

Setting up a Jenkins-Server and creating a Jenkins-Maven-Pipeline for Tomcat 6.0.53

Installing Jenkins

The following instructions are based on this site: <https://pkg.jenkins.io/debian-stable/>

1. To use the Debian package repository of Jenkins add it's key to the system by executing the following command via the terminal: `wget -q -O https://pkg.jenkins.io/debian-stable/jenkins.io.key | sudo apt-key add -`
2. Add the following line to the file "sources.list" which can be found in /etc/apt/: `deb https://pkg.jenkins.io/debian-stable binary/`
3. Run the command "sudo apt-get update" and then the command "sudo apt-get install jenkins" (in that order)

You should be able to access Jenkins in a web browser now.

Getting started

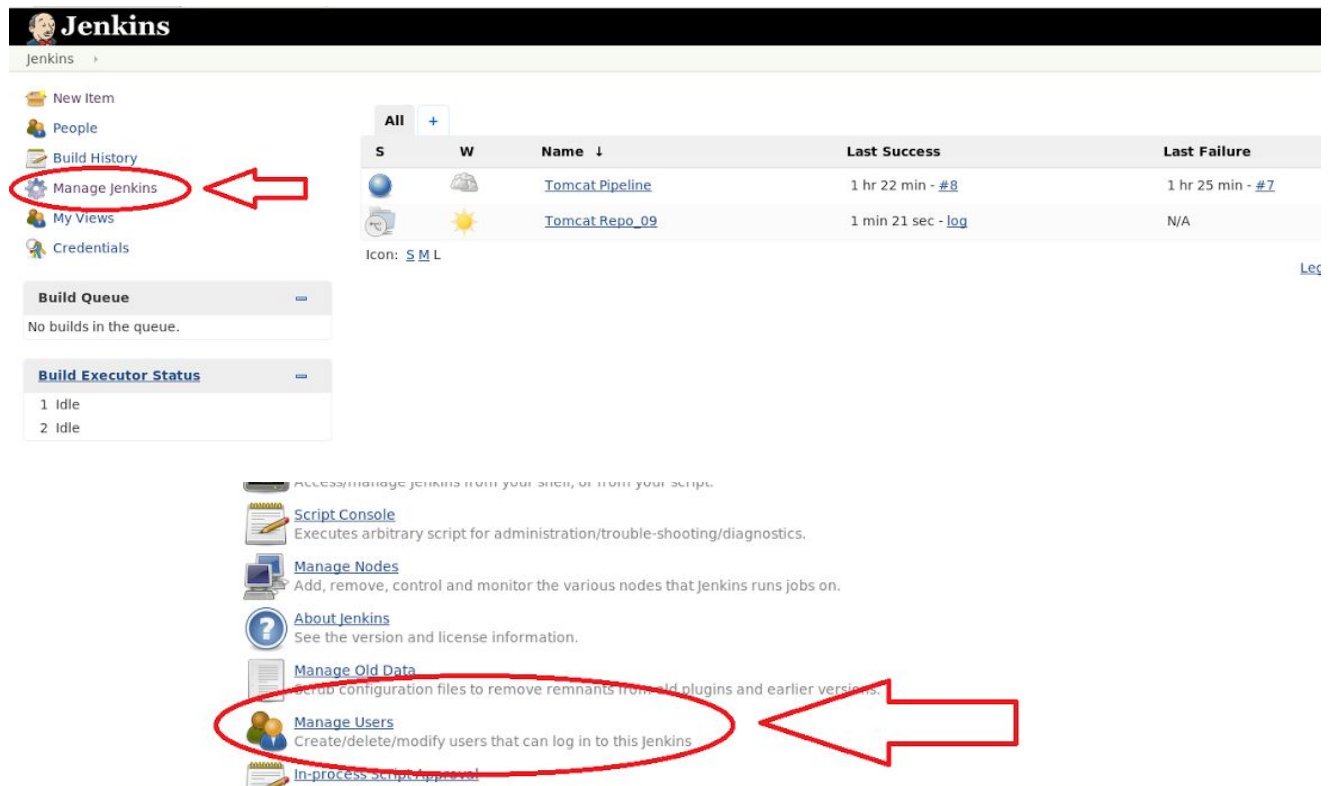
1. Open a web browser and type in localhost:8080
2. On your system navigate to /var/log/jenkins/ and copy the initial password from jenkins.log
3. Paste the password into the field in the web browser
4. Select plugins to install. Make sure the plugins "Pipeline" and "GitHub" are selected (most of the plugins should already be selected by default)

Creating A User

When you first login to Jenkins you use the initial password found in jenkins.log. To avoid having to type in the same long password again the next time you login, you should create another user.

