about its security. Such an occurrence increases the maintenance costs of the code, as the maintainer might modify it unnecessarily, or might use it improperly, creating one or more new vulnerabilities.

The appendix of this report provides the complete table of confirmed and suspicious diagnostics, providing details associated with each.

6.3 Static Analysis Tools

The SCALe analysis employed the following SA tools, as described below.

6.3.1 Fortify 360 SCA

Fortify 360 is a commercial product developed by Fortify Software (now a part of Hewlett Packard). The product provides an extensive suite of tools for software security assurance. SCALe focuses on the *static code analyzer* (SCA) tool. It can be used to analyze software written in C, C++, Java, .NET, ASP.NET, Cold-Fusion, “Classic” ASP, PHP, VB6, VBScript, JavaScript, PL/SQL, T-SQL, and COBOL, as well as configuration files. More information on Fortify SCA is available at

<http://www8.hp.com/us/en/software-solutions/static-code-analysis-sast/>

6.3.2 Coverity Prevent

Coverity Prevent is a commercial product developed by Coverity, Inc. (now a part of Synopsys). The product also provides an extensive suite of tools for software security assurance. SCALe focuses on the Coverity Static Analysis tool, which can be used to analyze software written in C, C++, Java, or C#. More information on Coverity is available at

<http://www.coverity.com>

6.3.3 Rosecheckers

The Rosecheckers project has been internally developed at the CERT Division to provide a static analysis tool for analyzing C and C++ code. The project was designed to enforce the rules in *The CERT C Coding Standard* and *The CERT C++ Coding Standard*. Each rule in the standard that can be analyzed statically has one or more code checkers as part of the Rosecheckers project. The source for the Rosecheckers project is freely downloadable at

<http://rosecheckers.sourceforge.net>

The website also provides a virtual machine containing a complete build of the Rosecheckers project on Linux.

The Rosecheckers project leverages the Compass/ROSE project developed at Lawrence Livermore National Laboratory. This project provides a high-level API for accessing the abstract syntax tree (AST) of a C or C++ source code file. More information on Compass/ROSE is available at

<http://rosecompiler.org>

6.3.4 Other Tools

Most compilers provide warnings for questionable code. Consequently, a compiler can serve as a simple SA tool, although compilers provide significantly fewer diagnostics than dedicated tools. Furthermore, several SA tools require the software to be compiled in order to function. Coverity, for instance, operates by monitoring the build as it progresses, and running its analysis on each file as it is compiled. Consequently, a program that cannot be completely built cannot be completely analyzed by Coverity.

Because of this liability, compilation of the software is a crucial first step, and CERT harvests any diagnostics produced by the compiler and performs the same analysis on them as for other SA tools.

Conclusion

The code in PolyORB-HI-C is shown to be of high quality. However, even high quality code can contain important secure coding violations that need to be addressed, as shown in the findings. AADL code generators and their runtime systems should receive a secure coding analysis if they will be used in security-sensitive applications

Acknowledgments

Thanks to David Svoboda for providing background on SCALE and the tools within it.

References

[Gui 2008]

Shenglin Gui, Liang Ma, Lei Luo, Limeng Yin, Yun Li. “UCaG: An Automatic C Code Generator for AADL Based Upon DeltaOS,” *Proceedings, International Conference on Advanced Computer Theory and Engineering*. IEEE, 2008.

[IEC 60812 2006]

International Electrotechnical Commission (IEC). *Analysis techniques for system reliability – Procedure for failure mode and effects analysis (FMEA)*, 2nd ed. (IEC 60812:2006(E)). IEC, 2006.

[ISAE 2015]

Institute for Space and Aeronautics Engineering. Ocarina AADL model processor. ISAE, 2015. <http://www.openaadl.org/ocarina.html>

[ISO 9899 2011]

International Organization for Standardization/International Electrotechnical Commission (ISO/IEC). *Information technology – Programming languages – C (ISO/IEC 9899:2011)*. ISO/IEC, 2011.

[ISO 9945 2009]

International Organization for Standardization/International Electrotechnical Commission/Institute for Electrical and Electronics Engineers (ISO/IEC/IEEE). *Information technology – Portable Operating System Interface (POSIX) Base Specifications, Issue 7 (ISO/IEC/IEEE 9945:2009)*. ISO/IEC/IEEE, 2009.

[SAE 2012]

SAE International. *Architecture Analysis & Design Language (AADL)*, 3rd ed. (SAE AS 5506B). SAE, 2012.

[Seacord 2014]

Robert Seacord. *The CERT C Coding Standard*, 2nd ed. Addison-Wesley Professional, 2014. <https://www.securecoding.cert.org/confluence/x/HQE>

[Svoboda 2015]

David Svoboda. *SCALE Analysis of JasPer Codebase*. Software Engineering Institute, 2015.

[TPT 2015]

Telecom ParisTech. Refinement of AADL Models for Synthesis of Embedded Systems (RAMSES). TPT, 2015. <http://penelope.enst.fr/aadl/wiki/Projects>

Appendix: Complete List of Secure Coding Violations

Verdict	Path	Line	Message	Rule
True	src/drivers/po_hi_driver_sockets.c	610	The function <code>__po_hi_driver_sockets_init()</code> in <code>po_hi_driver_sockets.c</code> might be able to write outside the bounds of allocated memory on line 610, which could corrupt data, cause the program to crash, or lead to the execution of malicious code.	STR31-C
True	src/po_hi_transport.c	246	The function <code>__po_hi_transport_set_sending_func()</code> in <code>po_hi_transport.c</code> writes one location past the bounds of <code>__po_hi_transport_devices_sending_funcs</code> on line 246, which could corrupt data, cause the program to crash, or lead to the execution of malicious code.	STR31-C
True	src/po_hi_transport.c	247	warning: control reaches end of non-void function [-Wreturn-type]	EXP33-C
True	src/po_hi_main.c	88	warning: function declaration isn't a prototype [-Wstrict-prototypes]	DCL20-C
True	src/po_hi_main.c	95	warning: function declaration isn't a prototype [-Wstrict-prototypes]	DCL20-C
True	src/po_hi_main.c	221	warning: function declaration isn't a prototype [-Wstrict-prototypes]	DCL20-C
True	src/po_hi_main.c	290	warning: function declaration isn't a prototype [-Wstrict-prototypes]	DCL20-C
True	examples/aadlv1/d3.1.3-1/toy.c	34	warning: function declaration isn't a prototype [-Wstrict-prototypes]	DCL20-C
True	examples/aadlv1/d3.1.3-1/toy.c	41	warning: function declaration isn't a prototype [-Wstrict-prototypes]	DCL20-C

True	exam- ples/aadv1/d 3.1.3-1/toy.c	48	warning: function declaration isn't a prototype [- Wstrict-prototypes]	DCL20-C
True	exam- ples/aadv1/d 3.1.3-1/toy.c	62	warning: function declaration isn't a prototype [- Wstrict-prototypes]	DCL20-C
True	exam- ples/aadv2/d 3.1.3-1/toy.c	34	warning: function declaration isn't a prototype [- Wstrict-prototypes]	DCL20-C
True	exam- ples/aadv2/d 3.1.3-1/toy.c	41	warning: function declaration isn't a prototype [- Wstrict-prototypes]	DCL20-C
True	exam- ples/aadv2/d 3.1.3-1/toy.c	48	warning: function declaration isn't a prototype [- Wstrict-prototypes]	DCL20-C
True	exam- ples/aadv2/d 3.1.3-1/toy.c	62	warning: function declaration isn't a prototype [- Wstrict-prototypes]	DCL20-C
True	exam- ples/aadv2/fi le- store/pinger.c	11	warning: function declaration isn't a prototype [- Wstrict-prototypes]	DCL20-C
True	exam- ples/aadv2/fi le- store/pinger.c	34	warning: function declaration isn't a prototype [- Wstrict-prototypes]	DCL20-C
True	exam- ples/aadv2/fi le- store/pinger.c	58	warning: function declaration isn't a prototype [- Wstrict-prototypes]	DCL20-C

