[2019-09-12] Change Log

- New version of CQM (2.4.0) and Kiuwan Engine
    - Support for KOTLIN programming language
    - Added Support for RM/COBOL dialect in COBOL technology.
    - Duplicated Code rules are not longer mandatory

New version of CQM (2.4.0) and Kiuwan Engine

Main features are:

- ***Support for KOTLIN programming language***
- ***Support for RM/Cobol dialect***
- ***Duplicated Code rules are not longer mandatory***

CQM is the default Model (i.e. a concrete set of active and pre-configured rules):

- If you are using ***CQM***,
    - ***new rules will automatically become active*** and will be applied to new analyses
- If you are using your own **custom model, your model remains unchaged,** but *you can modify it and activate the new rules* (in case you want to be applied to your code).

You can find new rules by comparing this release of CQM against previous version. A detailed description of the behavior of these new rules is available in rule’s description.

This ***new version of Kiuwan Engine*** incorporates ***bug fixes, performance and reliability improvements in rules and parsers***.

Kiuwan Engine is the binary code executed when an analysis is run.

- ***If the engine is not blocked*** in your Kiuwan account, ***the engine will upgrade automatically*** to the last version of Kiuwan Engine once a new analysis is run
- ***If the engine is blocked***, your kiuwan ***engine will not be modified***.

A ***new version of Kiuwan Engine*** has been released that incorporates ***bug fixes, performance and reliability improvements in rules and parsers***.

Kiuwan Engine is the binary code executed when an analysis is run.

- ***If the engine is not blocked*** in your Kiuwan account, ***the engine will upgrade automatically*** to the last version of Kiuwan Engine once a new analysis is run
- ***If the engine is blocked***, your kiuwan ***engine will not be modified***.

Support for KOTLIN programming language

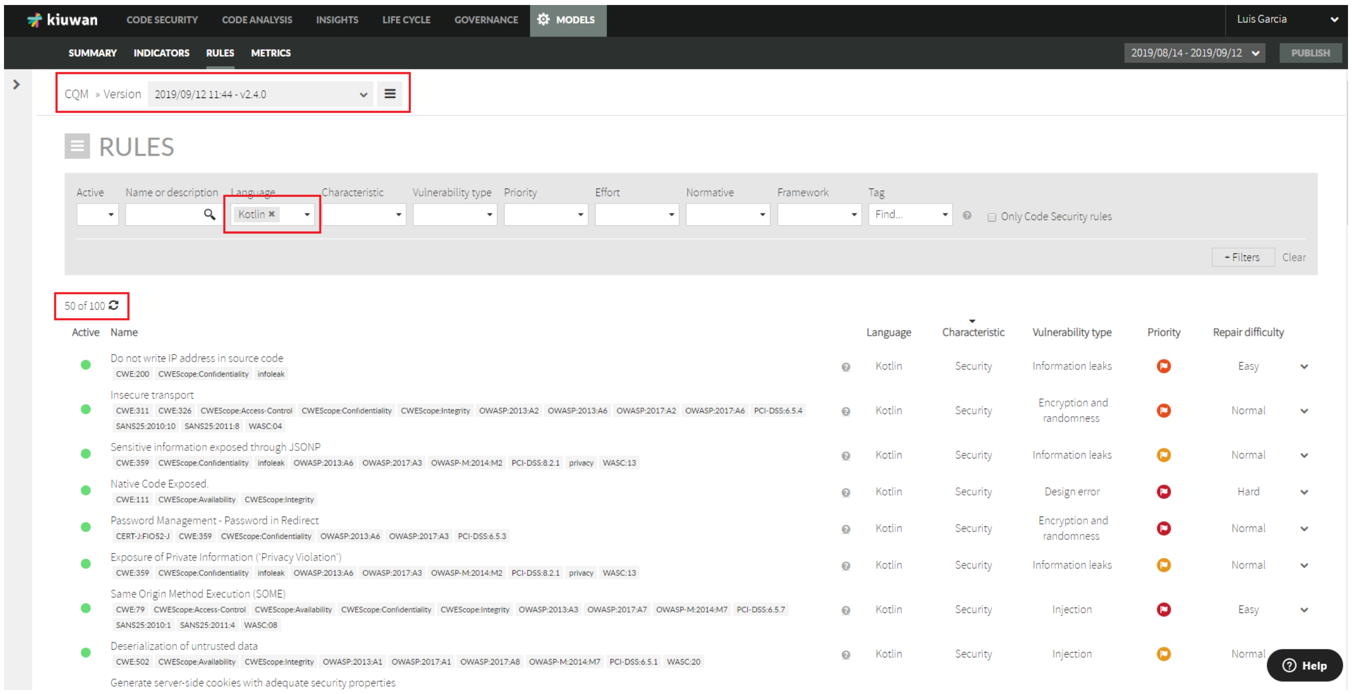
Kotlin is a cross-platform, statically typed, general-purpose programming language with type inference and it's ***Google’s preferred language for Android app development*** since 7 May 2019.

