

# Slack Plugin

## Plugin Information

View Slack Notification on [the plugin site](#) for more information.

**This plugin is up for adoption.** Want to help improve this plugin? [Click here to learn more!](#)

This plugin allows to post build notifications to a [Slack](#) channel.

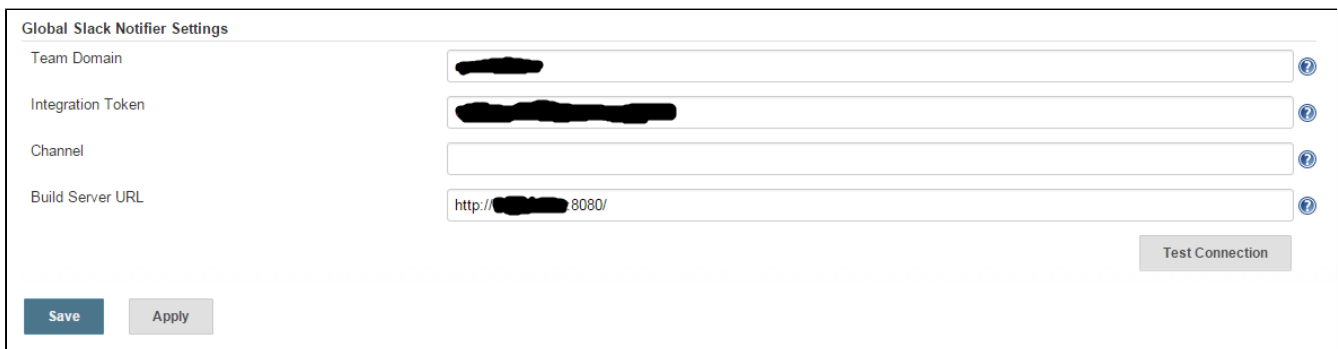
## Documentation

The instructions presented below are incomplete. Follow the instructions on the [Slack Plugin's Github home page](#). They work great.

The Slack plugin allows for configuring a default Slack channel, or a Slack channel may be defined on a per build task basis. There are several things that need to be done to get the Slack plugin working. The projects home page is on [Github](#).

## Global configuration

1. Configure the "Jenkins", "Manage Jenkins", "Configure System", "Global Slack Notifier Settings" values. Scroll to the bottom of the screen or search for the word "Slack" to find the entry.
2. Enter the Team domain, Integration Token, and Build Server URL into the indicated text fields.
3. If you have a default channel name (e.g. #builds) enter this name into the "Channel" text field.
4. If you have entered a default channel name, click the "Test Connection" button.
5. Verify a test message is displayed in the default Slack channel.



The screenshot shows the "Global Slack Notifier Settings" configuration page. It contains four text input fields: "Team Domain", "Integration Token", "Channel", and "Build Server URL". The "Build Server URL" field has a default value of "http://[redacted]:8080/". There are help icons (question marks) to the right of each field. At the bottom right is a "Test Connection" button. At the bottom left are "Save" and "Apply" buttons.

## Project Configuration

1. Configure the "Jenkins", <project>, "Configure", "Slack Notifications" values.
2. Select the check boxes for the notifications desired when the build task is executed.
3. If the project has a different channel from the default perform the following steps.
  1. Click on the "Advanced..." button.
  2. Enter the "Team Domain", "Integration Token" (probably the same as the Global token), and "Project Channel" name in the indicated text fields. Do not rely on the defaults to populate any of the fields. Enter the information even if it is redundant with the Global configuration.
4. Click on the "Test Connection" button.
5. Verify a test message was sent to the appropriate Slack channel.
6. Click on the "Save" button to retain these changes.

**Slack Notifications**

Notify Build Start	<input checked="" type="checkbox"/>
Notify Aborted	<input checked="" type="checkbox"/>
Notify Failure	<input checked="" type="checkbox"/>
Notify Not Built	<input type="checkbox"/>
Notify Success	<input checked="" type="checkbox"/>
Notify Unstable	<input type="checkbox"/>
Notify Back To Normal	<input checked="" type="checkbox"/>
Notify Repeated Failure	<input checked="" type="checkbox"/>
Include Test Summary	<input type="checkbox"/>
<input type="checkbox"/> Include Custom Message	
Show Commit List with Titles and Authors	<input type="checkbox"/>
Team Domain	<input type="text" value="████████████████████"/>
Integration Token	<input type="text" value="██"/>
Project Channel	<input type="text" value="████████"/>

## Post-build Actions

If you forget this section, the test buttons will show the channel test messages, but the build execution will not post anything to the channel.

1. Scroll to the "Post-build Actions" section of the project configuration.
2. Click on the "Add post-build action" button.
3. Select the "Slack Notifications" entry from the list displayed. The "Slack Notifications" section will appear in the window.
4. Click on the "Save" button to retain these changes.

Send build artifacts over FTP

Set build status on GitHub commit

Slack Notifications

**Slack Notifications**

## Jenkins Pipeline Support

```
slackSend channel: '#jenkins-latest', color: 'good', message: 'Slack
Message', teamDomain: 'beedemo', token: 'token'
```

## Test the configuration

Once the project has been configured run a test build and confirm the Slack messages are posted as expected.

## Configure plugin via Groovy script

Either automatically upon [Jenkins post-initialization](#) or through [Jenkins script console](#), example:

```
import com.cloudbees.jenkins.plugins.sshcredentials.impl.*
import com.cloudbees.plugins.credentials.*
import com.cloudbees.plugins.credentials.common.*
import com.cloudbees.plugins.credentials.domains.Domain
import com.cloudbees.plugins.credentials.impl.*
import hudson.util.Secret
import java.nio.file.Files
import jenkins.model.Jenkins
import net.sf.json.JSONObject
import org.jenkinsci.plugins.plaincredentials.impl.*

// parameters
def slackCredentialParameters = [
    description: 'Slack Jenkins integration token',
    id:          'slack-token',
    secret:      '1234567890asdfghjklqwert'
]

def slackParameters = [
    slackBaseUrl:          'https://mycompany.slack.com/services/hooks
/jenkins-ci/',
    slackBotUser:          'true',
    slackBuildServerUrl:   'https://jenkins.mycompany.com:8083/',
    slackRoom:             '#jenkins',
    slackSendAs:           'Jenkins',
    slackTeamDomain:       'mycompany',
    slackToken:             '',
    slackTokenCredentialId: 'slack-token'
]

// get Jenkins instance
Jenkins jenkins = Jenkins.getInstance()

// get credentials domain
def domain = Domain.global()

// get credentials store
def store = jenkins.getExtensionList('com.cloudbees.plugins.credentials.
SystemCredentialsProvider')[0].getStore()

// get Slack plugin
```

```
def slack = jenkins.getExtensionList(jenkins.plugins.slack.
SlackNotifier.DescriptorImpl.class)[0]

// define secret
def secretText = new StringCredentialsImpl(
    CredentialsScope.GLOBAL,
    slackCredentialParameters.id,
    slackCredentialParameters.description,
    Secret.fromString(slackCredentialParameters.secret)
)

// define form and request
JSONObject formData = ['slack': ['tokenCredentialId': 'slack-token']]
as JSONObject
def request = [getParameter: { name -> slackParameters[name] }] as org.
kohsuke.stapler.StaplerRequest

// add credential to Jenkins credentials store
store.addCredentials(domain, secretText)

// add Slack configuration to Jenkins
slack.configure(request, formData)

// save to disk
slack.save()
jenkins.save()
```

## Changelog

Also on [Github](#)