True	examples/aadv2/packet-store/pinger.c	10	warning: function declaration isn't a prototype [-Wstrict-prototypes]	DCL20-C
True	examples/aadv2/packet-store/pinger.c	21	warning: function declaration isn't a prototype [-Wstrict-prototypes]	DCL20-C
True	examples/aadv2/packet-store/pinger.c	44	warning: function declaration isn't a prototype [-Wstrict-prototypes]	DCL20-C
True	src/po_hi_storage.c	115	Calling function "stat" to perform check on "file->filename".	FIO01-C
True	src/po_hi_storage.c	265	Calling function "stat" to perform check on "oldfile->filename".	FIO01-C
True	src/po_hi_storage.c	448	Calling function "stat" to perform check on "olddir->dirname".	FIO01-C
True	src/po_hi_transport.c	236	The function __po_hi_transport_get_sending_func() in po_hi_transport.c reads data from just outside the bounds of __po_hi_transport_devices_sending_funcs on line 236.	ARR30-C
True	src/po_hi_transport.c	370	The function __po_hi_get_device_configuration() in po_hi_transport.c reads data from just outside the bounds of __po_hi_devices_configuration_values on line 370.	ARR30-C
True	src/po_hi_task.c	492	Guarantee that array indices are within the valid range	ARR30-C

True	src/po_hi_transport.c	246	Guarantee that array indices are within the valid range	ARR30-C
True	src/drivers/po_hi_driver_sockets.c	320	Use rsize_t or size_t for all integer values representing the size of an object	INT01-C
True	src/drivers/po_hi_driver_sockets.c	646	Use rsize_t or size_t for all integer values representing the size of an object	INT01-C
True	src/po_hi_marshallers.c	77	Use rsize_t or size_t for all integer values representing the size of an object	INT01-C
True	src/po_hi_marshallers.c	84	Use rsize_t or size_t for all integer values representing the size of an object	INT01-C
True	src/po_hi_marshallers.c	85	Use rsize_t or size_t for all integer values representing the size of an object	INT01-C
True	src/po_hi_marshallers.c	94	Use rsize_t or size_t for all integer values representing the size of an object	INT01-C
True	src/po_hi_marshallers.c	99	Use rsize_t or size_t for all integer values representing the size of an object	INT01-C
True	src/po_hi_marshallers.c	100	Use rsize_t or size_t for all integer values representing the size of an object	INT01-C
True	src/po_hi_marshallers.c	115	Use rsize_t or size_t for all integer values representing the size of an object	INT01-C
True	src/po_hi_marshallers.c	122	Use rsize_t or size_t for all integer values representing the size of an object	INT01-C
True	src/po_hi_marshallers.c	129	Use rsize_t or size_t for all integer values representing the size of an object	INT01-C

True	src/po_hi_marshallers.c	143	Use rsize_t or size_t for all integer values representing the size of an object	INT01-C
True	src/po_hi_marshallers.c	150	Use rsize_t or size_t for all integer values representing the size of an object	INT01-C
True	src/po_hi_marshallers.c	157	Use rsize_t or size_t for all integer values representing the size of an object	INT01-C
True	src/po_hi_marshallers.c	162	Use rsize_t or size_t for all integer values representing the size of an object	INT01-C
True	src/po_hi_marshallers.c	163	Use rsize_t or size_t for all integer values representing the size of an object	INT01-C
True	src/po_hi_marshallers.c	167	Use rsize_t or size_t for all integer values representing the size of an object	INT01-C
True	src/po_hi_marshallers.c	168	Use rsize_t or size_t for all integer values representing the size of an object	INT01-C
True	src/po_hi_marshallers.c	172	Use rsize_t or size_t for all integer values representing the size of an object	INT01-C
True	src/po_hi_marshallers.c	173	Use rsize_t or size_t for all integer values representing the size of an object	INT01-C
True	src/po_hi_marshallers.c	177	Use rsize_t or size_t for all integer values representing the size of an object	INT01-C
True	src/po_hi_marshallers.c	178	Use rsize_t or size_t for all integer values representing the size of an object	INT01-C
True	src/po_hi_marshallers.c	182	Use rsize_t or size_t for all integer values representing the size of an object	INT01-C

True	src/po_hi_marshallers.c	183	Use rsize_t or size_t for all integer values representing the size of an object	INT01-C
True	src/po_hi_marshallers.c	188	Use rsize_t or size_t for all integer values representing the size of an object	INT01-C
True	src/po_hi_marshallers.c	189	Use rsize_t or size_t for all integer values representing the size of an object	INT01-C
True	src/po_hi_marshallers.c	194	Use rsize_t or size_t for all integer values representing the size of an object	INT01-C
True	src/po_hi_marshallers.c	195	Use rsize_t or size_t for all integer values representing the size of an object	INT01-C
True	src/po_hi_marshallers.c	199	Use rsize_t or size_t for all integer values representing the size of an object	INT01-C
True	src/po_hi_marshallers.c	200	Use rsize_t or size_t for all integer values representing the size of an object	INT01-C
True	src/po_hi_marshallers.c	204	Use rsize_t or size_t for all integer values representing the size of an object	INT01-C
True	src/po_hi_marshallers.c	205	Use rsize_t or size_t for all integer values representing the size of an object	INT01-C
True	src/po_hi_marshallers.c	210	Use rsize_t or size_t for all integer values representing the size of an object	INT01-C
True	src/po_hi_marshallers.c	211	Use rsize_t or size_t for all integer values representing the size of an object	INT01-C
True	src/po_hi_storage.c	628	Use rsize_t or size_t for all integer values representing the size of an object	INT01-C

True	src/po_hi_storage.c	672	Use <code>resize_t</code> or <code>size_t</code> for all integer values representing the size of an object	INT01-C
True	src/po_hi_storage.c	232	The window of time between the call to <code>stat()</code> and <code>unlink()</code> can be exploited to launch a privilege escalation attack.	FIO45-C
True	src/po_hi_storage.c	277	The window of time between the call to <code>stat()</code> and <code>rename()</code> can be exploited to launch a privilege escalation attack.	FIO45-C
True	src/po_hi_storage.c	395	The window of time between the call to <code>stat()</code> and <code>mkdir()</code> can be exploited to launch a privilege escalation attack.	FIO45-C
True	src/po_hi_storage.c	460	The window of time between the call to <code>stat()</code> and <code>rename()</code> can be exploited to launch a privilege escalation attack.	FIO45-C
True	include/po_hi_storage.h	152	note: expected "char *" but argument is of type "struct __po_hi_request_t"	EXP37-C
True	include/po_hi_storage.h	168	note: expected "char *" but argument is of type "struct __po_hi_request_t"	EXP37-C

