[2020-02-13] Change Log

Contents:

New version of CQM: 2.5.0

Main features:

- Support for GO
- New RPG-IV rules

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

- If you are using **CQM, the new rules will automatically become active** and will be applied to new analyses.
- If you are using your own **custom model, your model remains unchanged,** but you can modify it and activate the new rules if needed.

Support for GO

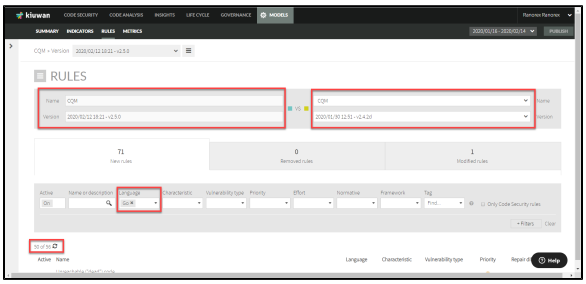
We have added 56 new rules to support SAST and SCA analysis with the GO programming language.

Go is a statically typed, compiled programming language designed by Google. It is syntactically similar to C , but with memory safety, garbage collection, structural typing, and CSP-style concurrency. It has risen in popularity in the last couple of years and is highly demanded in many companies worldwide.

To view the new rules, open the drop-down menu on the upper right corner and select **Models Management.**

Select the **default CQM model** on the left. Click on the **drop-down menu next to the Rules heading** an d select **Compare**.

Here you can compare the latest version of CQM with the previous one. Use the language filter to **filter by Go.**



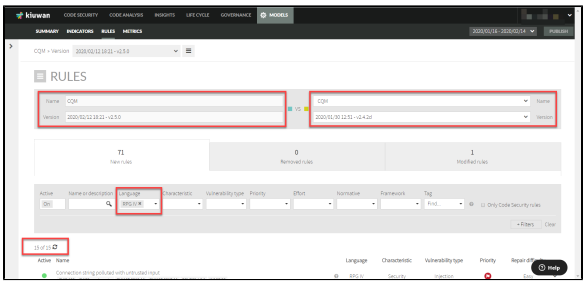
New RPG-IV rules

We have added 15 new rules for RPG-IV.

To view the new rules, open the drop-down menu on the upper right corner and select **Models Management.**

Select the **default CQM model** on the left. Click on the **drop-down menu next to the Rules heading** an d select **Compare**.

Here you can compare the latest version of CQM with the previous one. Use the language filter to **filter by RPG-IV.**



Support for Insights in Life Cycle

The Life Cycle module now supports also running a delivery analysis with Insights.

Two new checkpoints have been added to configure in your audits.:

- **Threshold for maximum Insights components by severity risk**: checks if the number of components that meet the defined severity risk filter is above the defined threshold.
- **Insights filter components by group, name, and version**: checks if any of the discovered components meet the defined group, name, version and comparator filter.

To learn more about it, read our updated Life Cycle documentation:

Audits Management#Insightscomponentscheckpoints

Audits Management#CreateaCheckpoint