Business Plan and Convener’s Report

ISO/IEC/JTC 1/SC 22/WG 23 (Programming Language Vulnerabilities)

Document: ISO/IEC JTC 1/SC 22/WG 23/N0873

Date: 2020-06-15

PERIOD COVERED: July 2019 – June 2020

SUBMITTED BY:

Convener, ISO/IEC JTC 1/SC 22/WG 23: Vulnerabilities  *Stephen Michell
Maurya Software*

*156 Shinny Ave,
Stittsville, Ontario K2V 0G4 Canada*

*Office: +1(613)299-9047
E-mail: stephen.michell@maurya.on.ca*

**1. MANAGEMENT SUMMARY**

* 1.1.  JTC 1/SC 22/WG 23  Programming Language Vulnerabilities

1.2.  PROJECT REPORT

1.2.1. COMPLETED PROJECTS

ISO/IEC TR 24772-1:2019, *Programing languages –* *Guidance to Avoiding Vulnerabilities in Programming Languages – Part 1: Language independent guidance*

This is a technical report

ISO/IEC TR 24772-2:2020, *Programing languages –* *Guidance to Avoiding Vulnerabilities in Programming Languages – Part 2: Vulnerability descriptions for programming language Ada*

This is a technical report

ISO/IEC TR 24772-1:2020 *Programing languages –* *Guidance to Avoiding Vulnerabilities in Programming Languages – Part 3: Vulnerability descriptions for programming language C*

This is a technical report

ISO/IEC 17960, *Code Signing for Source Code.* This project is to produce an International Standard, and has been published.

1.2.2. PROJECTS UNDERWAY

ISO/IEC TR 24772-4, *Guidance to Avoiding Vulnerabilities in Programming Languages – Part 4: Vulnerability descriptions for programming language Python.* This is the update of TR24772:2013 for Python vulnerabilities which was Annex E, following the project split of project 22.24772. Under development

ISO/IEC TR 24772-8, *Guidance to Avoiding Vulnerabilities in Programming Languages – Part 8: Vulnerability descriptions for programming language Fortran.* This is the update of TR 24772:2013 Annex C for language specific vulnerabilities for Fortran, following the project split of project 22.24772. Under development.

ISO/IEC TR 24772-10, *Guidance to Avoiding Vulnerabilities in Programming Languages – Part 3, Vulnerability descriptions for programming language C++.* This is a new part for language specific vulnerabilities for C++. Under development.

ISO/IEC TR 24772-11, *Guidance to Avoiding Vulnerabilities in Programming Languages – Part 8: Vulnerability descriptions for programming language Java.* This is a new part for language specific vulnerabilities for Java. Being published.

1.2.3. CANCELLED PROJECTS

none

1.2.4. COOPERATION and COMPETITION

Where appropriate, WG 23 has established active liaisons with other SC22 working groups, other JTC 1 subcommittee working groups (such as SC 27/WG 3 and SC 7 WG19) and other standards organizations, such as Ecma International. See the table in 2.3 for a list of liaisons.

There is no apparent direct competition with any other current SC22 working group or JTC 1 subcommittee.

**2. PERIOD REVIEW**

2.1. MARKET REQUIREMENTS

WG 23 is responding to the needs of the programming language community by inclusion. WG 23 will accept input and liaison by any and all appropriate organizations.

The marketplace demands robust, secure software. Vulnerabilities are the antithesis of robust, secure software. Many of the attacks on software-based systems succeed because the computer language used did not prevent the attack vector, and did not warn the developer that the code being produced contained flaws that could be used to generate attacks.

WG 23 has produced 3 editions of TR 24772 (the last one being TR 24772-1:2019), but there are vulnerabilities that still need to be identified, and programming languages that still need to be documented with regards to vulnerabilities.

2.2. ACHIEVEMENTS

WG 23 has published the first edition of TR 24772-1, -2 and -3 after splitting the original TR 24772 project and the TR into Part 1, language independent part, and Parts 2, 3, 4, 8, 10 and 11 for language-specific vulnerability descriptions for Ada, C, Python, Fortran, C++ and Java.