In Jenkins:


1. Go to “Manage Jenkins” > “Manage Users”



The screenshot shows the Jenkins dashboard. On the left sidebar, the 'Manage Jenkins' link is circled in red with a red arrow pointing to it. Below the dashboard, a list of links is shown, with 'Manage Users' circled in red and a red arrow pointing to it.

S	W	Name ↓	Last Success	Last Failure
		Tomcat Pipeline	1 hr 22 min - #8	1 hr 25 min - #7
		Tomcat Repo_09	1 min 21 sec - log	N/A

2. Fill in the input fields
3. Click “Create User”



The screenshot shows the Jenkins 'Create User' form. The form has fields for Username, Password, Confirm password, Full name, and E-mail address. A 'Create User' button is at the bottom.

Create User

Username:

Password:

Confirm password:

Full name:

E-mail address:

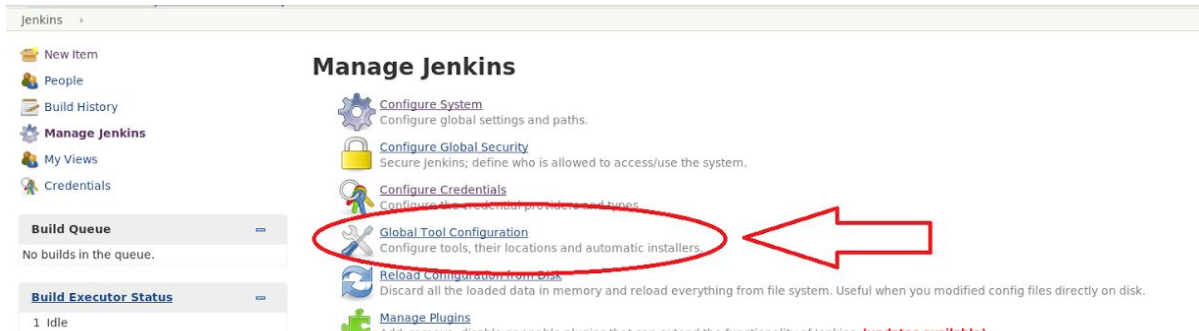
[Create User](#)

Configure Maven-Tool and JDK

Maven does not come with Jenkins by default. Note that the following steps are not the only solution.

In Jenkins:

Got to “Manage Jenkins” > “Global Tool Configuration”



1. Maven
 - a. Scroll down to section “Maven”
 - b. Click on “Add Maven”
 - c. Type in name for Maven tool and select “Install automatically” and version.
 - d. Click “Apply” and “Save”
2. JDK
 - a. Scroll down to section “JDK”
 - b. Click on “Add JDK”
 - c. Type in name for JDK and select “Install automatically” and version.
 - d. Click “Apply” and “Save”

Creating Jenkinsfile for Tomcat 6.0.53

This is the heart of this exercise. The Jenkinsfile is one way of defining a Jenkins Pipeline. Here you define different stages through which your project will be built, tested and deployed. The Jenkinsfile has to be located in your project’s repository (the root directory should do).

In GitHub-Repository:

1. Create Jenkinsfile
2. Use the following structure:

```

pipeline {
  agent any

  triggers {
    pollSCM('*/*5 * * * *')
  }

  tools{
    maven 'apache-maven-3.5.2'
    jdk 'java-sdk-1.8'
  }
  stages {
    stage('Build') {
      steps {
        sh 'cd Tomcat && mvn clean compile'
      }
    }
    stage('Test'){
      steps {
        sh 'cd Tomcat && mvn test'
      }
    }
    stage('Deploy') {
      steps {
        sh 'cd Tomcat && mvn assembly:single'
      }
      post {
        success {
          archiveArtifacts(artifacts: 'Tomcat/target/*.jar', allowEmptyArchive: true)
        }
      }
    }
  }
}

```

“pipeline” - All valid Declarative Pipelines must be enclosed within a “pipeline” block.

