Tellus Plugin

Plugin Information

View TestLink on the plugin site for more information.

⚠️ The current version of this plugin may not be safe to use. Please review the following warnings before use:

- Credentials stored in plain text

⚠️ Older versions of this plugin may not be safe to use. Please review the following warnings before using an older version:

- Stored cross-site scripting vulnerability

⚠️ This plugin is up for adoption. Want to help improve this plugin? Click here to learn more!

This plug-in integrates Jenkins and TestLink and generates reports on automated test execution. With this plug-in you can manage your tests in TestLink, schedule and control in Jenkins, and execute using your favorite test execution tool (TestPartner, Selenium, TestNG, Perl modules, PHPUnit, among others). This plug-in is not part of TestLink project, issues regarding Jenkins and TestLink integration must be reported in Jenkins JIRA. It has been reported by users that it is also possible to use ReportNG, as this tool has an option to output TestNG XML too.

⚠️ When filing an issue, remember to include as much data as possible, but avoid including sensitive data, as attachments can take a while to be deleted from JIRA, as the process is manual and quite troublesome.

Jenkins TestLink Plug-in Tutorial

Check out this tutorial with step-by-step instructions on how to integrate Jenkins and TestLink using TestLink Plug-in.

- Jenkins TestLink Plug-in Tutorial (HTML)
- Jenkins TestLink Plug-in Tutorial (PDF)
- Jenkins TestLink Plug-in Tutorial (HTML) (German)
- Jenkins TestLink Plug-in Tutorial (HTML) (French)
- Jenkins TestLink Plug-in Tutorial (HTML) (Spanish)
- Jenkins TestLink Plug-in Tutorial (HTML) (Brazilian Portuguese)

Who is using Jenkins TestLink Plug-in

Let us know if you are using this plug-in.

Description

The TestLink plug-in integrates Jenkins with TestLink. TestLink plug-in uses testlink-java-api to access TestLink XML-RPC API. With the information that you provide in the build step configuration the plug-in retrieves automated tests from TestLink. With the plug-in, you are able to execute build steps that call testing tools. It reads TestNG, JUnit and TAP test report formats, used to update TestLink test cases’ executions.
The job configuration changed! Now there are three sections, what helps you to configure your job, and add extra build steps, result seeking strategies and general information about TestLink.

**TestLink configuration section**

**Test Execution section**
You can use environment variables in Test project name, Test plan name or Build name. For instance, build-project-$BUILD_NUMBER would be replaced by build-project- and the number of the Build executed in Jenkins.
TestLink Results

- **Build ID:** 11
- **Build Name:** build-lua-2011-10-06_21-35-07
- **Passed:** 0
- **Failed:** 1
- **Blocked:** 0
- **Not Run:** 0
- **Total:** 1
How does the plug-in execute my automated tests?

TestLink Plug-in retrieves automated test cases data from TestLink. This data is then set as environment variables. This way your build steps can make use of this information to execute your automated tests.

For instance, you could run ant passing a Test Class as parameter, more or less something like this, /opt/java/apache-ant-1.8.0/bin/ant -DtestClass=$TESTLINK_TESTCASE_TEST_CLASS. In this case, Test Case is the name of a custom field in TestLink.

List of environment variables available during TestLink Plug-in execution of single build steps (JENKINS-17622):

- TESTLINK_BUILD_NAME
- TESTLINK_TESTPROJECT_NAME
- TESTLINK_TESTCASE_TOTAL

List of environment variables available during TestLink Plug-in execution of iterative build steps:

- TESTLINK_TESTCASE_ID
- TESTLINK_TESTCASE_NAME
- TESTLINK_TESTCASE_TOTAL
- TESTLINK_TESTCASE_AUTHOR
- TESTLINK_TESTCASE_SUMMARY
- TESTLINK_BUILD_NAME
- TESTLINK_TESTCASE_TESTSUITELD
- TESTLINK_TESTPLAN_NAME
- TESTLINK_TESTCASE_TESTSUITENAME
- TESTLINK_TESTCASE тест_custom_field_name

For each Custom Field found, TestLink Plug-in defines an environment variable for with its name and value, as shown above (%CUSTOM_FIELD_NAME%). If the custom field name contains spaces the plug-in will convert them into _'s, i.e., Test Class can be accessed as %TESTLINK_TESTCASE_TEST_CLASS% in Windows or $TESTLINK_TESTCASE_TEST_CLASS% in *nix-like systems.