True	include/po_hi_storage.h	523	note: expected “__po_hi_storage_packet_t*” but argument is of type “struct __po_hi_request_t*”	EXP37-C
True	include/po_hi_storage.h	546	note: expected “__po_hi_storage_packet_t*” but argument is of type “struct __po_hi_request_t*”	EXP37-C
True	src/po_hi_marshallers.c	84	warning: implicit declaration of function “__po_hi_msg_swap_value” [-Wimplicit-function-declaration]	DCL31-C
True	examples/aadlv2/file-store/pinger.c	79	warning: implicit declaration of function “__po_hi_gqueue_store_out” [-Wimplicit-function-declaration]	DCL31-C
True	examples/aadlv2/packet-store/pinger.c	63	warning: implicit declaration of function “__po_hi_gqueue_store_out” [-Wimplicit-function-declaration]	DCL31-C
True	examples/aadlv2/file-store/pinger.c	46	warning: passing argument 2 of “__po_hi_storage_file_write” from incompatible pointer type [enabled by default]	EXP39-C
True	examples/aadlv2/file-store/pinger.c	64	warning: passing argument 2 of “__po_hi_storage_file_read” from incompatible pointer type [enabled by default]	EXP39-C
True	examples/aadlv2/packet-store/pinger.c	33	warning: passing argument 2 of “__po_hi_storage_packet_store_write” from incompatible pointer type [enabled by default]	EXP39-C

True	examples/aadv2/packet-store/pinger.c	49	warning: passing argument 2 of “__po_hi_storage_packet_store_read” from incompatible pointer type [enabled by default]	EXP39-C
True	src/po_hi_gqueue.c	389	Do not reuse variable names in subscopes: error	DCL01-C
True	src/po_hi_messages.c	71	Do not reuse variable names in subscopes: index	DCL01-C
True	src/po_hi_task.c	556	Do not reuse variable names in subscopes: time	DCL01-C
True	src/po_hi_task.c	597	Do not reuse variable names in subscopes: time	DCL01-C
True	src/po_hi_time.c	154	Do not reuse variable names in subscopes: time	DCL01-C
True	src/po_hi_time.c	161	Do not reuse variable names in subscopes: time	DCL01-C
True	src/po_hi_time.c	168	Do not reuse variable names in subscopes: time	DCL01-C
True	src/po_hi_time.c	175	Do not reuse variable names in subscopes: time	DCL01-C
True	src/po_hi_storage.c	73	Comparing an array to null is not useful: "file->filename == NULL".	MSC12-C
True	src/po_hi_storage.c	108	Comparing an array to null is not useful: "file->filename == NULL".	MSC12-C
True	src/po_hi_storage.c	149	Comparing an array to null is not useful: "file->filename == NULL".	MSC12-C

True	src/po_hi_storage.c	178	Comparing an array to null is not useful: "file->filename == NULL".	MSC12-C
True	src/po_hi_storage.c	210	Comparing an array to null is not useful: "file->filename == NULL".	MSC12-C
True	src/po_hi_storage.c	248	Comparing an array to null is not useful: "oldfile->filename == NULL".	MSC12-C
True	src/po_hi_storage.c	253	Comparing an array to null is not useful: "newfile->filename == NULL".	MSC12-C
True	src/po_hi_storage.c	293	Comparing an array to null is not useful: "file->filename == NULL".	MSC12-C
True	src/po_hi_storage.c	431	Comparing an array to null is not useful: "olddir->dirname == NULL".	MSC12-C
True	src/po_hi_storage.c	436	Comparing an array to null is not useful: "newdir->dirname == NULL".	MSC12-C
True	src/po_hi_storage.c	481	Comparing an array to null is not useful: "dir->dirname == NULL".	MSC12-C
True	src/po_hi_storage.c	540	Comparing an array to null is not useful: "new_current_directory->dirname == NULL".	MSC12-C
True	src/po_hi_storage.c	556	Comparing an array to null is not useful: "current_directory->dirname == NULL".	MSC12-C
True	src/drivers/po_hi_driver_sockets.c	251	warning: unused parameter "dev_id_addr" [-Wunused-parameter]	MSC13-C
True	src/po_hi_transport.c	206	Const-qualify immutable objects: device	DCL00-C

True	examples/aadlv1/flight-mgmt/flight_mgmt_rs/mgmt/flight-mgmt.c	64	Const-qualify immutable objects: cr_v	DCL00-C
True	examples/aadlv1/flight-mgmt/flight_mgmt_rs/mgmt/flight-mgmt.c	65	Const-qualify immutable objects: aoa_v	DCL00-C
True	examples/aadlv2/flight-mgmt/flight_mgmt_rs/mgmt/flight-mgmt.c	64	Const-qualify immutable objects: cr_v	DCL00-C
True	examples/aadlv2/flight-mgmt/flight_mgmt_rs/mgmt/flight-mgmt.c	65	Const-qualify immutable objects: aoa_v	DCL00-C
Suspicious	src/drivers/po_hi_driver_sockets.c	337	The function <code>__po_hi_sockets_poller()</code> in <code>po_hi_driver_sockets.c</code> uses the variable <code>dev_init</code> before it has been initialized.	EXP33-C
Suspicious	src/drivers/po_hi_driver_sockets.c	337	The function <code>__po_hi_sockets_poller()</code> in <code>po_hi_driver_sockets.c</code> uses the variable <code>sock</code> before it has been initialized.	EXP33-C

Suspicious	src/po_hi_storage.c	217	Calling function "stat" to perform check on "file->filename".	FIO01-C
Suspicious	src/po_hi_storage.c	389	Calling function "stat" to perform check on "dir->dirname".	FIO01-C
Suspicious	examples/aadlv1/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	144	Overrunning callee's array of size 4 by passing argument "stall_monitor_global_stall_warn" in call to "__po_hi_transport_send".	ARR30-C
Suspicious	examples/aadlv1/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	180	Overrunning callee's array of size 2 by passing argument "mgmt_hci_k" in call to "__po_hi_gqueue_store_in".	ARR30-C
Suspicious	examples/aadlv1/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	186	Overrunning callee's array of size 2 by passing argument "mgmt_hci_k" in call to "__po_hi_gqueue_store_in".	ARR30-C
Suspicious	examples/aadlv1/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	192	Overrunning callee's array of size 2 by passing argument "mgmt_hci_k" in call to "__po_hi_gqueue_store_in".	ARR30-C
Suspicious	examples/aadlv1/flight-mgmt/flight_	198	Overrunning callee's array of size 2 by passing argument "mgmt_hci_k" in call to "__po_hi_gqueue_store_in".	ARR30-C

	mgmt_rs/mgmt/activity.c			
Suspicious	examples/aadv1/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	234	Overrunning callee's array of size 2 by passing argument "mgmt_hci_k" in call to "__po_hi_gqueue_init".	ARR30-C
Suspicious	examples/aadv1/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	240	Overrunning callee's array of size 2 by passing argument "mgmt_hci_k" in call to "__po_hi_compute_next_period".	ARR30-C
Suspicious	examples/aadv1/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	243	Overrunning callee's array of size 2 by passing argument "mgmt_hci_k" in call to "__po_hi_gqueue_wait_for_incoming_event".	ARR30-C
Suspicious	examples/aadv1/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	244	Overrunning callee's array of size 2 by passing argument "mgmt_hci_k" in call to "__po_hi_compute_next_period".	ARR30-C
Suspicious	examples/aadv1/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	250	Overrunning callee's array of size 2 by passing argument "mgmt_hci_k" in call to "__po_hi_gqueue_get_count".	ARR30-C

