Groovy plugin

Plugin Information
View Groovy on the plugin site for more information.

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

- Arbitrary code execution vulnerability
- Script Security sandbox bypass
- Script security sandbox bypass

This plugin adds the ability to directly execute Groovy code.

Configuration

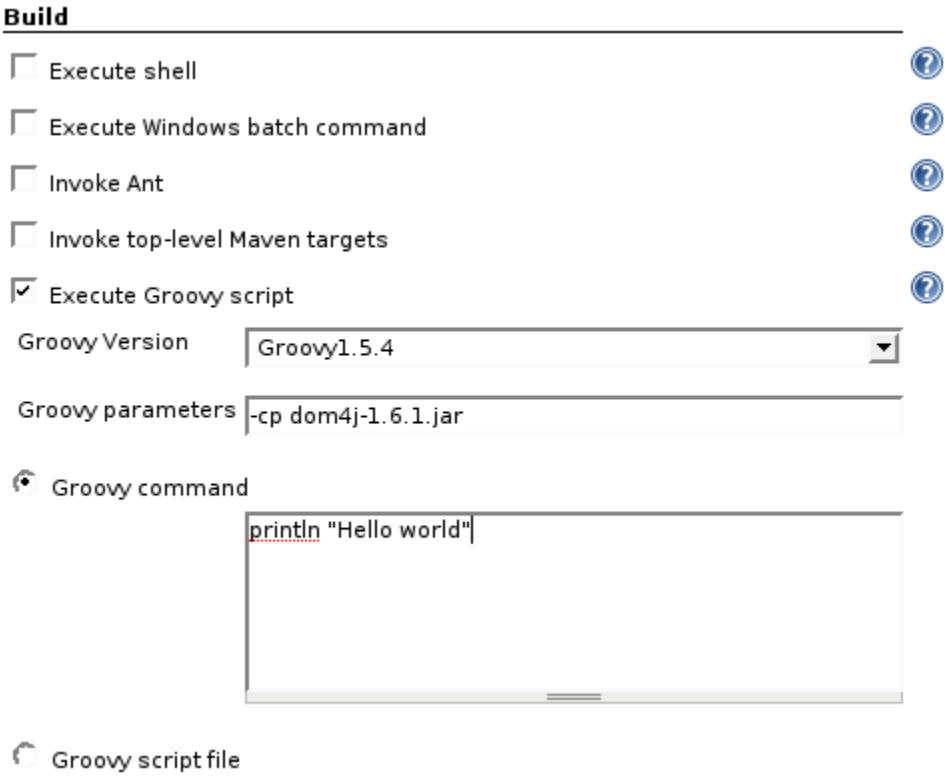
To configure available Groovy installation on your system, go to Jenkins configuration page, find section 'Groovy' and fill the form as shown bellow.

If you don't configure any Groovy installation and select `(Default)` option in a job, the plugin will fallback into calling just the `groovy` command, assuming you have `groovy` binary on the default path on given machine.