How does the plug-in know if a test passed or failed?

You can choose among different result seeking strategies in your job configuration. The plug-in supports three result formats, TestNG, JUnit and TAP.

TestNG result seeking strategies

**TestNG class name**: The plug-in matches the TestNG class name (e.g.: org.tap4j.TestParser) and the key custom field value.
**TestNG method name:** The plug-in matches the TestNG method name (e.g.: org.tap4j.TestParser#testTokenizer) and the key custom field value.

**TestNG suite name:** The plug-in matches the TestNG suite name (e.g.: functional tests) and the key custom field value.

**JUnit result seeking strategies**

**JUnit case class name:** The plug-in matches the JUnit case class name (e.g.: org.tap4j.TestParser) and the key custom field value.

**JUnit case name:** The plug-in matches the JUnit case class name (e.g.: testTokenizer) and the key custom field value.

**JUnit method name:** The plug-in matches the JUnit case class name (e.g.: org.tap4j.TestParser#testTokenizer) and the key custom field value.

**JUnit suite name:** The plug-in matches the JUnit suite name (e.g.: functional tests) and the key custom field value.

**TAP result seeking strategies**

**TAP file name:** The plug-in matches the TAP file name (e.g.: testFtpProtocol.tap) and the key custom field value.

**TAP file name multiple test points:** Adds one test execution in TestLink for each TAP Test Result with its execution status.

### Configuration

1. Download and install the latest version of TestLink ([http://testlink.org/](http://testlink.org/)).
2. Create automated tests in TestLink with Custom Field for automation.
3. Install the TestLink plug-in from the Jenkins Plugin Manager.
4. Define a TestLink configuration from the Configure System page.
5. Add a Invoke TestLink build step in the job you want execute automated tests.
6. Configure the required properties.

### Languages Supported

1. English (American)
2. Portuguese (Brazil)
3. Spanish (Thanks to César Fernandes de Almeida)
4. French (Thanks to Floréal Toumikian, Olivier Renault, Latifa Elkarama)

Want to see this plug-in in your language? Send us an e-mail and we will get in touch with the text that needs to be translated.

### Compatibility Matrix

The plug-in 1.0 and 1.1 versions were pilot projects that should be ignored. They have issues in distributed environments and depend on Maven. Please, consider updating your plug-in to the 2.0 version.

<table>
<thead>
<tr>
<th>Plugin</th>
<th>1.0</th>
<th>1.1</th>
<th>2.0</th>
<th>2.1</th>
<th>2.2</th>
<th>2.2.1</th>
<th>2.2.2</th>
<th>2.4</th>
<th>2.5</th>
<th>3.0</th>
<th>3.1</th>
<th>3.2</th>
<th>3.3</th>
<th>3.4</th>
<th>3.5</th>
<th>3.6</th>
<th>3.7</th>
<th>3.8</th>
<th>3.9</th>
<th>3.10</th>
<th>3.11</th>
</tr>
</thead>
<tbody>
<tr>
<td>TestLink</td>
<td>1.9.0</td>
<td>1.9.0</td>
<td>1.9.0</td>
<td>1.9.0</td>
<td>1.9.0</td>
<td>1.9.0</td>
<td>1.9.2</td>
<td>1.9.3</td>
<td>1.9.3</td>
<td>1.9.4 and 1.9.5</td>
<td>1.9.6</td>
<td>1.9.6</td>
<td>1.9.7</td>
<td>1.9.7</td>
<td>1.9.8</td>
<td>1.9.8</td>
<td>1.9.8 + 1.9.14</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jenkins</td>
<td>1.366</td>
<td>1.366</td>
<td>1.388+</td>
<td>1.388+</td>
<td>1.388+</td>
<td>1.388+</td>
<td>1.388+</td>
<td>1.388+</td>
<td>1.398+</td>
<td>1.398+</td>
<td>1.466+</td>
<td>1.466+</td>
<td>1.466+</td>
<td>1.466+</td>
<td>1.466+</td>
<td>1.466+</td>
<td>1.466+</td>
<td>1.466+</td>
<td>1.466+</td>
<td>1.466+</td>
<td>1.509</td>
</tr>
<tr>
<td>Hudson</td>
<td>1.366</td>
<td>1.366</td>
<td>1.388</td>
<td>1.388-1.395</td>
<td>1.388-1.395</td>
<td>1.388-1.395</td>
<td>1.388-1.395</td>
<td>?</td>
<td>Not supported</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

