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

Function return types should be consistent with their type hint

Code Smell

Character classes in regular expressions should not contain the same character twice

Code Smell

Type checks shouldn't be confusing

Code Smell

Regular expressions should not be too complicated

Code Smell

Builtins should not be shadowed by local variables

Code Smell

Implicit string and byte concatenations should not be confusing

Code Smell

Identity comparisons should not be used with cached typed

Code Smell

Expressions creating sets should not have duplicate values

Code Smell

Expressions creating dictionaries should not have duplicate keys

Code Smell

Special method "__exit__" should not re-raise the provided exception

Code Smell

Unused scope-limited definitions should be removed

Using non-standard cryptographic algorithms is security-sensitive

Analyze your code

Security Hotspot | Critical | cwe sans-top25 owasp

The use of a non-standard algorithm is dangerous because a determined attacker may be able to break the algorithm and compromise whatever data has been protected. Standard algorithms like `Argon2PasswordHasher`, `BCryptPasswordHasher`, … should be used instead.

This rule tracks creation of `BasePasswordHasher` subclasses for Django applications.

Recommended Secure Coding Practices

- Use a standard algorithm instead of creating a custom one.

Sensitive Code Example

```
class CustomPasswordHasher(BasePasswordHasher):  # Sens
    # ...
```

See

- OWASP Top 10 2021 Category A2 - Cryptographic Failures
- OWASP Top 10 2017 Category A3 - Sensitive Data Exposure
- MITRE, CWE-327 - Use of a Broken or Risky Cryptographic Algorithm
- SANS Top 25 - Porous Defenses

Available In:

sonarcloud | sonarqube

Code Smell

Functions and methods should not have identical implementations

Code Smell

Unused private nested classes should be removed

Code Smell

String formatting should be used correctly

Code Smell

Conditional expressions should not be nested

Code Smell