2.3. RESOURCES

Seven national bodies have participated in the WG 23 meetings this year: Austria, Canada, China, Italy, Korea, UK, and the USA, as well as several liaisons.

Over the last several years WG 23 has made Web conferencing capabilities available for those that are finding it difficult to travel. At a typical WG 23, one-third to one-half of all participates are remote, but still participate meaningfully in the meeting. WG 23 finds that mixed-mode meetings work well in developing technical content. WG 23 would like to thank ISO for the Web conferencing support.

Of course, with the world-wide pandemic, WG 23 is holding all meetings virtually.

Liaison with five SC22 Language groups, and four groups outside of SC22 have been established. Liaisons fill a valuable role in that they identify the vulnerabilities that exist (and do not exist) in their language, produce the primary documentation of those vulnerabilities and turn them into the relevant language-dependent part in conjunction with the core team through the liaison individual.

Current WG 23 liaisons are:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Group** |  | **Name/Type** |  | **Person assigned** |
| SC 22/WG4 |  | Cobol |  | Robert Karlin, Chris Tandy |
| SC 22/WG5 |  | Fortran |  | Dan Nagel |
| SC 22/WG9 |  | Ada |  | Erhard Ploedereder |
| SC 22/ WG14 |  | C |  | Clive Pygott |
| SC 22/ WG 21 |  | C++ |  | Group  |

**3. FOCUS NEXT WORK PERIOD**

3.1.  DELIVERABLES

WG 23 has the following documents published:

 JTC 1 24772-1:2019, *Guidance to Avoiding Vulnerabilities in Programming Languages – Part 1: Language Independent*

JTC 1 24772-2:2020, *Guidance to Avoiding Vulnerabilities in Programming Languages – Part 2, Vulnerability descriptions for programming language Ada.*

JTC 1 24772-3:2020

*Guidance to Avoiding Vulnerabilities in Programming Languages – Part 3, Vulnerability descriptions for programming language C;*

WG 23 is working on the following parts:

JTC 1 24772-4, *Guidance to Avoiding Vulnerabilities in Programming Languages – Part 4: Vulnerability descriptions for programming language Python.*

JTC 1 24772-10, *Guidance to Avoiding Vulnerabilities in Programming Languages – Part 10: Vulnerability descriptions for programming language C++.*

JTC 1 24772-11, *Guidance to Avoiding Vulnerabilities in Programming Languages – Part 11: Vulnerability descriptions for programming language Java.*

3.2.  STRATEGIES

WG 23 decided in 2015 that a core document and seven language-specific annexes, with at least two or three more in planning, creates a maintenance burden that makes it difficult to keep all portions of the document up to date in a single document.

WG 23 therefore decided to split TR 24772 into a series of parts, as follows (see also clause 4.1 for the official request for SC 22 action):

* TR24772-1 *Information Technology — Programming languages — Guidance to avoiding vulnerabilities in programming languages through language selection and use – Part 1: Language Independent View*
* TR24772-2 *Information Technology — Programming languages — Guidance to avoiding vulnerabilities in programming languages through language selection and use – Part 2: Programming Language Ada*
* TR24772-3 *Information Technology — Programming languages — Guidance to avoiding vulnerabilities in programming languages through language selection and use – Part 3: Programming Language C*
* TR24772-4 *Information Technology — Programming languages — Guidance to avoiding vulnerabilities in programming languages through language selection and use – Part 4: Programming Language Python*
* TR24772-5 *Information Technology — Programming languages — Guidance to avoiding vulnerabilities in programming languages through language selection and use – Part 5: Programming Language Ruby*
* TR24772-6 *Information Technology — Programming languages — Guidance to avoiding vulnerabilities in programming languages through language selection and use – Part 6: Programming Language SPARK*
* TR24772-7 *Information Technology — Programming languages — Guidance to avoiding vulnerabilities in programming languages through language selection and use – Part 7: Programming Language PHP*
* TR24772-8 *Information Technology — Programming languages — Guidance to avoiding vulnerabilities in programming languages through language selection and use – Part 8: Programming Language Fortran*
* TR24772-9 *Information Technology — Programming languages — Guidance to avoiding vulnerabilities in programming languages through language selection and use –Part 9: Programming Language COBOL*
* TR24772-10 *Information Technology — Programming languages — Guidance to avoiding vulnerabilities in programming languages through language selection and use – Part 10: Programming Language C++.*
* 24772-11 *Information Technology — Programming languages — Guidance to avoiding vulnerabilities in programming languages through language selection and use – Part 11: Programming Language Java.* This is a new request to SC 22.