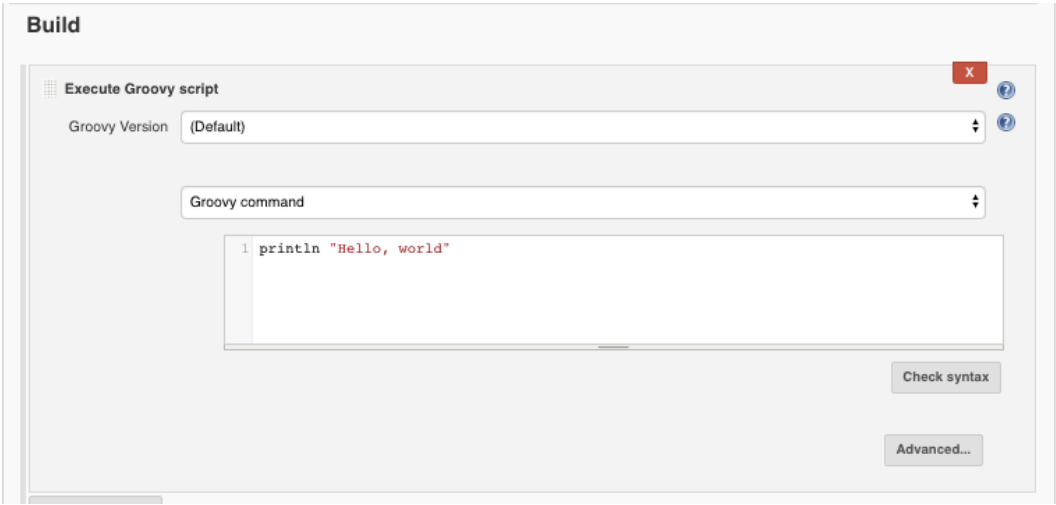
Usage

To create Groovy-based project, add new free-style project and select "Execute Groovy script" in the Build section, select previously configured Groovy installation and then type your command, or specify your script file name. In the second case path taken is relatively from the project workspace directory.

Jenkins 1.x:

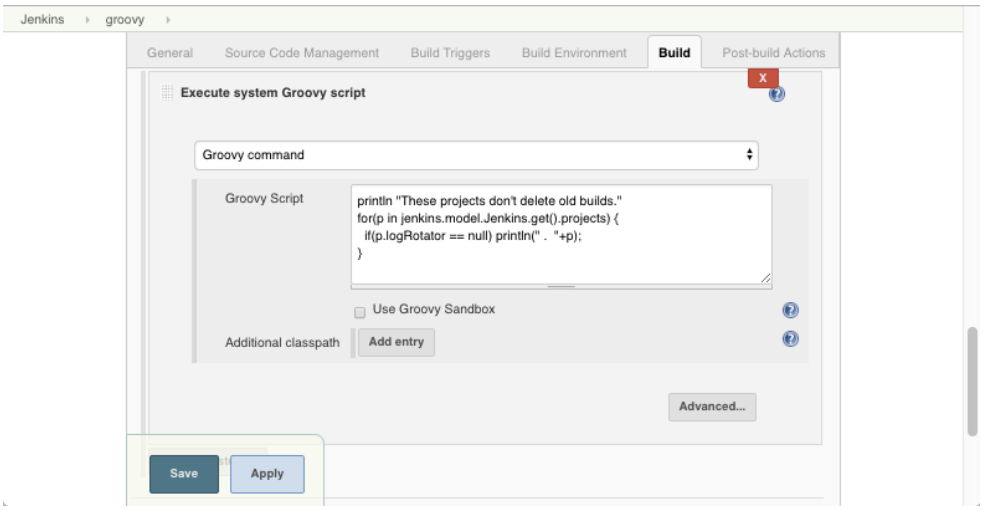


Jenkins 2.x:

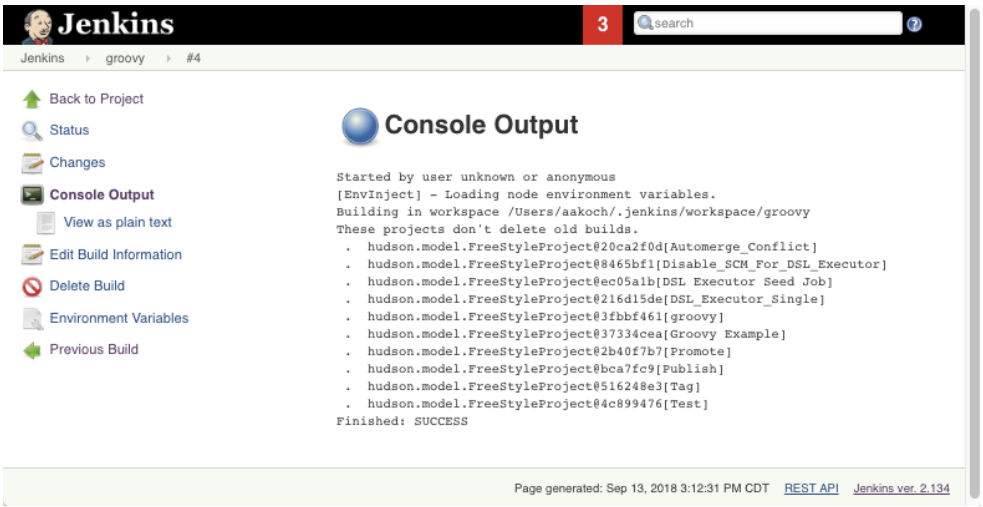


The plugin also adds the functionality of the Script Console to the project configuration page.

