

- Secrets
- ABAP
- Apex
- C
- C++
- CloudFormation
- COBOL
- 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... 🔍

🐞 Bug
Track lack of copyright and license headers
🕸 Code Smell
HTTP response headers should not be vulnerable to injection attacks
🔒 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
Encrypting data is security-sensitive
🛡 Security Hotspot
Using regular expressions is security-sensitive
🛡 Security Hotspot
Dynamically executing code is security-sensitive
🛡 Security Hotspot
Cyclomatic Complexity of functions should not be too high
🕸 Code Smell
Control flow statements "if", "for", "while", "try" and "with" should not be nested too deeply
🕸 Code Smell

Increment and decrement operators should not be used

Analyze your code

🐞 Bug ⬆ Major ?

Python has no pre/post increment/decrement operator. For instance, `x++` and `x--` will fail to parse. More importantly, `++x` and `--x` will do nothing. To increment a number, simply write `x += 1`.

Noncompliant Code Example

```
++x # Noncompliant
```

Compliant Solution

```
x += 1
```

Available In:

sonarlint 🚫 | 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](#)


Cyclomatic Complexity of classes should not be too high

 Code Smell

"\" should only be used as an escape character outside of raw strings

 Bug

Using shell interpreter when executing OS commands is security-sensitive

 Security Hotspot

Functions should use "return" consistently

 Code Smell

Python parser failure