Secrets
ABAP
Apex
C
C++
CloudFormation
COBOL
C#
CSS
Flex
Go
HTML
Java
JavaScript
Kotlin
Objective C
PHP
PL/I
PL/SQL
Python
RPG
Ruby
Scala
Swift
Terraform
Text
TypeScript
T-SQL
VB.NET
VB6
XML

Python static code analysis

Unique rules to find Bugs, Vulnerabilities, Security Hotspots, and Code Smells in your PYTHON code

All rules 216 | Vulnerability 29 | Bug 55 | Security Hotspot 31 | Code Smell 101

Tags

Search by name...

Code Smell

Local variable and function parameter names should comply with a naming convention

Code Smell

Field names should comply with a naming convention

Code Smell

Class names should comply with a naming convention

Code Smell

Method names should comply with a naming convention

Code Smell

Track uses of "TODO" tags

Code Smell

HTML autoescape mechanism should not be globally disabled

Vulnerability

Variables, classes and functions should be either defined or imported

Bug

"__exit__" should accept type, value, and traceback arguments

Bug

"return" and "yield" should not be used in the same function

Bug

Track lack of copyright and license headers

Code Smell

HTTP response headers should not be vulnerable to injection attacks

Collection content should not be replaced unconditionally

Analyze your code

Bug Major suspicious

It is highly suspicious when a value is saved in a collection for a given key or index and then unconditionally overwritten. Such replacements are likely errors.

This rule raises an issue when the `__setitem__` method of the same object is called multiple times with the same index, slice or key without any other action done between the calls.

Noncompliant Code Example

```
def swap(mylist, index1, index2):
    tmp = mylist[index2]
    mylist[index2] = mylist[index1]
    mylist[index2] = tmp  # Noncompliant

list2 = [0,1,2,3,4,5,6,7,8,9]
list2[3:5] = [42,42]
list2[3:5] = [42,42]  # Noncompliant

mymap = {'a': {}}
mymap['a']['b'] = 42
mymap['a']['b'] = 42  # Noncompliant
```

Available In:

sonarlint sonarcloud sonarqube

Vulnerability

Regular expressions should be syntactically valid

Bug

Sending emails is security-sensitive

Security Hotspot

Reading the Standard Input is security-sensitive

Security Hotspot

Using command line arguments is security-sensitive

Security Hotspot