You can schedule your system management script...



...and then observe progress in the build log.



Groovy Script vs System Groovy Script

The plain "Groovy Script" is run in a forked JVM, on the slave where the build is run. It's the basically the same as running the "groovy" command and pass in the script.

The system Groovy script on the other hand runs inside the Jenkins master's JVM. Thus it will have access to all the internal objects of Jenkins, so you can use this to alter the state of Jenkins. It is similar to the Jenkins Script Console functionality.

Security

System groovy jobs has access to whole Jenkins, therefore only users with admin rights can add system Groovy build step and configure the system Groovy script. Permissions are not checked when the build is triggered (i.e. also uses without admin rights can also run the script). The idea is to allow users run some well defined (defined by admin) system tasks when they need it (e.g. put slave offline/online, when user wants to start some debugging on slave). To have Jenkins instance secure, the support for Token macro plugin has to be switched off, see section below.

Token macro plugin support

Groovy plugin provides support for Token Macro Plugin. Expression is *GROOVY* with parameter *script*:

```
${GROOVY,script = "return hudson.model.Hudson.instance.pluginManager.plugins"}
```

By default, the support for token macro pressing is switched off and has to be switch on in global config page.

If token macro processing via Token Macro Plugin is allowed, the evaluation of macro is done in System Groovy, therefore any user can run arbitrary system script, regardless he has administer permission!

Examples

Retrieving parameters and triggering another build

Execute a system Groovy script like:

```
import hudson.model.*
import hudson.AbortException
import hudson.console.HyperlinkNote
import java.util.concurrent.CancellationException

// Retrieve parameters of the current build
def foo = build.buildVariableResolver.resolve("FOO")
println "FOO=$foo"

// Start another job
def job = Hudson.instance.getJob('MyJobName')
def anotherBuild
try {
    def params = [
      new StringParameterValue('FOO', foo),
    ]
    def future = job.scheduleBuild2(0, new Cause.UpstreamCause(build), new ParametersAction(params))
    println "Waiting for the completion of " + HyperlinkNote.encodeTo('/' + job.url, job.fullDisplayName)
    anotherBuild = future.get()
} catch (CancellationException x) {
    throw new AbortException("${job.fullDisplayName} aborted.")
}
println HyperlinkNote.encodeTo('/' + anotherBuild.url, anotherBuild.fullDisplayName) + " completed. Result was
" + anotherBuild.result

// Check that it succeeded
build.result = anotherBuild.result
if (anotherBuild.result != Result.SUCCESS && anotherBuild.result != Result.UNSTABLE) {
    // We abort this build right here and now.
    throw new AbortException("${anotherBuild.fullDisplayName} failed.")
}

// Do something with the output.
// On the contrary to Parameterized Trigger Plugin, you may now do something from that other build instance.
// Like the parsing the build log (see http://javadoc.jenkins-ci.org/hudson/model/FreeStyleBuild.html )
// You probably may also wish to update the current job's environment.
build.addAction(new ParametersAction(new StringParameterValue('BAR', '3')))
```

Retrieve properties

To retrieve properties defined in the Properties field use:

```
System.getProperty('FOO')
```

Usage with pipeline

Currently the plugin does not support pipeline syntax. One workaround from Unknown User (alexander_samoylov) was mentioned here: https://stackoverflow.com/a/58381147/4807875.

