How to manage Kiuwan defects when I do not completely agree with them

This guide shows you how to manage Kiuwan defects in four different scenarios.

Contents:

Introduction

Kiuwan gathers evidence from the application's source code using in-house developed static analyzers and rules.

During source code scanning, Kiuwan looks for patterns (*bad smells*) that are not conformant to widely accepted **best programming practices**, from security to quality points of view. We call those evidence **defects** (or **vulnerabilities**, if related to security).

The term **defect** has an obvious negative connotation. However, depending on the scenario, it might happen that the identified defect is triggered by a rule that is not needed for a specific application. In this case, Kiuwan provides useful mechanisms to manage similar situations.

Here is an explanation of an everyday scenario:

If a doctor (Dr. Kiuwan) checks the health of a patient and detects overweight, lack of exercise and bad nutrition in the patient, these are no doubt defects if you follow best health practices. Dr. Kiuwan can detect this and raise a flag.

These defects are not applicable in every case. For example, in the case of a patient with heart problems, the defect of lack of exercise is not needed. Dr. Kiuwan can change the priority of this defect or remove it completely.

Scenario 1: “This rule is not applicable at all to my code”

Deactivate the rule if it does not make sense for the application.

For example, there’s a common bad practice that is to allow the use of the back navigation button in a web application. It’s commonly accepted that you should “catch” the use of the back button and do not rely on the browser.

Some mobile applications are built using some frameworks (Cordova, for example) that generate javascript and HTML code that does not prevent the use of the back button. In these cases, the above rule does not make sense. It does not apply at all to the application, so the obvious solution is to deactivate it.

Please, visit How to Deactivate a Rule for help on how to do it.

Scenario 2: “The rule is generally applicable, but this defect is not really a defect”

You could find that a Kiuwan rule is generally helpful and must be kept active. But, in some concrete cases, it is not applicable (or it is not properly working) and some defects should not be considered in the analysis.

Reasons to silence (or mute) those defects can be of very different nature. It could be that in some specific points the rule is not applicable, or might be false positives.

Use the **Mute Defect** functionality to keep the rule active but discard those specific defects.

You can fine-tune this muting mechanism to suit your needs:

- Mute concrete defects in specific source files
- Mute a complete source file
- Mute a rule

After muting, those defects will not be considered in the Kiuwan indicators' calculations.

Please visit Muting defects for help on how to do it.

In case you are muting a defect because you consider that “is not a defect” and the rule is not properly working (i.e. a false positive), please do not hesitate to contact us to report the issue. We will do our best to fix it as soon as possible.

Scenario 3: “Some files should not be analyzed and defects found in those files should be not be raised”

A very common scenario is when you are analyzing your code but you are obtaining defects in files that you cannot fix for whatever reason. Common reasons are because those files are open source, generated code or third-party code.

> **Exclude those files from the analysis** if you cannot modify the source code, but their defects are decreasing the indicators of your application.

Kiuwan provides an exclusion mechanism by which you can tell Kiuwan what files should not be analyzed (individually or matched by regular expressions) or even what files only should be analyzed.

You can visit Manage Different Analyses in Kiuwan and Advanced KLA How-Tos to see how to do it (in the cloud or locally, respectively).

Scenario 4: “Some rules would need to be adapted to my circumstance to properly work”

The Kiuwan rules come with default parameters, however, most of them can be customized.

> If you find that a rule does not exactly cover your needs, please look at its **configuration parameters**. It's likely that you find how to instruct the rule to suit your needs.

Depending on the rule, the customization level can be different, so you should always consult the rule documentation to know the customization level and tune its behavior according to your needs.

Please visit Rules Management for further information.