# [2018-09-12] Change Log



---

## New version of CQM, Kiuwan Engine and SAP Extractor

*Main features* of this release are:

1. ***Kiuwan CQM (v1.2.19) and Engine***
   - Increased support for **security** in **VB.NET** (78 new security rules)
   - Increased support for **security** in **Objective-C** (43 new security rules)
   - Additional support for **NoSQL Injection** (added support for **Java, PHP, Python and Objective-C**)
2. **New release of SAP Extractor for Kiuwan**
   a. New SAPEX distribution (as a **Transport Order**)
   b. Support for **On-Demand Analysis of In-progress Change Requests**

### Kiuwan CQM and Engine

A ***new version of CQM*** has been released that incorporate ***new rules*** *(*as detailed below).

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***.

#### Additional support for detection of NoSQL Injection vulnerabilities

Support has been added to Kiuwan to detect *NoSQL Injection vulnerabilities*.

Additionaly to JavaScript, C# and VB.NET, Kiuwan provides support for ***Java, PHP, Python and Objective-C***.

## New VB.NET security rules

- OPT.VBNET.CodeInjection
- OPT.VBNET.CodeInjectionWithDeserialization
- OPT.VBNET.CommandInjection
- OPT.VBNET.CrossSiteRequestForgery
- OPT.VBNET.CrossSiteScripting
- OPT.VBNET.DoSRegexp
- OPT.VBNET.InsecureRandomness
- OPT.VBNET.JSONInjection
- OPT.VBNET.LdapInjection
- OPT.VBNET.MVCNonActionPublicMethods
- OPT.VBNET.MVCPostInControllers
- OPT.VBNET.MVCPreventOverpostingModelDefinition
- OPT.VBNET.MVCPreventUnderpostingModelComposition
- OPT.VBNET.MVCPreventUnderpostingModelDefinition
- OPT.VBNET.MVCRemoveVersionHeader
- OPT.VBNET.OpenRedirect
- OPT.VBNET.PathTraversal
- OPT.VBNET.PotentialInfiniteLoop
- OPT.VBNET.ResourceLeakDatabase
- OPT.VBNET.ResourceLeakLdap
- OPT.VBNET.ResourceLeakStream
- OPT.VBNET.ResourceLeakUnmanaged
- OPT.VBNET.SEC.AccessibilitySubversionRule
- OPT.VBNET.SEC.AnonymousLdapBind
- OPT.VBNET.SEC.AvoidHostNameChecks
- OPT.VBNET.SEC.ConnectionStringParameterPollution
- OPT.VBNET.SEC.CookiesInSecurityDecision
- OPT.VBNET.SEC.CrossSiteHistoryManipulation
- OPT.VBNET.SEC.DangerousFileUpload
- OPT.VBNET.SEC.HardcodedCredential
- OPT.VBNET.SEC.HardcodedCryptoKey
- OPT.VBNET.SEC.HardcodedNetworkAddress
- OPT.VBNET.SEC.HardcodedSalt
- OPT.VBNET.SEC.HttpParameterPollution
- OPT.VBNET.SEC.HttpRequestValueShadowing
- OPT.VBNET.SEC.HttpSplittingRule
- OPT.VBNET.SEC.ImproperAuthentication
- OPT.VBNET.SEC.InformationExposureThroughDebugLog
- OPT.VBNET.SEC.InformationExposureThroughErrorMessage
- OPT.VBNET.SEC.InsecureEmailTransport
- OPT.VBNET.SEC.InsecureTransport
- OPT.VBNET.SEC.LogForging
- OPT.VBNET.SEC.MailCommandInjection
- OPT.VBNET.SEC.MainMethodInWebApplication
- OPT.VBNET.SEC.MissingStandardErrorHandling
- OPT.VBNET.SEC.NoSQLInjection
- OPT.VBNET.SEC.PlaintextStorageOfPassword
- OPT.VBNET.SEC.ProcessControl
- OPT.VBNET.SEC.ProperPaddingWithPublicKeyCrypto
- OPT.VBNET.SEC.RegistryManipulation
- OPT.VBNET.SEC.ResourceInjection
- OPT.VBNET.SEC.SerializableClassContainingSensitiveData
- OPT.VBNET.SEC.ServerInsecureTransport
- OPT.VBNET.SEC.SettingManipulation
- OPT.VBNET.SEC.StaticDatabaseConnection
- OPT.VBNET.SEC.TemporaryFilesLeft
- OPT.VBNET.SEC.TrustBoundaryViolation
- OPT.VBNET.SEC.UnsafeCookieRule
- OPT.VBNET.SEC.UnsafeReflection
- OPT.VBNET.SEC.UnvalidatedAspNetModel
- OPT.VBNET.SEC.UserControlledSQLPrimaryKey
- OPT.VBNET.SEC.XMLEntityInjection
- OPT.VBNET.ServerSideRequestForgery
- OPT.VBNET.SqlInjection
- OPT.VBNET.StoredCrossSiteScripting
- OPT.VBNET.SystemInformationLeak
- OPT.VBNET.TooMuchOriginsAllowed
- OPT.VBNET.UncheckedInputInLoopCondition
- OPT.VBNET.UncheckedReturnValue
- OPT.VBNET.WeakCryptographicHash
- OPT.VBNET.WeakEncryption
- OPT.VBNET.WeakKeySize
- OPT.VBNET.WeakSymmetricEncryptionAlgorithm
- OPT.VBNET.WeakSymmetricEncryptionModeOfOperation
- OPT.VBNET.XMLInjection
- OPT.VBNET.XPathInjection
- OPT.VBNET.XQueryInjection

