[2019-05-28] Change Log

New version of CQM and Kiuwan Engine

Main features of this release are:

1. ***Kiuwan CQM (v2.3.0) and Engine***
   - ***14 New PL-SQL security rules***
   - ***9 New Transact-SQL security rules***

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.

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***.

New PL-SQL security rules

- Improper Neutralization of Special Elements used in an OS Command ('OS Command Injection')
- Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting')
- Cursor Snarfing
- Use of Hard-coded Credentials
- Unvalidated data in HTTP response header or in cookies ('HTTP Response Splitting')
- Do not allow to control the URL used in a redirect by an unvalidated input
- External Control of File Name or Path
- Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting')
- Server-Side Request Forgery (SSRF)
- Potential malicious code
- Unqualified database items in AUTHID CURRENT_USER routine
- Avoid using an user controlled Primary Key into a query
- Weak cryptographic hashes cannot guarantee data integrity
- Weak symmetric encryption algorithm.

New Transact-SQL security rules

- Improper Neutralization of Special Elements used in an OS Command ('OS Command Injection')
- Dangerous procedure / function called.
- 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.