Security

Hotspot

Tags

Security Hotspot Oritical

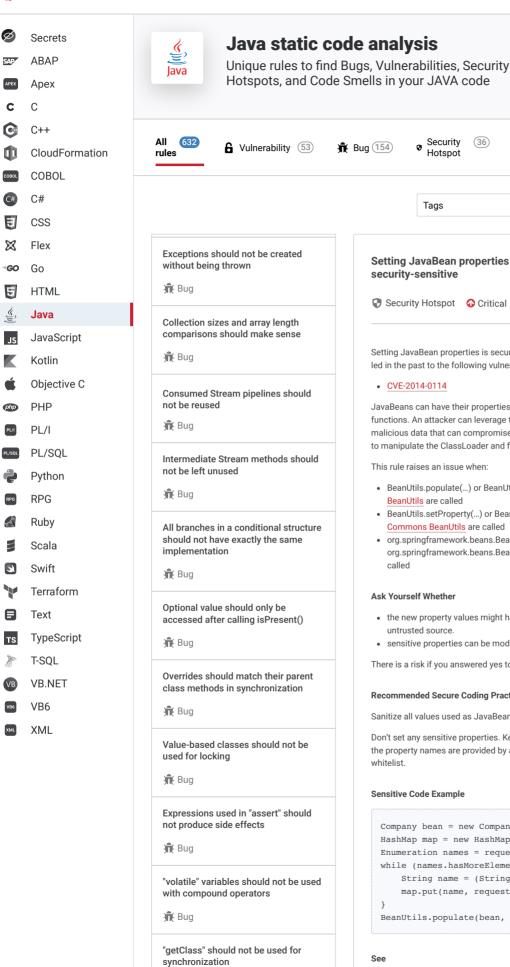
led in the past to the following vulnerability:

(36)



Products ✓

O Quick 42 Fix





⊗ Code

Smell

(389)

Search by name.

cwe owasp cert

Setting JavaBean properties is security sensitive. Doing it with untrusted values has

• CVE-2014-0114

JavaBeans can have their properties or nested properties set by population functions. An attacker can leverage this feature to push into the JavaBean malicious data that can compromise the software integrity. A typical attack will try to manipulate the ClassLoader and finally execute malicious code.

This rule raises an issue when:

- BeanUtils.populate(...) or BeanUtilsBean.populate(...) from Apache Commons BeanLitils are called
- BeanUtils.setProperty(...) or BeanUtilsBean.setProperty(...) from Apache Commons BeanUtils are called
- org.springframework.beans.BeanWrapper.setPropertyValue(...) or org.springframework.beans.BeanWrapper.setPropertyValues(...) from Spring is called

## Ask Yourself Whether

- the new property values might have been tampered with or provided by an untrusted source
- sensitive properties can be modified, for example: class.classLoader

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

## **Recommended Secure Coding Practices**

Sanitize all values used as JavaBean properties.

Don't set any sensitive properties. Keep full control over which properties are set. If the property names are provided by an unstrusted source, filter them with a whitelist

## Sensitive Code Example

```
Company bean = new Company();
HashMap map = new HashMap();
Enumeration names = request.getParameterNames();
while (names.hasMoreElements()) {
   String name = (String) names.nextElement();
    map.put(name, request.getParameterValues(name));
BeanUtils.populate(bean, map); // Sensitive: "map" is popula
```

- OWASP Top 10 2021 Category A3 Injection
- OWASP Top 10 2021 Category A8 Software and Data Integrity Failures
- OWASP Top 10 2017 Category A1 Injection

📆 Bug

Min and may used in combination

should not always return the same value



Assignment of lazy-initialized members should be the last step with double-checked locking



"String" calls should not go beyond their bounds



Raw byte values should not be used in bitwise operations in combination with shifts

- <u>MITRE, CWE-915</u> Improperly Controlled Modification of Dynamically-Determined Object Attributes
- <u>CERT, MSC61-J.</u> Do not use insecure or weak cryptographic algorithms
- Derived from FindSecBugs rule BEAN\_PROPERTY\_INJECTION

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