

-  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... 

| |
|--|
| Functions should not have too many lines of code |
|  Code Smell |
| Track uses of "NOSONAR" comments |
|  Code Smell |
| Track comments matching a regular expression |
|  Code Smell |
| Statements should be on separate lines |
|  Code Smell |
| Functions should not contain too many return statements |
|  Code Smell |
| Files should not have too many lines of code |
|  Code Smell |
| Lines should not be too long |
|  Code Smell |
| Methods and properties that don't access instance data should be static |
|  Code Smell |
| New-style classes should be used |
|  Code Smell |
| Parentheses should not be used after certain keywords |
|  Code Smell |
| Track "TODO" and "FIXME" comments that do not contain a reference to a person |
|  Code Smell |
| Module names should comply with a naming convention |

Creating cookies without the "secure" flag is security-sensitive

Analyze your code

 Security Hotspot

 Minor 

 cwe privacy sans-top25 owasp

When a cookie is protected with the `secure` attribute set to `true` it will not be sent by the browser over an unencrypted HTTP request and thus cannot be observed by an unauthorized person during a man-in-the-middle attack.

Ask Yourself Whether

- the cookie is for instance a *session-cookie* not designed to be sent over non-HTTPS communication.
- it's not sure that the website contains **mixed content** or not (ie HTTPS everywhere or not)

There is a risk if you answered yes to any of those questions.

Recommended Secure Coding Practices

- It is recommended to use HTTPS everywhere so setting the `secure` flag to `true` should be the default behaviour when creating cookies.
- Set the `secure` flag to `true` for session-cookies.

Sensitive Code Example

Flask

```
from flask import Response

@app.route('/')
def index():
    response = Response()
    response.set_cookie('key', 'value') # Sensitive
    return response
```

Compliant Solution

Flask

```
from flask import Response

@app.route('/')
def index():
    response = Response()
    response.set_cookie('key', 'value', secure=True) #
    return response
```

See

Code Smell

Comments should not be located at the end of lines of code

Code Smell

Lines should not end with trailing whitespaces

Code Smell

Files should contain an empty newline at the end

Code Smell

Long suffix "L" should be upper case

Code Smell

- [OWASP Top 10 2021 Category A4](#) - Insecure Design
- [OWASP Top 10 2021 Category A5](#) - Security Misconfiguration
- [OWASP Top 10 2017 Category A3](#) - Sensitive Data Exposure
- [MITRE, CWE-311](#) - Missing Encryption of Sensitive Data
- [MITRE, CWE-315](#) - Cleartext Storage of Sensitive Information in a Cookie
- [MITRE, CWE-614](#) - Sensitive Cookie in HTTPS Session Without 'Secure' Attribute
- [SANS Top 25](#) - Porous Defenses

Available In:

sonarcloud  | sonarqube 

© 2008-2022 SonarSource S.A., Switzerland. All content is copyright protected. SONAR, SONARSOURCE, SONARLINT, SONARQUBE and SONARCLOUD are trademarks of SonarSource S.A. All other trademarks and copyrights are the property of their respective owners. All rights are expressly reserved.
[Privacy Policy](#)