- 3.7 is broken with 1.9.8 due to a backward incompatibility in TestLink XML-RPC API, please use 3.8 or higher

### Resources

**Can’t show details. Ask your admin to whitelist this Jira URL.**

[View these issues in Jira](viewtheseisssuesinJira)
1. Enabling SSL for Jenkins TestLink plugin in Tomcat
2. Google Test + TAP Listener + Jenkins TestLink Plug-in
7. Lighting talk for Encontro Ágil 2010, at IME-USP.
10. Sponsor company: TupiLabs - Brazil
11. Presentation at STPCon Spring March 2012 in New Orleans - USA.

Sponsors

For commercial support, please get contact us via @tupilabs

Release Notes

Release 3.16 (2019-02-07)
1. JENKINS-48488 - Request to make Testcase version available in Jenkins Build environment

Release 3.15 (2018-12-29)

1. Updated testlink-java-api to 1.9.17-0
2. Updated Jenkins parent in pom.xml
3. Updated dependencies (lang, io, codec)
4. Set project Java to 8
5. Tested with TestLink 1.9.17 (basic workflow, with a TAP file name strategy)

Release 3.14 (2018-03-28)

1. JENKINS-50445 - Add tap4j model objects to the whitelist for serialization to make TAP reporting compatible with Jenkins 2.102+
2. JENKINS-49302 - Fix escaping of summary reports (regression in 3.13)
3. PR #31 - Fix issues in French localization

Release 3.13 (2017-)

1. Pull request #26: make the testcase external ID visible as an environment variable in a Jenkins shell thanks @johnwalker247!
2. Pull request #22: Updateparser to version 0.5 with code fixes thanks @yasassri
3. Fix security issue

Release 3.12 (2016-04-17)

1. Upgraded tap4j(http://tap4j.org)
2. Upgraded testlink-java-api(https://github.com/kinow/testlink-java-api)
3. Pull request #20: Plan and build custom fields support thanks @maiksaray!

Release 3.11 (2015-11-14)

1. TAP test plans support "n..m" with n>1
2. Added build number and error message in notes of JUnit Test Result
3. Update TestLinkSite.java

Release 3.10

1. https://issues.jenkins-ci.org/browse/JENKINS-20599

Release 3.9

1. https://issues.jenkins-ci.org/browse/JENKINS-20587
2. https://issues.jenkins-ci.org/browse/JENKINS-20589

Release 3.8
1. JENKINS-20014: Jenkins Testlink plugin reports "Found 0 automated test cases in TestLink."
2. Reverted JENKINS-17567: Allow the plug-in to filter test cases by last execution status due to a backward incompatibility in TestLink 1.9.8 XML-RPC API
3. Updated tap4j to 4.0.4

Release 3.7
1. Merged pull request https://github.com/jenkinsci/testlink-plugin/pull/8 that adds platforms to the job config

Release 3.6
1. JENKINS-17801: TAP-attachments produce file not found error
2. JENKINS-15790: Name of Test Case is null
3. JENKINS-19209: Testlink Plugin Not Run Test should mark build as failed
4. JENKINS-19390: Testlink Plugin did not manage to get 2nd execution status
5. Updated testlink-java-api to 1.9.7-0

Release 3.5
1. JENKINS-17622: Cannot access TestLink environment variables within Jenkins build
2. JENKINS-16460: Mark in Jenkins build with "NOT RUN" Test Cases and display in the UI (with colours, etc)

Release 3.4
1. JENKINS-17567: Allow the plug-in to filter test cases by last execution status
2. JENKINS-13821: When test uses DataProvider it mark in TestLink only by last result
3. JENKINS-17642: Trouble finding test results using TAP result seeking strategy
4. Quick profiling with Yourkit. Nothing worth of refactoring was found. Thanks to Yourkit for providing an Open Source license to us.