- OPT.VBNET.XSLTInjection

### New Objective-C security rules

- OPT.OBJECTIVEC.SECURITY.AvoidSMS
- OPT.OBJECTIVEC.SECURITY.BiometricWithoutMessage
- OPT.OBJECTIVEC.SECURITY.CommandInjectionRule
- OPT.OBJECTIVEC.SECURITY.ConnectionStringParameterPollution
- OPT.OBJECTIVEC.SECURITY.ExecutionAfterRedirect
- OPT.OBJECTIVEC.SECURITY.HardcodedCryptoKey
- OPT.OBJECTIVEC.SECURITY.HardcodedIp
- OPT.OBJECTIVEC.SECURITY.HardcodedUsernamePassword
- OPT.OBJECTIVEC.SECURITY.HttpParameterPollutionRule
- OPT.OBJECTIVEC.SECURITY.HttpResponseCachingLeak
- OPT.OBJECTIVEC.SECURITY.HttpSplittingRule
- OPT.OBJECTIVEC.SECURITY.InformationExposureThroughErrorMessage
- OPT.OBJECTIVEC.SECURITY.InsecureTemporaryFile
- OPT.OBJECTIVEC.SECURITY.KeyboardCachingLeak
- OPT.OBJECTIVEC.SECURITY.MailCommandInjection
- OPT.OBJECTIVEC.SECURITY.MissingContentValidation
- OPT.OBJECTIVEC.SECURITY.MissingPasswordFieldMasking
- OPT.OBJECTIVEC.SECURITY.NoSQLInjection
- OPT.OBJECTIVEC.SECURITY.LogForging
- OPT.OBJECTIVEC.SECURITY.PasswordInCommentRule
- OPT.OBJECTIVEC.SECURITY.PasswordInConfigurationFile
- OPT.OBJECTIVEC.SECURITY.PasteboardCachingLeak
- OPT.OBJECTIVEC.SECURITY.PlaintextStorageInACookieRule
- OPT.OBJECTIVEC.SECURITY.PotentialInfiniteLoop
- OPT.OBJECTIVEC.SECURITY.PrivacyViolation
- OPT.OBJECTIVEC.SECURITY.ResourceInjection
- OPT.OBJECTIVEC.SECURITY.ScreenCachingLeak
- OPT.OBJECTIVEC.SECURITY.SensitiveCoreData
- OPT.OBJECTIVEC.SECURITY.SensitiveDataAccessedFromItunes
- OPT.OBJECTIVEC.SECURITY.SensitiveNoSQL
- OPT.OBJECTIVEC.SECURITY.SensitiveSQL
- OPT.OBJECTIVEC.SECURITY.SensitiveUserDefaults
- OPT.OBJECTIVEC.SECURITY.SerializableClassContainingSensitiveData
- OPT.OBJECTIVEC.SECURITY.SerializationInjection
- OPT.OBJECTIVEC.SECURITY.ServerTrustCredentialCheck
- OPT.OBJECTIVEC.SECURITY.ThirdPartyKeyboardAllowed
- OPT.OBJECTIVEC.SECURITY.UncheckedInputInLoopCondition
- OPT.OBJECTIVEC.SECURITY.UnsafeCookie
- OPT.OBJECTIVEC.SECURITY.URLSchemeHijacking
- OPT.OBJECTIVEC.SECURITY.UserControlledSQLPrimaryKey
- OPT.OBJECTIVEC.SECURITY.WeakKeyDerivationIteration
- OPT.OBJECTIVEC.SECURITY.WeakKeyDerivationPassword
- OPT.OBJECTIVEC.SECURITY.XMLInjection

## New release of SAP Extractor for Kiuwan

### New SAPEX distribution (as a Transport Order)

Older versions of SAPEX were delivered as source files that needed to be manually imported into SAP.

This version includes a Transport Order that you can directly import into your SAP system, avoiding manually import all the SAPEX sources.

You can find detailed info at SAP Extractor for Kiuwan

### Support for On-Demand Analysis of In-progress Change Requests

This new version include SAP programs so you may analyze a Chage Request (or Task) currently in progress.

You can find detailed info at SAP Extractor for Kiuwan and Local use - In-Progress Deliveries