Kotlin is ***officially supported by Google for mobile development on Android***. It targets the JVM, but also compiles to JavaScript or native code (via LLVM), and it's an alternative to the standard Java compiler.

As Kotlin is becoming a widely adopted programming language, ***Kiuwan*** now incorporates ***support to analyze Kotlin source files***, thus searching for code and desing conditions that are indicative of ***security vulnerabilities***.

Kiuwan provides ***100 Rules*** specifically suited to ***Kotlin*** programming language.

You can find these rules going to ***Models Management***, select ***CQM*** and search for ***Rules*** applying to **Kotlin *language***



CQM » Version 2019/09/12 11:44 - v2.4.0

RULES

Active | Name or description | Language: Kotlin | Characteristic | Vulnerability type | Priority | Effort | Normative | Framework | Tag: Find... | Only Code Security rules

50 of 100

Active	Name	Language	Characteristic	Vulnerability type	Priority	Repair difficulty
●	Do not write IP address in source code CWE:200 CWEScope:Confidentiality infoleak	Kotlin	Security	Information leaks	🔴	Easy
●	Insecure transport CWE:311 CWE:326 CWEScope:Access-Control CWEScope:Confidentiality CWEScope:Integrity OWASP:2013:A2 OWASP:2013:A6 OWASP:2017:A2 OWASP:2017:A6 PCI-DSS:6.5.4 SANS25:2010:10 SANS25:2011:8 WASC:04	Kotlin	Security	Encryption and randomness	🔴	Normal
●	Sensitive information exposed through JSONP CWE:359 CWEScope:Confidentiality infoleak OWASP:2013:A6 OWASP:2017:A3 OWASP-M:2014:M2 PCI-DSS:8.2.1 privacy WASC:13	Kotlin	Security	Information leaks	🟠	Normal
●	Native Code Exposed. CWE:111 CWEScope:Availability CWEScope:Integrity	Kotlin	Security	Design error	🔴	Hard
●	Password Management - Password in Redirect CERT-J:FIO52-J CWE:359 CWEScope:Confidentiality OWASP:2013:A6 OWASP:2017:A3 PCI-DSS:6.5.3	Kotlin	Security	Encryption and randomness	🔴	Normal
●	Exposure of Private Information ('Privacy Violation') CWE:359 CWEScope:Confidentiality infoleak OWASP:2013:A6 OWASP:2017:A3 OWASP-M:2014:M2 PCI-DSS:8.2.1 privacy WASC:13	Kotlin	Security	Information leaks	🟠	Normal
●	Same Origin Method Execution (SOME) CWE:79 CWEScope:Access-Control CWEScope:Availability CWEScope:Confidentiality CWEScope:Integrity OWASP:2013:A3 OWASP:2017:A7 OWASP-M:2014:M7 PCI-DSS:6.5.7 SANS25:2010:1 SANS25:2011:4 WASC:08	Kotlin	Security	Injection	🔴	Easy
●	Deserialization of untrusted data CWE:502 CWEScope:Availability CWEScope:Integrity OWASP:2013:A1 OWASP:2017:A1 OWASP:2017:A8 OWASP-M:2014:M7 PCI-DSS:6.5.1 WASC:20	Kotlin	Security	Injection	🟠	Normal
	Generate server-side cookies with adequate security properties					

Added Support for RM/COBOL dialect in COBOL technology.

Support for ***RM/Cobol*** dialect has been added to current list of supported Cobol dialects.

All ***versions between 9.x to 12.16*** (latest version at the moment of writing this post) have been succesfully tested.

Current list of ***supported Cobol dialects***:

- Cobol85,
- OS/VS Cobol II
- IBM Enterprise COBOL for z/OS v5
- IBM ILE COBOL 5
- HP COBOL for TNS (Tandem NonStop)
- AcuCOBOL-GT
- Net Express COBOL 5 / Server Express COBOL
- MicroFocus RM/COBOL 9.x to 12.16

Duplicated Code rules are not longer mandatory

When creating a new model, it was mandatory to activate duplicated code rules (i.e. rules that check for repeated blocks in source code).

This new Kiuwan release no longer forces duplicated code rules to be activated in your custom model.

- Standard pseudo-random number generators cannot withstand cryptographic attacks.
- Denial of Service by externally controlled sleep time
- Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection')

- Too broad privileges granted.
- Avoid using an user controlled Primary Key into a query
- Weak cryptographic hashes cannot guarantee data integrity
- Weak symmetric encryption algorithm.