“agent” - The agent section specifies where the entire Pipeline, or a specific stage will execute in the Jenkins environment depending on where the agent section is placed.

“Triggers” - Here one can specify a trigger for polling. The value for pollSCM stands for a poll for changes every 5 minutes.

“tools” - This is a section for the defining of tools to auto-install and put on the PATH. Note that these tools must be pre-configured manually in “Manage Jenkins” -> “Global Tool Configuration”

“stages”/“stage” - All the relevant work done by a Pipeline will be wrapped in one or more stage directives

“steps” - Steps are wrapped in stages and they take a block of Scripted Pipeline.

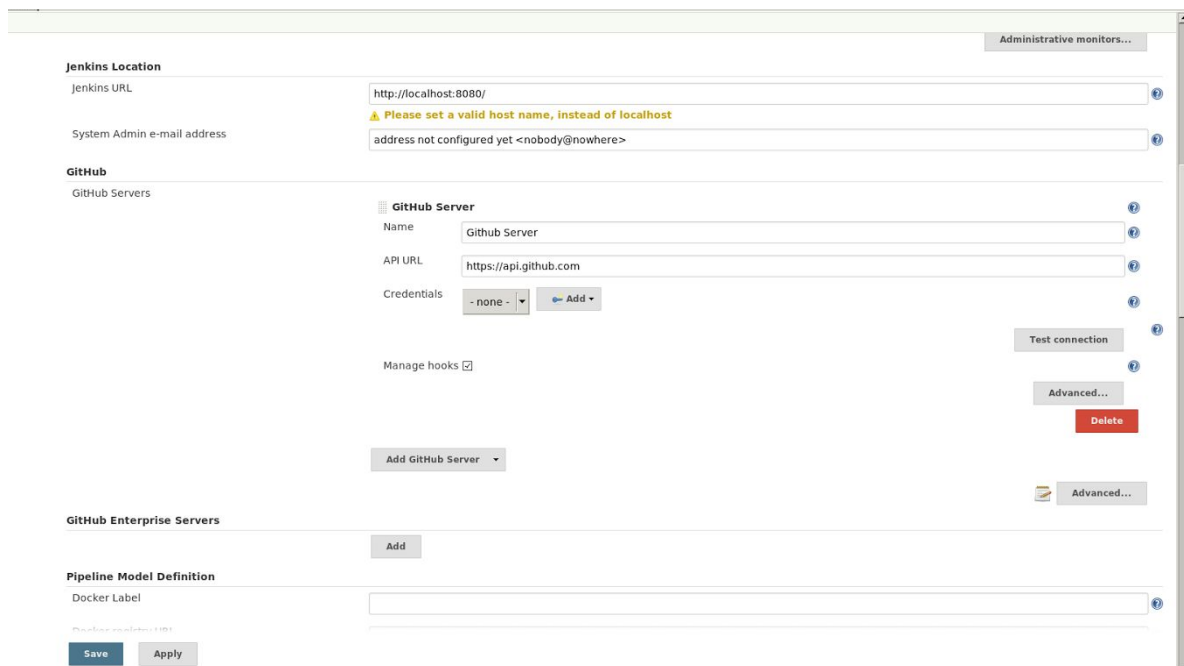
“post” - Defines action which will be run at the end of the Pipeline run or stage

Configuring GitHub for Jenkins

Got to “Manage Jenkins” > “Configure System”



1. Scroll down to section “GitHub” and click “Add GitHub Server”
2. Enter a Name and add credentials
 - a. Make sure to set “Kind” to “Secret” Text. Here you should paste a token which you can generate here: <https://github.com/settings/tokens> into the field “Secret”