3.3.  RISKS

Progress on parts 4, 10 and 11 for which work items are allocated are showing reasonable progress. Some of the other parts for which work items have not been initiated require the identification of resources within other working groups or external experts to undertake the work.

3.4.  OPPORTUNITIES

No special opportunities arose during the past year.

3.5.  WORK PROGRAM PRIORITIES

See 4.1.

**4. OTHER ITEMS**

4.1. POSSIBLE ACTION REQUESTS AT FORTHCOMING 2019 PLENARY

4.1.1 Register the following project with ISO CS

*JTC 1 NP TR 24772-11 Guidance to avoiding vulnerabilities in programming languages through language selection and use – Part 11: Programming language Java. (Project Editor Stephen Michell)*

Initiate the following projects with the editors as noted:

* + JTC 1 NP TR 24772-4, Guidance to avoiding vulnerabilities in programming languages through language selection and use – Programming language Python. (Project Editor Stephen Michell)
	+ JTC 1 NP TR 24772-6, Guidance to avoiding vulnerabilities in programming languages through language selection and use – Part 6: Programming language SPARK.  (Project Editor Stephen Michell)
	+ JTC 1 NP TR 24772-10, Guidance to avoiding vulnerabilities in programming languages through language selection and use – Part 10: Programming language C++.  (Project Editor Stephen Michell)
	+ JTC 1 NP TR 24772-11 Guidance to avoiding vulnerabilities in programming languages through language selection and use – Part 11: Programming language Java. (Project Editor Stephen Michell)

4.2.  ELECTRONIC DOCUMENT DISTRIBUTION

Documents relevant to ISO/IEC/JTC1/SC22 processing are being entered on the ISO eCommittee web site for WG 23. WG 23 conducts some of its detailed technical discussion using the email reflector maintained by Keld Simonsen. WG 23 also has an ftp and Web site at http://open-std.org/jtc1/sc22/wg23.

4.4. RECENT MEETINGS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Date | Place | # attendees | Host |
| 57 | 27-28 Aug 2018 | Zoom | 6 |  |
| 58 | 8-9 Nov 2018 | San Diego CA with WG 21 | 9 | USA, WG 21 |
| 59 | 21 January 2019 | Zoom Meeting | 5 | Convenor |
| 60 | Cancelled |  |  |  |
| 61 | 20-22 Feb 2019 | Kona, Hawaii with WG 21 | 7 | USA, WG 21 |
| 62 | 6 May 2019 | Electronic | 5 |  |
| 63 | 16-18 July 2019 | Cologne Germany with WG 21 |  | Germany, WG 21 |
| 64 |  |  |  |  |
| 65 | 15 October 2019 | Zoom Meeting | 6 |  |
| 66 | 6-9 November 2020 | Belfast, UK with WG 21 | 6 |  |
| 67 | 10-12 Feb 2020 | Prague, Czech Republic with WG 21 |  |  |
| 68 | 23-24 Feb 2020 | Las Vegas NV with INCITS Fortran |  |  |
| 69 |  |  |  |  |

4.5. FUTURE MEETINGS

#70 Zoom Meeting Sep 1, 2 and 3 2000-2200 UTC

#71 USA or Zoom meeting Nov 2020 (with WG 21)

#72 Kona, HA or Zoom 22 Feb 2021

#73 Las Vegas with WG 5 TBD June 2020

#68 TBD April 2020