Suspicious	examples/aadlv1/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	252	Overrunning callee's array of size 2 by passing argument "mgmt_hci_k" in call to "__po_hi_gqueue_get_value".	ARR30-C
Suspicious	examples/aadlv1/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	253	Overrunning callee's array of size 2 by passing argument "mgmt_hci_k" in call to "__po_hi_gqueue_next_value".	ARR30-C
Suspicious	examples/aadlv1/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	263	Overrunning callee's array of size 2 by passing argument "mgmt_hci_k" in call to "__po_hi_gqueue_next_value".	ARR30-C
Suspicious	examples/aadlv1/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	271	Overrunning callee's array of size 2 by passing argument "mgmt_hci_k" in call to "__po_hi_gqueue_next_value".	ARR30-C
Suspicious	examples/aadlv1/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	273	Overrunning callee's array of size 2 by passing argument "mgmt_hci_k" in call to "on_gear_cmd".	ARR30-C
Suspicious	examples/aadlv1/flight-mgmt/flight_	279	Overrunning callee's array of size 2 by passing argument "mgmt_hci_k" in call to "__po_hi_gqueue_next_value".	ARR30-C

	mgmt_rs/mgmt/activity.c			
Suspicious	examples/aadv1/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	291	Overrunning callee's array of size 4 by passing argument "hci_global_gear_req" in call to "__po_hi_transport_send".	ARR30-C
Suspicious	examples/aadv1/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	291	Overrunning callee's array of size 2 by passing argument "mgmt_hci_k" in call to "__po_hi_transport_send".	ARR30-C
Suspicious	examples/aadv1/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	293	Overrunning callee's array of size 2 by passing argument "mgmt_hci_k" in call to "__po_hi_wait_for_next_period".	ARR30-C
Suspicious	examples/aadv1/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	328	Overrunning callee's array of size 2 by passing argument "mgmt_landing_gear_k" in call to "__po_hi_gqueue_store_in".	ARR30-C
Suspicious	examples/aadv1/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	334	Overrunning callee's array of size 2 by passing argument "mgmt_landing_gear_k" in call to "__po_hi_gqueue_store_in".	ARR30-C

Suspicious	examples/aadlv1/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	364	Overrunning callee's array of size 2 by passing argument "mgmt_landing_gear_k" in call to "__po_hi_gqueue_init".	ARR30-C
Suspicious	examples/aadlv1/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	370	Overrunning callee's array of size 2 by passing argument "mgmt_landing_gear_k" in call to "__po_hi_compute_next_period".	ARR30-C
Suspicious	examples/aadlv1/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	373	Overrunning callee's array of size 2 by passing argument "mgmt_landing_gear_k" in call to "__po_hi_gqueue_wait_for_incoming_event".	ARR30-C
Suspicious	examples/aadlv1/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	374	Overrunning callee's array of size 2 by passing argument "mgmt_landing_gear_k" in call to "__po_hi_compute_next_period".	ARR30-C
Suspicious	examples/aadlv1/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	380	Overrunning callee's array of size 2 by passing argument "mgmt_landing_gear_k" in call to "__po_hi_gqueue_next_value".	ARR30-C
Suspicious	examples/aadlv1/flight-mgmt/flight_	382	Overrunning callee's array of size 2 by passing argument "mgmt_landing_gear_k" in call to "on_req".	ARR30-C

	mgmt_rs/mgmt/activity.c			
Suspicious	examples/aadv1/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	388	Overrunning callee's array of size 2 by passing argument "mgmt_landing_gear_k" in call to "__po_hi_gqueue_next_value".	ARR30-C
Suspicious	examples/aadv1/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	390	Overrunning callee's array of size 2 by passing argument "mgmt_landing_gear_k" in call to "on_dummy_in".	ARR30-C
Suspicious	examples/aadv1/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	400	Overrunning callee's array of size 4 by passing argument "landing_gear_global_ack" in call to "__po_hi_transport_send".	ARR30-C
Suspicious	examples/aadv1/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	400	Overrunning callee's array of size 2 by passing argument "mgmt_landing_gear_k" in call to "__po_hi_transport_send".	ARR30-C
Suspicious	examples/aadv1/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	402	Overrunning callee's array of size 4 by passing argument "landing_gear_global_dummy_out" in call to "__po_hi_transport_send".	ARR30-C

Suspicious	examples/aadlv1/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	402	Overrunning callee's array of size 2 by passing argument "mgmt_landing_gear_k" in call to "__po_hi_transport_send".	ARR30-C
Suspicious	examples/aadlv1/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	404	Overrunning callee's array of size 2 by passing argument "mgmt_landing_gear_k" in call to "__po_hi_wait_for_next_period".	ARR30-C
Suspicious	examples/aadlv1/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	437	Overrunning callee's array of size 2 by passing argument "mgmt_operator_k" in call to "__po_hi_gqueue_init".	ARR30-C
Suspicious	examples/aadlv1/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	443	Overrunning callee's array of size 2 by passing argument "mgmt_operator_k" in call to "__po_hi_compute_next_period".	ARR30-C
Suspicious	examples/aadlv1/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	447	Overrunning callee's array of size 2 by passing argument "mgmt_operator_k" in call to "on_operator".	ARR30-C
Suspicious	examples/aadlv1/flight-mgmt/flight_	449	Overrunning callee's array of size 4 by passing argument "operator_global_gear_cmd" in call to "__po_hi_transport_send".	ARR30-C

	mgmt_rs/mgmt/activity.c			
Suspicious	examples/aadv1/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	449	Overrunning callee's array of size 2 by passing argument "mgmt_operator_k" in call to "__po_hi_transport_send".	ARR30-C
Suspicious	examples/aadv1/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	451	Overrunning callee's array of size 2 by passing argument "mgmt_operator_k" in call to "__po_hi_wait_for_next_period".	ARR30-C
Suspicious	examples/aadv2/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	144	Overrunning callee's array of size 4 by passing argument "stall_monitor_global_stall_warn" in call to "__po_hi_transport_send".	ARR30-C
Suspicious	examples/aadv2/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	180	Overrunning callee's array of size 2 by passing argument "mgmt_hci_k" in call to "__po_hi_gqueue_store_in".	ARR30-C
Suspicious	examples/aadv2/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	186	Overrunning callee's array of size 2 by passing argument "mgmt_hci_k" in call to "__po_hi_gqueue_store_in".	ARR30-C