Changelog

Release 2.2 (2019-03-06)

- Fix security issue

Release 2.1 (2019-01-28)

- Fix security issue

Release 2.0 (2017-04-10)

- Arbitrary code execution by unprivileged user (SECURITY-292)
- continue with code cleanup - fixed Findbugs issues

Release 1.30 (2016-11-18)

- XSS protection
- code cleanup

Release 1.28, 1.29 (2016-01-05)

- code cleanup

Release 1.27 (2015-08-05)

- `Callable` roles are properly checked

Release 1.26 (2015-07-27)

- Ensured correct position of class path option (JENKINS-29577)
- Improved help (pr #18)

Release 1.25 (2015-05-11)

- Made default choice also for System Groovy script to avoid zero height of textarea (JENKINS-25455)
- Add help file for Groovy version (JENKINS-12988)
- Made setting Groovy installations thread-safe (JENKINS-28287)

Release 1.24 (2014-11-09)

- Ensure non-zero height of Groovy command text box, making it default choice when adding new build step (JENKINS-25455)

Release 1.23 (2014-10-27)

- Set up correct GROOVY_HOME environment variable (JENKINS-25275)

Release 1.22 (2014-09-30)

- Fixed slashes conversion in script parameters (JENKINS-24870)

Release 1.21 (2014-09-18)

- Allow spaces in script parameters (JENKINS-24757)

Release 1.20 (2014-07-30)

- Unable to specify multiple jars on class path for a system groovy script (JENKINS-23997)

Release 1.19 (2014-07-07)

- Better parsing of parameters passed to Groovy binary, Apache commons-exec used for parsing (JENKINS-23617)

Release 1.18 (2014-05-13)

- NPE fixes (JENKINS-17171)

Release 1.17 (2014-05-09)

- Allow whitespaces in properties (passed via -D switch) (pull13)

Release 1.16 (2014-04-07)

- Upgrade to @DataBoundConstructor (JENKINS-6797)
- Fixed typo in warrning message (pull12)

Release 1.15 (2014-01-31)

- Syntax highlighting
- Syntax validation button
- Prepare for Jenkins core upgrade to Groovy 2.x (pull9)

Release 1.14 (2013-07-02)

- Right to run the System Groovy script changed from ADMINISTER to RUN_SCRIPTS (pull7)

Release 1.13 (2013-03-01)

- Added build context (build, launcher, listener) into system groovy build step (pull6)

Release 1.12 (2012-03-08)

- Fixed configuration of Token macro (pull5)

Release 1.11 (2012-02-26)

- Enabled env. variables expansion class path, groovy and script parameters

Release 1.10 (2012-02-09)

- Fixed possible job configuration corruption when user isn't admin (JENKINS-12080)
- Avoid NPE, add fallback if groovy executable is misoncifured (JENKINS-11652)

Release 1.9 (2011-09-14)

- Auto installer (JENKINS-7113 and JENKINS-10920)
- Fixed error message on global config page (JENKINS-10768)
- Expansion of job parameters (JENKINS-10525)
- Full access to JAVA_OPTS (i.e. parameters like -Xmx can be set up)
- Editable class path

Release 1.8 (2011-05-13)

- Fixed a configuration persistence problem that can create huge config.xml

Release 1.7 (2011-03-09)

- Added support for Token Macro Plugin

Release 1.6 (2011-02-08)

- Fixed security issue

Release 1.5 (2010-11-10)

- Classloader for actual System Groovy (JENKINS-6068)
- Allowed groovy.bat in addition to groovy.exe (JENKINS-6839)
- Temp files are removed (JENKINS-3269)
- Hudson global properties are expanded in groovy script file path (JENKINS-8048)
- Upgraded to 1.358 (JENKINS-6081)

Release 1.4 (2009-12-29)

- Improve error message for missing groovy executable
- Update uses of deprecated APIs

Release 1.2

- Added possibility to specify properties and script parameters.
- Added script source choice (file/command) for system groovy scripts.
- Used .exe instead of .bat on Windows (as suggested by Scott Armit).
- Added configuration option for classpath and initial variable bindings for system groovy scripts.

Known bugs

- Configuring more builders at once actually doesn't absolutely work. If you need more groovy builders in your project, you have to configure them one by one and always save project configuration before you add new one.