Release 3.3
1. Jenkins-17442: When connectin to TestLink find error ClassCastException (due to a bug in TL XML-RPC API, the plug-in wasn't working with TestLink 1.9.6)
2. Jenkins-16118: Testlink Plug-In: get Custom field information for Testplans and Testprojects
3. Jenkins-17023: Jenkins Testlink plugin Found 0 test result is show while test is runned
4. Jenkins-15588: TestLink is not getting Updated
5. Jenkins-17147: The configuration for testing method name and data provider is not saved

Release 3.2
1. Updated Jenkins version to 1.466
2. The plug-in now should work correctly with other plug-ins that require a BuildStepDescriptor, as Conditional Build Step
3. JENKINS-15486: Documentation is needed for Project GTest Sample Tap&Testlink for C++ testing
4. JENKINS-15343: Unable to use Conditional BuildStep Plugin with Testlink Plugin

Release 3.1.5
JENKINS-10904 - Include test step information as env vars. We had to update testlink-java-api, as the bug was in there. However, the TestLink data stored in builds will be lost. So if you need any of the data, back it up before updating the plug-in.

Release 3.1.2
Added test summary. This way the user can see more details in TestLink reports.

Release 3.1.1
Fixing bugs in JUnit and TestNG strategies

Release 3.1
1. Created Result Seeking Strategy extension point
2. Fixed all blocker issues
3. Added more result seeking strategies (TestNG method name, for example)
4. Added new contributors to the project
5. Code reviewed and refactored several parts of the code, it's way cleaner now
6. Added configuration to make optional attachments upload

Release 3.0.2
Minor improvements for issues found during update of Jenkins TestLink Plug-in Tutorial

Release 3.0.1
JENKINS-11264 - Test execution notes being added twice and incorrectly formatted
Release 3.0

1. JENKINS-10623 - Organize and update French, Spanish and Brazilian Portuguese translation and documentation
2. JENKINS-9054 - Add support to platforms in TestLink plug-in
3. JENKINS-10809 - Add a way to call other build steps in test execution for the plug-in (Yay for DRY!)
4. JENKINS-10849 - OutOfMemoryError using TestLink plugin (Thanks to YourKit!)

Release 2.5

1. JENKINS-9811 Add a POST <Single test command>-field, to execute a process after the plug-in iterates the retrieved automated test cases
2. JENKINS-9672 Test link custom field parser split the String value by semicolon or comma.
3. JENKINS-9993 Add root element for JUnit test results

Release 2.2.2

1. FIXED-9444 - Add environment variables to single test command (actually, I used some code from Jenkins core to execute both commands now :-)

Release 2.2.1

1. French translation
2. Small issue with non-existent option Debug in Job configuration
3. FIXED-9229 - JUnit wrong status
4. Portuguese i18n messages typo

Release 2.2

1. JUnit BUG fix
2. Enhancement of the Build logs

Release 2.1

1. Migration from Hudson to Jenkins
2. Test transaction property.
3. i18n (version 1.0 supports only English).
4. Add Javascript validation functions to each field in the global and config pages.
5. Add single test command feature (it will enable running test suites)
6. Fixed Java class headers
7. JFreechart graphics betterment
8. Fixed BUG JENKINS-8636 TestLink Plugin FATAL: Error creating test project
9. Fixed BUG JENKINS-8531 Unexpressive error message

Release 2.0.1

1. Fixed BUG 8292 (java.lang.NullPointerException at hudson.plugins.testlink.updater.TestLinkTestStatusUpdater.updateTestCases (TestLinkTestStatusUpdater.java:55)

Release 2.0

1. Settings automated tests properties as environment variables.
3. JUnit parser.
4. TestNG parser.

Release 1.1

1. Execute Test Suites.
2. Let the user define the custom fields name.
3. Implement feature that lets user to use the latest revision from a SVN repository as Build name.

Release 1.0

1. First version of the plug-in (the development was guided based on the source code of the following plug-ins: CCM, Sonar (how to ref maven installations)).

Roadmap

1. Keep compatibility with latest versions of Jenkins and TestLink.
2. Add test case steps (JENKINS-10904)

Open source licenses donated for this project
YourKit is kindly supporting open source projects with its full-featured Java Profiler. YourKit, LLC is the creator of innovative and intelligent tools for profiling Java and .NET applications. Take a look at YourKit’s leading software products: 
YourKit Java Profiler and 
YourKit .NET Profiler.