Suspicious	examples/aadv2/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	192	Overrunning callee's array of size 2 by passing argument "mgmt_hci_k" in call to "__po_hi_gqueue_store_in".	ARR30-C
Suspicious	examples/aadv2/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	198	Overrunning callee's array of size 2 by passing argument "mgmt_hci_k" in call to "__po_hi_gqueue_store_in".	ARR30-C
Suspicious	examples/aadv2/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	234	Overrunning callee's array of size 2 by passing argument "mgmt_hci_k" in call to "__po_hi_gqueue_init".	ARR30-C
Suspicious	examples/aadv2/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	240	Overrunning callee's array of size 2 by passing argument "mgmt_hci_k" in call to "__po_hi_compute_next_period".	ARR30-C
Suspicious	examples/aadv2/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	243	Overrunning callee's array of size 2 by passing argument "mgmt_hci_k" in call to "__po_hi_gqueue_wait_for_incoming_event".	ARR30-C
Suspicious	examples/aadv2/flight-mgmt/flight_	244	Overrunning callee's array of size 2 by passing argument "mgmt_hci_k" in call to "__po_hi_compute_next_period".	ARR30-C

	mgmt_rs/mgmt/activity.c			
Suspicious	examples/aadv2/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	250	Overrunning callee's array of size 2 by passing argument "mgmt_hci_k" in call to "__po_hi_gqueue_get_count".	ARR30-C
Suspicious	examples/aadv2/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	252	Overrunning callee's array of size 2 by passing argument "mgmt_hci_k" in call to "__po_hi_gqueue_get_value".	ARR30-C
Suspicious	examples/aadv2/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	253	Overrunning callee's array of size 2 by passing argument "mgmt_hci_k" in call to "__po_hi_gqueue_next_value".	ARR30-C
Suspicious	examples/aadv2/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	263	Overrunning callee's array of size 2 by passing argument "mgmt_hci_k" in call to "__po_hi_gqueue_next_value".	ARR30-C
Suspicious	examples/aadv2/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	271	Overrunning callee's array of size 2 by passing argument "mgmt_hci_k" in call to "__po_hi_gqueue_next_value".	ARR30-C
Suspicious	examples/aadv2/flight-	273	Overrunning callee's array of size 2 by passing argument "mgmt_hci_k" in call to "on_gear_cmd".	ARR30-C

	mgmt/flight_mgmt_rs/mgmt/activity.c			
Suspicious	examples/aadlv2/flight_mgmt/flight_mgmt_rs/mgmt/activity.c	279	Overrunning callee's array of size 2 by passing argument "mgmt_hci_k" in call to "__po_hi_gqueue_next_value".	ARR30-C
Suspicious	examples/aadlv2/flight_mgmt/flight_mgmt_rs/mgmt/activity.c	291	Overrunning callee's array of size 4 by passing argument "hci_global_gear_req" in call to "__po_hi_transport_send".	ARR30-C
Suspicious	examples/aadlv2/flight_mgmt/flight_mgmt_rs/mgmt/activity.c	291	Overrunning callee's array of size 2 by passing argument "mgmt_hci_k" in call to "__po_hi_transport_send".	ARR30-C
Suspicious	examples/aadlv2/flight_mgmt/flight_mgmt_rs/mgmt/activity.c	293	Overrunning callee's array of size 2 by passing argument "mgmt_hci_k" in call to "__po_hi_wait_for_next_period".	ARR30-C
Suspicious	examples/aadlv2/flight_mgmt/flight_mgmt_rs/mgmt/activity.c	328	Overrunning callee's array of size 2 by passing argument "mgmt_landing_gear_k" in call to "__po_hi_gqueue_store_in".	ARR30-C

Suspicious	examples/aadlv2/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	334	Overrunning callee's array of size 2 by passing argument "mgmt_landing_gear_k" in call to "__po_hi_gqueue_store_in".	ARR30-C
Suspicious	examples/aadlv2/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	364	Overrunning callee's array of size 2 by passing argument "mgmt_landing_gear_k" in call to "__po_hi_gqueue_init".	ARR30-C
Suspicious	examples/aadlv2/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	370	Overrunning callee's array of size 2 by passing argument "mgmt_landing_gear_k" in call to "__po_hi_compute_next_period".	ARR30-C
Suspicious	examples/aadlv2/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	373	Overrunning callee's array of size 2 by passing argument "mgmt_landing_gear_k" in call to "__po_hi_gqueue_wait_for_incoming_event".	ARR30-C
Suspicious	examples/aadlv2/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	374	Overrunning callee's array of size 2 by passing argument "mgmt_landing_gear_k" in call to "__po_hi_compute_next_period".	ARR30-C
Suspicious	examples/aadlv2/flight-mgmt/flight_	380	Overrunning callee's array of size 2 by passing argument "mgmt_landing_gear_k" in call to "__po_hi_gqueue_next_value".	ARR30-C

	mgmt_rs/mgmt/activity.c			
Suspicious	examples/aadv2/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	382	Overrunning callee's array of size 2 by passing argument "mgmt_landing_gear_k" in call to "on_req".	ARR30-C
Suspicious	examples/aadv2/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	388	Overrunning callee's array of size 2 by passing argument "mgmt_landing_gear_k" in call to "__po_hi_gqueue_next_value".	ARR30-C
Suspicious	examples/aadv2/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	390	Overrunning callee's array of size 2 by passing argument "mgmt_landing_gear_k" in call to "on_dummy_in".	ARR30-C
Suspicious	examples/aadv2/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	400	Overrunning callee's array of size 4 by passing argument "landing_gear_global_ack" in call to "__po_hi_transport_send".	ARR30-C
Suspicious	examples/aadv2/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	400	Overrunning callee's array of size 2 by passing argument "mgmt_landing_gear_k" in call to "__po_hi_transport_send".	ARR30-C

Suspicious	examples/aadv2/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	402	Overrunning callee's array of size 4 by passing argument "landing_gear_global_dummy_out" in call to "__po_hi_transport_send".	ARR30-C
Suspicious	examples/aadv2/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	402	Overrunning callee's array of size 2 by passing argument "mgmt_landing_gear_k" in call to "__po_hi_transport_send".	ARR30-C
Suspicious	examples/aadv2/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	404	Overrunning callee's array of size 2 by passing argument "mgmt_landing_gear_k" in call to "__po_hi_wait_for_next_period".	ARR30-C
Suspicious	examples/aadv2/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	437	Overrunning callee's array of size 2 by passing argument "mgmt_operator_k" in call to "__po_hi_gqueue_init".	ARR30-C
Suspicious	examples/aadv2/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	443	Overrunning callee's array of size 2 by passing argument "mgmt_operator_k" in call to "__po_hi_compute_next_period".	ARR30-C
Suspicious	examples/aadv2/flight-mgmt/flight_	447	Overrunning callee's array of size 2 by passing argument "mgmt_operator_k" in call to "on_operator".	ARR30-C

	mgmt_rs/mgmt/activity.c			
Suspicious	examples/aadv2/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	449	Overrunning callee's array of size 4 by passing argument "operator_global_gear_cmd" in call to "__po_hi_transport_send".	ARR30-C
Suspicious	examples/aadv2/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	449	Overrunning callee's array of size 2 by passing argument "mgmt_operator_k" in call to "__po_hi_transport_send".	ARR30-C
Suspicious	examples/aadv2/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	451	Overrunning callee's array of size 2 by passing argument "mgmt_operator_k" in call to "__po_hi_wait_for_next_period".	ARR30-C
Suspicious	examples/aadv2/some-types/some_types_impl/node_b/activity.c	204	Overrunning callee's array of size 2 by passing argument "node_b_struct_rcv_thread_k" in call to "__po_hi_gqueue_store_in".	ARR30-C
Suspicious	examples/aadv2/some-types/some_types_impl/node_b/activity.c	231	Overrunning callee's array of size 2 by passing argument "node_b_struct_rcv_thread_k" in call to "__po_hi_gqueue_init".	ARR30-C

