


# Maven Surefire Plugin

The [Surefire Plugin](#) is used during the `test` phase of the `build lifecycle` to execute the `unit tests` of an application. It generates `reports` in two different file formats:

- Plain text files ( `*.txt` )
- XML files ( `*.xml` )

By default, these files are generated at `${basedir}/target/surefire-reports`.


For an HTML format of the report, please see the [Maven Surefire Report Plugin](#) .


## Goals Overview

The Surefire Plugin has only one goal:

- `surefire:test` runs the `unit tests` of an application.

## Usage

General instructions on [how to use the Surefire Plugin](#) can be found on the [usage page](#). Some more specific use cases are described in the examples listed below. Last but not least, users occasionally contribute additional examples, tips or errata to the [plugin's wiki page](#) . In case you still have questions regarding the plugin's usage, please have a look at the [FAQ](#) and feel free to contact the [user mailing list](#). The posts to the mailing list are archived and could already contain the answer to your question as part of an older thread. Hence, it is also worth browsing/searching the [mail archive](#).

If you feel like the plugin is missing a feature or has a defect, you can file a feature request or bug report in our [issue tracker](#). When creating a new issue, please provide a comprehensive description of your concern. Especially for fixing bugs it is crucial that the developers can reproduce your problem. For this reason, entire debug logs, POMs or most preferably little demo projects attached to the issue are very much appreciated. Of course, patches are welcome, too. Contributors can check out the project from our [source repository](#) and will find supplementary information in the [guide to helping with Maven](#) .

## Examples

The following examples show how to use the Surefire Plugin in more advanced use cases:

- [Using TestNG](#)
- [Using JUnit](#)
- [Using POJO Tests](#)
- [Skipping Tests](#)
- [Inclusions and Exclusions of Tests](#)
- [Running a Single Test](#)
- [Class Loading and Forking](#)
- [Debugging Tests](#)
- [Using System Properties](#)
- [Configuring the Classpath](#)
- [Selecting Providers](#)
- [Fork Options and Parallel Test Execution](#)