[2021-11-17] Change Log

Contents

Keynotes

- Added .jsx extension in javascript languages analysis
- KLA (KiuwanLocalAnalyzer) now runs under Java 16
- New Cobol preprocessor script

A new script (cobolPreprocess.xml) for pre-processing COBOL sources and replacing COPY statements with the content of copybooks is provided. This tool is useful when COPY statements are used in a way that makes the common strategy of parsing separately COBOL programs and copybooks lead to a high rate of parse errors but with the cost of losing the original source code lines in reported defects, due to plain code substitution. The script is located in the local analyzer bin directory, and usage is as follows:

```
ant -f cobolPreprocess.xml run
  -Dinput=SOURCES_DIR
  [-Ddialect=cobol85|cobolibm|cobolmicrofocus|coboltandem|acucobol|rmcobol]
  [-Dfreeform=true|false] [-Dencoding=ENCODING]
  [-DmarginType=autodetect|no_margin|left_margin|right_margin|both_margin|
               already_transformed|tandem|tandem_ansi|terminal]
  [-Dinclude=INCLUDE_PATTERNS] [-Dexclude=EXCLUDE_PATTERNS]
  [-DprogramExtensions=extensions] [-DcopyExtensions=extensions]
  [-Doutput=OUTPUT_DIR]
```

where:

- SOURCES_DIR: The input
- dialect: cobol85, cobolibm, cobolmicrofocus, coboltandem, acucobol, rmcobol.
  Default: cobol85.
- marginType: The margin type to use when formatting. The default, autodetect, tries to detect the margin format heuristically. both_margin is the ANSI format.
- freeform: true | false. If true, free-format COBOL. Default: false.
- encoding: The encoding for reading and writing files. Default: UTF-8.
- programExtensions: Extensions for COBOL programs. Default: cob,cbl,cobol,pco.
- copyExtensions: Extensions for COBOL copybooks. Default: cpy,copy.
- include: Comma-separated patterns of files to include. Two asterisks mean 'all directories and subdirectories'. Default: */.
- exclude: Comma-separated patterns of files to include. Default: empty.
- OUTPUT_DIR: Directory where the pre-processed files will be written. Defaults to the current directory.

- Added support for the JavaScript Vue.js framework. The following rules were added:
  - VueComponentDataMustBeFunction: Component data must be a function.
  - VueForWithoutKey: Always use key with v-for.
  - VueHtmlEscapeDisabled: Vue HTML escaping is disabled.
  - VueIfWithForDirective: Never use v-if on the same element as v-for.
- Update CWETOP25 tags to 2021 version
- Added new mapping for the latest 2021 OWASP Top10 list.

Fixed Issues

Kiuwan server

- **SAS-5625** OOM with Insights analysis
- **SAS-5787** Fix long computation times with empty group + artifact dependency when computing obsolescence in Kiuwan

KLA (KiuwanLocalAnalyzer)

- **QAK-6707** Add .jsx extension in the default configuration
- **QAK-6694** Upgrade libraries for running under Java 16
- **QAK-6706** COBOL preprocessor script: Deploy for KLA

Engine

- **QAK-6640** Add support for VUE framework
- **QAK-6642** Possible false positives in rule OPT.CPP.CERTC.EXP33
- **QAK-6643** Possible false positive in rule OPT.CPP.CorrectUseMemoryLeaks
- **QAK-6662** Possible false positive on rule OPT.C.CERTC.STR31
- **QAK-6664** Parsing error JCL
- **QAK-6666** [FP] OPT.JAVA.SEC_JAVA.CrossSiteScriptingRule
- **QAK-6683** False positive / no sense datapath on Java rule: Trust boundary violation
- **QAK-6687** False positives in Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting')
- **QAK-6690** Review EAR rules for the rest of technologies (Go, Kotlin, Objective-C, PHP, Python, Scala)
- **QAK-6691** Inconsistent results for rule "OPT.JAVA.SPRING.AvoidBeansWithTheSameIdAcrossDiferentDescriptors"
- **QAK-6692** False positive "Evaluate integer expressions in a larger size before comparing or assigning to that size" in C file
- **QAK-6694** Upgrade libraries for running under Java 16
- **QAK-6695** CWETOP25:2010:13 should be removed
- **QAK-6698** Update CWETOP25 tags to 2021 version
- **QAK-6699** Bug in PHP rule: Avoid unused private fields
- **QAK-6700** COBOL paser errors (AcuCOBOL)
- **QAK-6701** False positive of PT.JAVA.SEC_JAVA.CodeInjectionWithDeserializationRule (ZD-4720)
- **QAK-6703** COBOL Tandem parse errors
- **QAK-6704** False positive in OPT.KOTLIN.UnreachableCode
- **QAK-6706** COBOL preprocessor script
- **QAK-6707** Adding .jsx extension in the default configuration
- **QAK-6708** Analysis Failing when trying with Java returned 1 and AN-1 errors on both KLA and cloud
- **QAK-6709** Bug on the rule "Follow the limit for number of return statements"
- **QAK-6710** Fix dependency issues in power script parser and rules
- **QAK-6711** [FP] OPT.JAVA.SEC_JAVA.PotentialInfiniteLoop
- **QAK-6712** Possible false positive in the rule OPT.JAVA.FMETODOS.SAOP
- **QAK-6713** Possible false positive in the rule: OPT.JAVA.DECLARA.UCDC
- **QAK-6714** Possible false positive in rule OPT.CPP.CERTC.EXP33
- **QAK-6717** New OWASP ranking
- **QAK-6720** Parsing Error in .VB File
- **QAK-6722** False positive on Prevent denial of service attack through malicious regular expression ('Regex Injection') (ZD-5002)
- **QAK-6723** Parse errors in COBOL app
- **QAK-6724** Unable to parse cobol file: Error at line 1: Encountered: $COPYRIGHT
- **QAK-6725** Parsing Error in .4gl (Informix) files
- **QAK-6727** False positive in OPT.NATURAL.NAT_PF.UseWithLimitClauseInReadAndFind
- **QAK-6728** False positive Improper Control of Generation of Code ('Code Injection') (ZD-5068).
- **QAK-6729** COBOL Parse error: Encountered EXEC PBCF
- **FOG-249** INS - Failure detecting components (null components)
- **FOG-250** glob-base / preserve lost components
- **FOG-251** False Negative: CVE-2021-21252 - jQuery Validation Plugin
- **FOG-252** Missing CVE reference in Insights component
- **FOG-253** Possible error in Insight Vulnerability CVE-2021-23406