Suspicious	examples/aadlv2/some-types/some_types_impl/node_b/activity.c	237	Overrunning callee's array of size 2 by passing argument "node_b_struct_rcv_thread_k" in call to "__po_hi_compute_next_period".	ARR30-C
Suspicious	examples/aadlv2/some-types/some_types_impl/node_b/activity.c	240	Overrunning callee's array of size 2 by passing argument "node_b_struct_rcv_thread_k" in call to "__po_hi_gqueue_wait_for_incoming_event".	ARR30-C
Suspicious	examples/aadlv2/some-types/some_types_impl/node_b/activity.c	241	Overrunning callee's array of size 2 by passing argument "node_b_struct_rcv_thread_k" in call to "__po_hi_compute_next_period".	ARR30-C
Suspicious	examples/aadlv2/some-types/some_types_impl/node_b/activity.c	242	Overrunning callee's array of size 2 by passing argument "node_b_struct_rcv_thread_k" in call to "__po_hi_gqueue_get_count".	ARR30-C
Suspicious	examples/aadlv2/some-types/some_types_impl/node_b/activity.c	244	Overrunning callee's array of size 2 by passing argument "node_b_struct_rcv_thread_k" in call to "__po_hi_gqueue_get_value".	ARR30-C

Suspicious	examples/aadlv2/some-types/some_types_impl/node_b/activity.c	245	Overrunning callee's array of size 2 by passing argument "node_b_struct_rcv_thread_k" in call to "__po_hi_gqueue_next_value".	ARR30-C
Suspicious	examples/aadlv2/some-types/some_types_impl/node_b/activity.c	250	Overrunning callee's array of size 2 by passing argument "node_b_struct_rcv_thread_k" in call to "__po_hi_wait_for_next_period".	ARR30-C
Suspicious	examples/aadlv2/unseeker/sunseekercontrolsysteype_local/sunseeker/activity.c	82	No conditions allow control to exit the loop.	MSC21-C
Suspicious	examples/aadlv2/unseeker/sunseekercontrolsysteype_local/sunseeker/activity.c	169	No conditions allow control to exit the loop.	MSC21-C
Suspicious	examples/aadlv1/d3.1.3-1/toy_exam-	30	No conditions allow control to exit the loop.	MSC21-C

	ple_sample_1/gnc_tm tc_pos/activity.c			
Suspicious	examples/aadlv1/d 3.1.3- 1/toy_example_sample_1/gnc_tm tc_pos/activity.c	57	No conditions allow control to exit the loop.	MSC21-C
Suspicious	examples/aadlv1/flight- mgmt/flight_ mgmt_rs/mgmt/activity.c	53	No conditions allow control to exit the loop.	MSC21-C
Suspicious	examples/aadlv1/flight- mgmt/flight_ mgmt_rs/mgmt/activity.c	139	No conditions allow control to exit the loop.	MSC21-C
Suspicious	examples/aadlv1/flight- mgmt/flight_ mgmt_rs/mgmt/activity.c	241	No conditions allow control to exit the loop.	MSC21-C
Suspicious	examples/aadlv1/flight- mgmt/flight_ mgmt_rs/mgmt/activity.c	371	No conditions allow control to exit the loop.	MSC21-C

	mgmt_rs/mgmt/activity.c			
Suspicious	examples/aadv1/flight_mgmt/flight_mgmt_rs/mgmt/activity.c	444	No conditions allow control to exit the loop.	MSC21-C
Suspicious	examples/aadv1/ping/ping_local/node_a/activity.c	52	No conditions allow control to exit the loop.	MSC21-C
Suspicious	examples/aadv1/ping/ping_local/node_a/activity.c	130	No conditions allow control to exit the loop.	MSC21-C
Suspicious	examples/aadv1/rma/rma_impl/node_a/activity.c	28	No conditions allow control to exit the loop.	MSC21-C
Suspicious	examples/aadv1/rma/rma_impl/node_a/activity.c	55	No conditions allow control to exit the loop.	MSC21-C
Suspicious	examples/aadv1/sun-seeker/sun-seekercontrolsystem_ty	82	No conditions allow control to exit the loop.	MSC21-C

	pe_local/sunseeker/activity.c			
Suspicious	examples/aadlv1/sunseeker/sunseekercontrolsystem_type_local/sunseeker/activity.c	169	No conditions allow control to exit the loop.	MSC21-C
Suspicious	examples/aadlv2/cpp/cpp_test_impl/node_a/activity.c	28	No conditions allow control to exit the loop.	MSC21-C
Suspicious	examples/aadlv2/d3.1.3-1/toy_example_sample_1/gnc_tm_tc_pos/activity.c	30	No conditions allow control to exit the loop.	MSC21-C
Suspicious	examples/aadlv2/d3.1.3-1/toy_example_sample_1/gnc_tm_tc_pos/activity.c	57	No conditions allow control to exit the loop.	MSC21-C
Suspicious	examples/aadlv2/file-	51	No conditions allow control to exit the loop.	MSC21-C

	store/ping_implementation/node_a/activity.c			
Suspicious	examples/aadlv2/file-store/ping_implementation/node_a/activity.c	87	No conditions allow control to exit the loop.	MSC21-C
Suspicious	examples/aadlv2/file-store/ping_implementation/node_b/activity.c	77	No conditions allow control to exit the loop.	MSC21-C
Suspicious	examples/aadlv2/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	53	No conditions allow control to exit the loop.	MSC21-C
Suspicious	examples/aadlv2/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	139	No conditions allow control to exit the loop.	MSC21-C
Suspicious	examples/aadlv2/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	241	No conditions allow control to exit the loop.	MSC21-C

Suspicious	examples/aadlv2/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	371	No conditions allow control to exit the loop.	MSC21-C
Suspicious	examples/aadlv2/flight-mgmt/flight_mgmt_rs/mgmt/activity.c	444	No conditions allow control to exit the loop.	MSC21-C
Suspicious	examples/aadlv2/import/ping_nactive/node_a/activity.c	52	No conditions allow control to exit the loop.	MSC21-C
Suspicious	examples/aadlv2/import/ping_nactive/node_b/activity.c	77	No conditions allow control to exit the loop.	MSC21-C
Suspicious	examples/aadlv2/monitor/ping_impl/node_a/activity.c	54	No conditions allow control to exit the loop.	MSC21-C
Suspicious	examples/aadlv2/monitor/ping_impl/node_b/activity.c	77	No conditions allow control to exit the loop.	MSC21-C

Suspicious	examples/aadlv2/packet-store/ping_impl/node_a/activity.c	51	No conditions allow control to exit the loop.	MSC21-C
Suspicious	examples/aadlv2/packet-store/ping_impl/node_a/activity.c	87	No conditions allow control to exit the loop.	MSC21-C
Suspicious	examples/aadlv2/packet-store/ping_impl/node_b/activity.c	77	No conditions allow control to exit the loop.	MSC21-C
Suspicious	examples/aadlv2/producer-consumer/pc_simple_impl/pr_a/activity.c	52	No conditions allow control to exit the loop.	MSC21-C
Suspicious	examples/aadlv2/producer-consumer/pc_simple_impl/pr_a/activity.c	130	No conditions allow control to exit the loop.	MSC21-C
Suspicious	examples/aadlv2/producer-consumer/pc_sim	78	No conditions allow control to exit the loop.	MSC21-C

	ple_impl/pr_b/activity.c			
Suspicious	examples/aadlv2/producer-consumer/pc_simple_impl/pr_b/activity.c	128	No conditions allow control to exit the loop.	MSC21-C
Suspicious	examples/aadlv2/some-types-stdint/some_types_stdint_impl/node_a/activity.c	55	No conditions allow control to exit the loop.	MSC21-C
Suspicious	examples/aadlv2/some-types-stdint/some_types_stdint_impl/node_b/activity.c	78	No conditions allow control to exit the loop.	MSC21-C
Suspicious	examples/aadlv2/some-types-stdint/some_types_stdint_impl/node_b/activity.c	158	No conditions allow control to exit the loop.	MSC21-C
Suspicious	examples/aadlv2/some-types/some_types_impl/no	58	No conditions allow control to exit the loop.	MSC21-C

