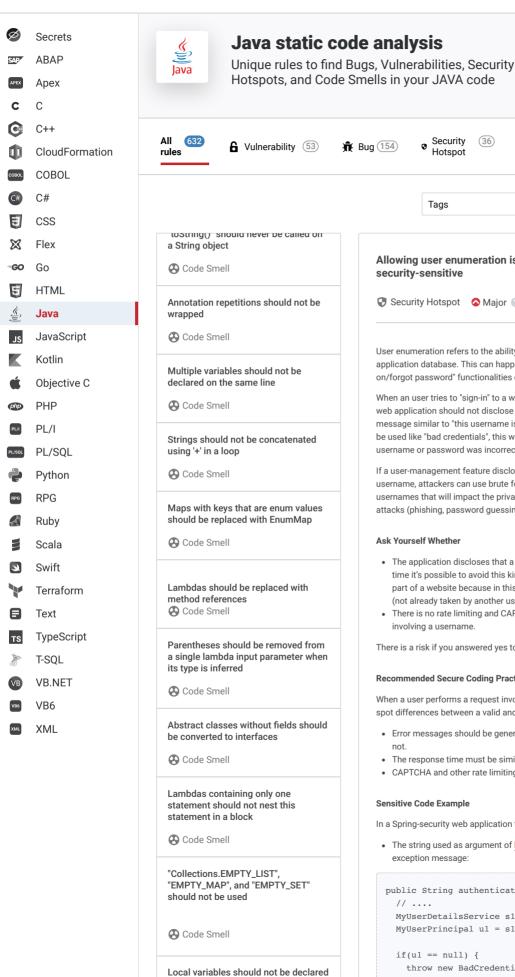
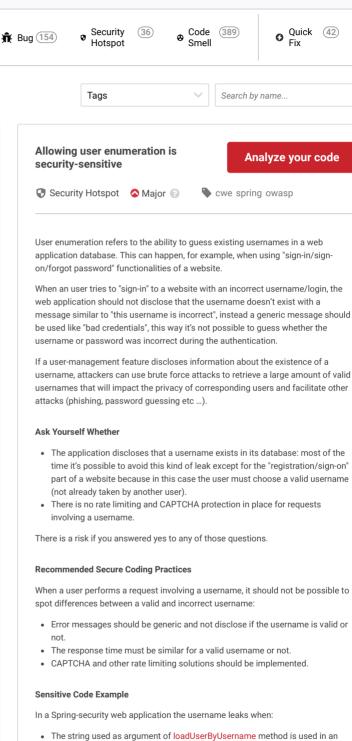


Products ✓





exception message:

if(u1 == null) {

public String authenticate(String username, String password)

throw new BadCredentialsException(username+" doesn't exi

MyUserDetailsService s1 = new MyUserDetailsService(); MyUserPrincipal u1 = s1.loadUserByUsername(username);

and then immediately returned or

thrown

Oude Silieli

Unused local variables should be removed

Code Smell

Private fields only used as local variables in methods should become local variables

Code Smell

"public static" fields should be constant

Code Smell

Loops should not contain more than a single "break" or "continue" statement

• <u>UsernameNotFoundException</u> is thrown (except when it is in the loadUserByUsername method):

```
public String authenticate(String username, String password)
// ....
if(user == null) {
    throw new UsernameNotFoundException("user not found");
}
// ....
}
```

• HideUserNotFoundExceptions is set to false:

DaoAuthenticationProvider daoauth = new DaoAuthenticationPro daoauth.setUserDetailsService(new MyUserDetailsService()); daoauth.setPasswordEncoder(new BCryptPasswordEncoder()); daoauth.setHideUserNotFoundExceptions(false); // Sensitive builder.authenticationProvider(daoauth);

Compliant Solution

In a Spring-security web application:

 the same message should be used regardless of whether it is the wrong user or password:

```
public String authenticate(String username, String password)
  Details user = null;
try {
    user = loadUserByUsername(username);
} catch (UsernameNotFoundException | DataAccessException e
    // Hide this exception reason to not disclose that the u
}
if (user == null || !user.isPasswordCorrect(password)) {
    // User should not be able to guess if the bad credenti
    throw new BadCredentialsException("Bad credentials");
}
```

• HideUserNotFoundExceptions should be set to true:

DaoAuthenticationProvider daoauth = new DaoAuthenticationPro daoauth.setUserDetailsService(new MyUserDetailsService()); daoauth.setPasswordEncoder(new BCryptPasswordEncoder()); daoauth.setHideUserNotFoundExceptions(true); // Compliant builder.authenticationProvider(daoauth);

See

- OWASP Top 10 2021 Category A1 Broken Access Control
- OWASP Top 10 2017 Category A2 Broken Authentication
- MITRE, CWE-200 Exposure of Sensitive Information to an Unauthorized Actor

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