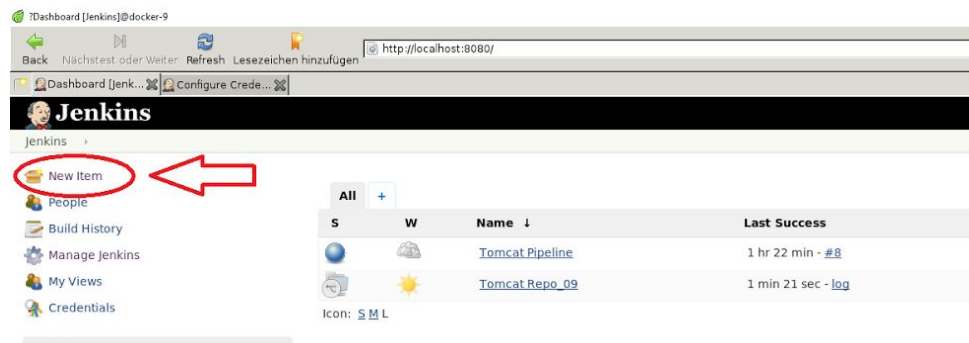


3. Click “Apply” and “Save”

Creating a Jenkins Pipeline for Tomcat 6.0.53 via Maven

In Jenkins:

1. Go to “New Item”



The screenshot shows the 'Enter an item name' form in Jenkins. The 'Pipeline' option is selected under the 'Freestyle project' section. The form includes a text input field for the item name, which contains the text 'Pipeline'. Below the input field, there is a list of item types with their descriptions:

- Freestyle project**: This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system, and this can be even used for something other than software build.
- Pipeline**: Orchestrates long-running activities that can span multiple build slaves. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.
- Multi-configuration project**: Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.
- Folder**: Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates a separate namespace, so you can have multiple things of the same name as long as they are in different folders.
- GitHub Organization**: Scans a GitHub organization (or user account) for all repositories matching some defined markers.
- Multibranch Pipeline**: Creates a set of Pipeline projects according to detected branches in one SCM repository.

At the bottom, there is a section for 'If you want to create a new item from other existing, you can use this option:' with a 'Copy from' dropdown menu and an 'OK' button.

2. Click on “Multibranch Pipeline”, enter a pipeline name and click on “OK”
3. Under section “Branch Sources” click “Add” Source -> “GitHub”
4. Add Credentials with “Kind” set to “Username and Password”
5. Type in name of the repository owner and chose the right repository

GitHub

Credentials: Sonnyboy3428/***** Add

Owner: lsd-lecture

Repository: repo-09

Behaviors

Discover branches

Strategy: All branches Delete

Discover pull requests from origin

Strategy: Merging the pull request with the current target branch revision Delete

Discover pull requests from forks

Strategy: Merging the pull request with the current target branch revision

Trust: Contributors Delete

Add

Property strategy: All branches get the same properties

Save Apply

6. Click “Apply” and “Save”

Building a Jenkins Job

If the Jenkins Job was configured correctly you should witness the first build. You can monitor the success of your builds on the dashboard (starting page) or by clicking on the builds you would like to have a closer look at.

Jenkins jenkins Test log out

New Item People Build History Manage Jenkins My Views Credentials

Build Queue No builds in the queue.

Build Executor Status 1 Idle 2 Idle

All	S	W	Name	Last Success	Last Failure	Last Duration
			Tomcat Pipeline	1 hr 22 min - #8	1 hr 25 min - #7	1 min 4 sec
			Tomcat Repo_09	1 min 21 sec - log	N/A	2,2 sec

Icon: [S](#) [M](#) [L](#)

Legend RSS for all RSS for failures RSS for just latest builds

Jenkins jenkins Tomcat Repo_09 master Test log out

Up Status Changes Build Now View Configuration Full Stage View GitHub Pipeline Syntax

Branch master

Full project name: Tomcat Repo_09/master

[Last Successful Artifacts](#) lsd-app-1.0-SNAPSHOT-jar-with-dependencies.jar 6.81 MB View

[Recent Changes](#)

Stage View

	Declarative: Checkout SCM	Declarative: Tool Install	Build	Test	Deploy
Average stage times: (Average full run time: ~1min 3s)	3s	62ms	25s	4s	32s
#8 Nov 15, 2017 10:35 PM	1s	55ms	27s	5s	32s
#7 Nov 15, 2017 10:31 PM	2s	62ms	27s	4s	31s
#6 Nov 15, 2017 10:29 PM					
#5 Nov 15, 2017 10:24 PM					
#4 Nov 15, 2017 10:21 PM					
#3 Nov 15, 2017 10:16 PM					
#2 Nov 15, 2017 9:42 PM					