	de_a/activ-ity.c			
Suspi-cious	exam-ples/aadlv2/some-types/some_types_impl/node_b/activ-ity.c	78	No conditions allow control to exit the loop.	MSC21-C
Suspi-cious	exam-ples/aadlv2/some-types/some_types_impl/node_b/activ-ity.c	158	No conditions allow control to exit the loop.	MSC21-C
Suspi-cious	exam-ples/aadlv2/some-types/some_types_impl/node_b/activ-ity.c	238	No conditions allow control to exit the loop.	MSC21-C
Suspi-cious	exam-ples/aadlv2/sun-seeker/sun-seekercon-trolsystem_ty-pe_dist/sun-seekercon-troller/activ-ity.c	82	No conditions allow control to exit the loop.	MSC21-C
Suspi-cious	exam-ples/aadlv2/s	82	No conditions allow control to exit the loop.	MSC21-C

	un-seeker/sun-seekercontrolsystem_type_dist/sun-seekerplant/activity.c			
Suspicious	examples/aadv1/sun-seeker/sun-seekercontrolsystem_type_local/sun-seeker/sun-seekercontroller.c	31	Detect and handle floating point errors	FLP03-C
Suspicious	examples/aadv1/sun-seeker/sun-seekercontrolsystem_type_local/sun-seeker/sun-seekercontroller.c	32	Detect and handle floating point errors	FLP03-C
Suspicious	examples/aadv1/sun-seeker/sun-seekercontrolsystem_type_local/sun-seeker/sun-seekercontroller.c	34	Detect and handle floating point errors	FLP03-C

Suspicious	examples/aadv1/sun-seeker/sun-seekercontrolsystem_type_local/sun-seeker/sun-seekercontroller.c	36	Detect and handle floating point errors	FLP03-C
Suspicious	examples/aadv1/sun-seeker/sun-seekercontrolsystem_type_local/sun-seeker/sun-seekercontroller.c	38	Detect and handle floating point errors	FLP03-C
Suspicious	examples/aadv1/sun-seeker/sun-seekercontrolsystem_type_local/sun-seeker/sun-seekerplant.c	19	Detect and handle floating point errors	FLP03-C
Suspicious	examples/aadv1/sun-seeker/sun-seekercontrolsystem_ty	21	Detect and handle floating point errors	FLP03-C

	pe_local/sun-seeker/sun-seekerplant.c			
Suspicious	examples/aadv1/sun-seeker/sun-seekercontrolsystem_typed_local/sun-seeker/sun-seekerplant.c	22	Detect and handle floating point errors	FLP03-C
Suspicious	examples/aadv1/sun-seeker/sun-seekercontrolsystem_typed_local/sun-seeker/sun-seekerplant.c	24	Detect and handle floating point errors	FLP03-C
Suspicious	examples/aadv1/sun-seeker/sun-seekercontrolsystem_typed_local/sun-seeker/sun-seekerplant.c	25	Detect and handle floating point errors	FLP03-C
Suspicious	examples/aadv1/sun-seeker/sun-seekercontrolsystem_typed_local/sun-seeker/sun-seekerplant.c	28	Detect and handle floating point errors	FLP03-C

	pe_local/sun-seeker/sun-seekerplant.c			
Suspicious	examples/aadv1/sun-seeker/sun-seekercontrolsystem_type_local/sun-seeker/sun-seekerplant.c	30	Detect and handle floating point errors	FLP03-C
Suspicious	examples/aadv2/sun-seeker/sun-seekercontrolsystem_type_dist/sun-seekercontroller/sun-seekercontroller.c	31	Detect and handle floating point errors	FLP03-C
Suspicious	examples/aadv2/sun-seeker/sun-seekercontrolsystem_type_dist/sun-seekercontroller/sun-seekercontroller.c	32	Detect and handle floating point errors	FLP03-C
Suspicious	examples/aadv2/s	34	Detect and handle floating point errors	FLP03-C

	un-seeker/sun-seekercontrolsysteem_type_dist/sun-seekercontroller/sun-seekercontroller.c			
Suspicious	examples/aadv2/sun-seeker/sun-seekercontrolsysteem_type_dist/sun-seekercontroller/sun-seekercontroller.c	36	Detect and handle floating point errors	FLP03-C
Suspicious	examples/aadv2/sun-seeker/sun-seekercontrolsysteem_type_dist/sun-seekercontroller/sun-seekercontroller.c	38	Detect and handle floating point errors	FLP03-C
Suspicious	examples/aadv2/sun-seeker/sun-seekercontrolsysteem_type	19	Detect and handle floating point errors	FLP03-C

	pe_dist/sunseekerplant/sunseekerplant.c			
Suspicious	examples/aadv2/sunseeker/sunseekercontrolsystem_type_dist/sunseekerplant/sunseekerplant.c	21	Detect and handle floating point errors	FLP03-C
Suspicious	examples/aadv2/sunseeker/sunseekercontrolsystem_type_dist/sunseekerplant/sunseekerplant.c	22	Detect and handle floating point errors	FLP03-C
Suspicious	examples/aadv2/sunseeker/sunseekercontrolsystem_type_dist/sunseekerplant/sunseekerplant.c	24	Detect and handle floating point errors	FLP03-C
Suspicious	examples/aadv2/s	25	Detect and handle floating point errors	FLP03-C

	un-seeker/sun-seekercontrolsystem_type_dist/sun-seekerplant/sun-seekerplant.c			
Suspicious	examples/aadlv2/sun-seeker/sun-seekercontrolsystem_type_dist/sun-seekerplant/sun-seekerplant.c	28	Detect and handle floating point errors	FLP03-C
Suspicious	examples/aadlv2/sun-seeker/sun-seekercontrolsystem_type_dist/sun-seekerplant/sun-seekerplant.c	30	Detect and handle floating point errors	FLP03-C
Suspicious	examples/aadlv2/sun-seeker/sun-seekercontrolsystem_type_local/sun-seeker/sun-seekercontroller.c	31	Detect and handle floating point errors	FLP03-C

Suspicious	examples/aadv2/sunseeker/sunseekercontrolsystem_type_local/sunseeker/sunseekercontroller.c	32	Detect and handle floating point errors	FLP03-C
Suspicious	examples/aadv2/sunseeker/sunseekercontrolsystem_type_local/sunseeker/sunseekercontroller.c	34	Detect and handle floating point errors	FLP03-C
Suspicious	examples/aadv2/sunseeker/sunseekercontrolsystem_type_local/sunseeker/sunseekercontroller.c	36	Detect and handle floating point errors	FLP03-C
Suspicious	examples/aadv2/sunseeker/sunseekercontrolsystem_type_local/sun-	38	Detect and handle floating point errors	FLP03-C

	seeker/sun-seekercontroller.c			
Suspicious	examples/aadlv2/sun-seeker/sun-seekercontrolsystem_type_local/sun-seeker/sun-seekerplant.c	19	Detect and handle floating point errors	FLP03-C
Suspicious	examples/aadlv2/sun-seeker/sun-seekercontrolsystem_type_local/sun-seeker/sun-seekerplant.c	21	Detect and handle floating point errors	FLP03-C
Suspicious	examples/aadlv2/sun-seeker/sun-seekercontrolsystem_type_local/sun-seeker/sun-seekerplant.c	22	Detect and handle floating point errors	FLP03-C
Suspicious	examples/aadlv2/sun-seeker/sun-seekercontrolsystem_type	24	Detect and handle floating point errors	FLP03-C

