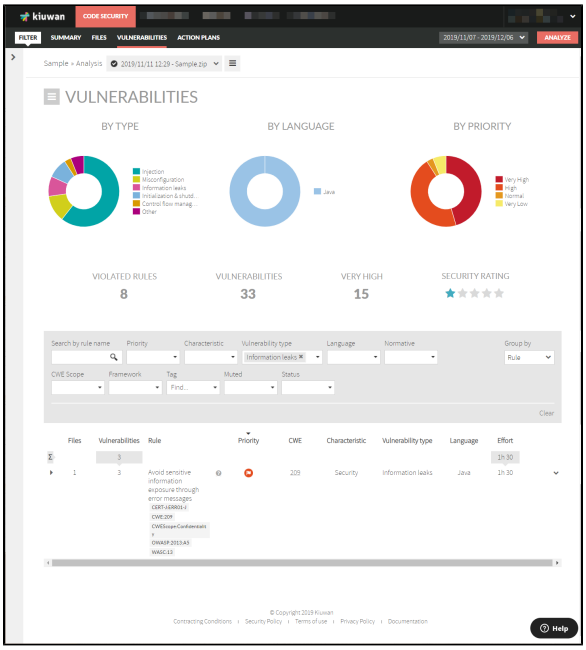
Vulnerabilities

This section will guide you through the **vulnerabilities** dashboard of Kiuwan Code Security.

Contents:

The Code Security Vulnerabilities dashboard provides a detailed view of the application's vulnerabilities. It allows you to:

- Search for vulnerabilities according to multiple search criteria
- Order and group vulnerabilities by different characteristics
- Inspect details of every single vulnerabilities
- Access to vulnerability description and remediation tips



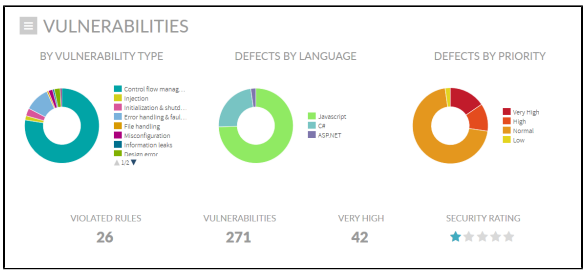
Summary data

The summary section displays group information on vulnerabilities:

- **By Vulnerability** Type: number of vulnerabilities for every type (please see Vulnerability Types)
- **By Language**: number of vulnerabilities found for every programming language
- **By Priority**: number of vulnerabilities found by priority (according to security rules priorities as defined in the model used for the analysis)

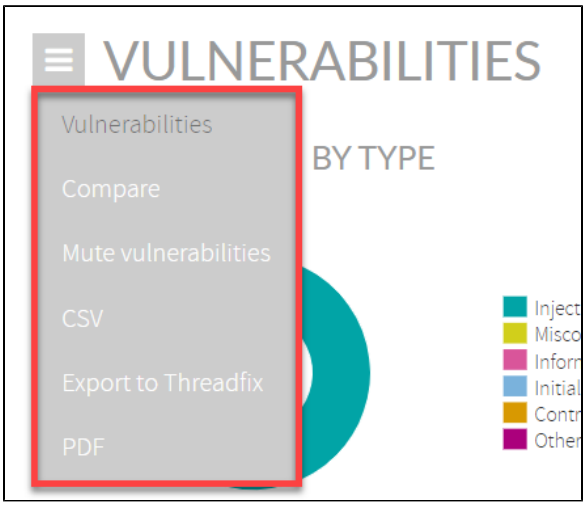
Figures are also displayed for

- **Violated Rules**: number of security rules (checks) with associated vulnerabilities
- **Vulnerabilities**: total number of vulnerabilities found in app source code
- **Very High**: number of Very High vulnerabilities
- **Security Rating**: overall application security rating



Click the burger menu on the top-left to:

- Compare analysis results with any other analysis
- Mute vulnerabilities
- Export vulnerabilities to CSV format
- Export to ThreadFix
- Export in PDF format



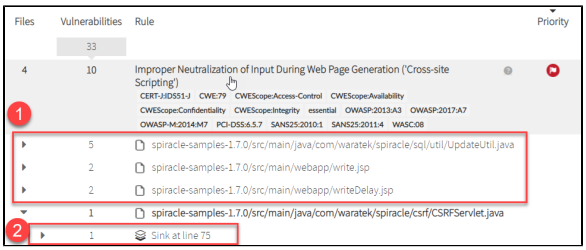
Detailed data

Along with these metrics, Vulnerability page displays a full list of defects that you can browse, filter and order by following criteria:

- Search by rule name
- Priority - low to high
- Characteristic - main software analytics categorization of the selected rule
- Vulnerability Type - security topic addressed by the selected security rule
- Language - programming language
- Normative - security standard
- CWE Scope
- Framework
- Tags
- Muted - muted rules or not
- Status - reviewed or not

You can also group by **Rule** or **Files**

Click on a vulnerability row to see more details:



1. Which files contain this vulnerability
2. The specific location of the vulnerability in the source code

For further information, please visit Understanding Data-Flow Vulnerabilities