	pe_local/sun-seeker/sun-seekerplant.c			
Suspicious	examples/aadlv2/sun-seeker/sun-seekercontrolsystem_typed_pe_local/sun-seeker/sun-seekerplant.c	25	Detect and handle floating point errors	FLP03-C
Suspicious	examples/aadlv2/sun-seeker/sun-seekercontrolsystem_typed_pe_local/sun-seeker/sun-seekerplant.c	28	Detect and handle floating point errors	FLP03-C
Suspicious	examples/aadlv2/sun-seeker/sun-seekercontrolsystem_typed_pe_local/sun-seeker/sun-seekerplant.c	30	Detect and handle floating point errors	FLP03-C
Suspicious	src/po_hi_gqueue.c	132	The function <code>__po_hi_gqueue_init()</code> in <code>po_hi_gqueue.c</code> never uses the value it assigns to the variable <code>err</code> on line 132.	MSC13-C

Suspicious	src/po_hi_gqueue.c	134	The function <code>__po_hi_gqueue_init()</code> in <code>po_hi_gqueue.c</code> never uses the value it assigns to the variable <code>err</code> on line 134.	MSC13-C
Suspicious	src/po_hi_gqueue.c	140	The function <code>__po_hi_gqueue_init()</code> in <code>po_hi_gqueue.c</code> never uses the value it assigns to the variable <code>err</code> on line 140.	MSC13-C
Suspicious	src/po_hi_gqueue.c	144	The function <code>__po_hi_gqueue_init()</code> in <code>po_hi_gqueue.c</code> never uses the value it assigns to the variable <code>err</code> on line 144.	MSC13-C
Suspicious	src/po_hi_gqueue.c	146	The function <code>__po_hi_gqueue_init()</code> in <code>po_hi_gqueue.c</code> never uses the value it assigns to the variable <code>err</code> on line 146.	MSC13-C
Suspicious	src/po_hi_gqueue.c	333	The function <code>__po_hi_gqueue_store_in()</code> in <code>po_hi_gqueue.c</code> never uses the initial value it assigns to the variable <code>err</code> on line 333.	MSC13-C
Suspicious	src/po_hi_gqueue.c	369	The function <code>__po_hi_gqueue_wait_for_incoming_event()</code> in <code>po_hi_gqueue.c</code> never uses the initial value it assigns to the variable <code>error</code> on line 369.	MSC13-C
Suspicious	src/po_hi_gqueue.c	389	The function <code>__po_hi_gqueue_wait_for_incoming_event()</code> in <code>po_hi_gqueue.c</code> never uses the initial value it assigns to the variable <code>error</code> on line 389.	MSC13-C
Suspicious	src/po_hi_storage.c	324	warning: unused parameter "oldfile" [-Wunused-parameter]	MSC13-C
Suspicious	src/po_hi_storage.c	324	warning: unused parameter "newfile" [-Wunused-parameter]	MSC13-C
Suspicious	src/po_hi_storage.c	329	warning: unused parameter "file" [-Wunused-parameter]	MSC13-C

Suspicious	src/po_hi_storage.c	334	warning: unused parameter "file" [-Wunused-parameter]	MSC13-C
Suspicious	src/po_hi_storage.c	528	warning: unused parameter "dir" [-Wunused-parameter]	MSC13-C
Suspicious	src/po_hi_storage.c	533	warning: unused parameter "dir" [-Wunused-parameter]	MSC13-C
Suspicious	src/po_hi_storage.c	591	warning: unused parameter "store" [-Wunused-parameter]	MSC13-C
Suspicious	src/po_hi_storage.c	591	warning: unused parameter "file" [-Wunused-parameter]	MSC13-C
Suspicious	src/po_hi_storage.c	596	warning: unused parameter "store" [-Wunused-parameter]	MSC13-C
Suspicious	src/po_hi_storage.c	596	warning: unused parameter "file" [-Wunused-parameter]	MSC13-C
Suspicious	src/po_hi_task.c	556	warning: unused parameter "time" [-Wunused-parameter]	MSC13-C
Suspicious	src/po_hi_transport.c	378	warning: unused parameter "task_id" [-Wunused-parameter]	MSC13-C
Suspicious	src/po_hi_transport.c	378	warning: unused parameter "port" [-Wunused-parameter]	MSC13-C
Suspicious	examples/aadlv1/d3.1.3-1/toy.c	7	warning: unused parameter "n" [-Wunused-parameter]	MSC13-C
Suspicious	examples/aadlv1/flight-	16	warning: unused parameter "self" [-Wunused-parameter]	MSC13-C

	mgmt/flight-mgmt.c			
Suspicious	examples/aadv1/flight-mgmt/flight-mgmt.c	28	warning: unused parameter "self" [-Wunused-parameter]	MSC13-C
Suspicious	examples/aadv1/flight-mgmt/flight-mgmt.c	40	warning: unused parameter "self" [-Wunused-parameter]	MSC13-C
Suspicious	examples/aadv1/flight-mgmt/flight-mgmt.c	51	warning: unused parameter "self" [-Wunused-parameter]	MSC13-C
Suspicious	examples/aadv2/d3.1.3-1/toy.c	7	warning: unused parameter "n" [-Wunused-parameter]	MSC13-C
Suspicious	examples/aadv2/flight-mgmt/flight-mgmt.c	16	warning: unused parameter "self" [-Wunused-parameter]	MSC13-C
Suspicious	examples/aadv2/flight-mgmt/flight-mgmt.c	28	warning: unused parameter "self" [-Wunused-parameter]	MSC13-C
Suspicious	examples/aadv2/flight-	40	warning: unused parameter "self" [-Wunused-parameter]	MSC13-C

	mgmt/flight- mgmt.c			
Suspicious	exam- ples/aadv2/fl ight- mgmt/flight- mgmt.c	51	warning: unused parameter "self" [-Wunused-parameter]	MSC13-C

Contact Us

Software Engineering Institute
4500 Fifth Avenue, Pittsburgh, PA 15213-2612

Phone: 412/268.5800 | 888.201.4479

Web: www.sei.cmu.edu | www.cert.org

Email: info@sei.cmu.edu

Copyright 2015 Carnegie Mellon University

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the United States Department of Defense.

References herein to any specific commercial product, process, or service by trade name, trade mark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by Carnegie Mellon University or its Software Engineering Institute.

NO WARRANTY. THIS CARNEGIE MELLON UNIVERSITY AND SOFTWARE ENGINEERING INSTITUTE MATERIAL IS FURNISHED ON AN "AS-IS" BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY, OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT.

This material has been approved for public release and unlimited distribution except as restricted below.

Internal use:* Permission to reproduce this material and to prepare derivative works from this material for internal use is granted, provided the copyright and "No Warranty" statements are included with all reproductions and derivative works.

External use:* This material may be reproduced in its entirety, without modification, and freely distributed in written or electronic form without requesting formal permission. Permission is required for any other external and/or commercial use. Requests for permission should be directed to the Software Engineering Institute at permission@sei.cmu.edu.

* These restrictions do not apply to U.S. government entities.

Carnegie Mellon® and CERT® are registered marks of Carnegie Mellon University.

DM-0002798