



Hello Mr. Jenkins

Michał Mróz - mmroz122@gmail.com



What is Jenkins?

- Open source test automation server written in Java in 2004 as Hudson at Sun Microsystems.
- Became a fork of Hudson and changed name in 2011 to Jenkins.
- Builds and tests software and monitors status of builds.
- Over 1000+ plugins.
- Integrates with many tools: version control, reporting, testing tools, bug tracking.



Continuous Integration

- CI is a development practice that requires developers to integrate code into a shared repository several times a day. Each check-in is then verified by an automated build, allowing teams to detect problems early.
- By integrating regularly, you can detect errors quickly, and locate them more easily.



Installation

- Available for FreeBSD, OpenBSD, Ubuntu, Debian, Windows and more...
- To install (FreeBSD) ``pkg install jenkins``.

Getting Started

Unlock Jenkins

To ensure Jenkins is securely set up by the administrator, a password has been written to the log ([not sure where to find it?](#)) and this file on the server:

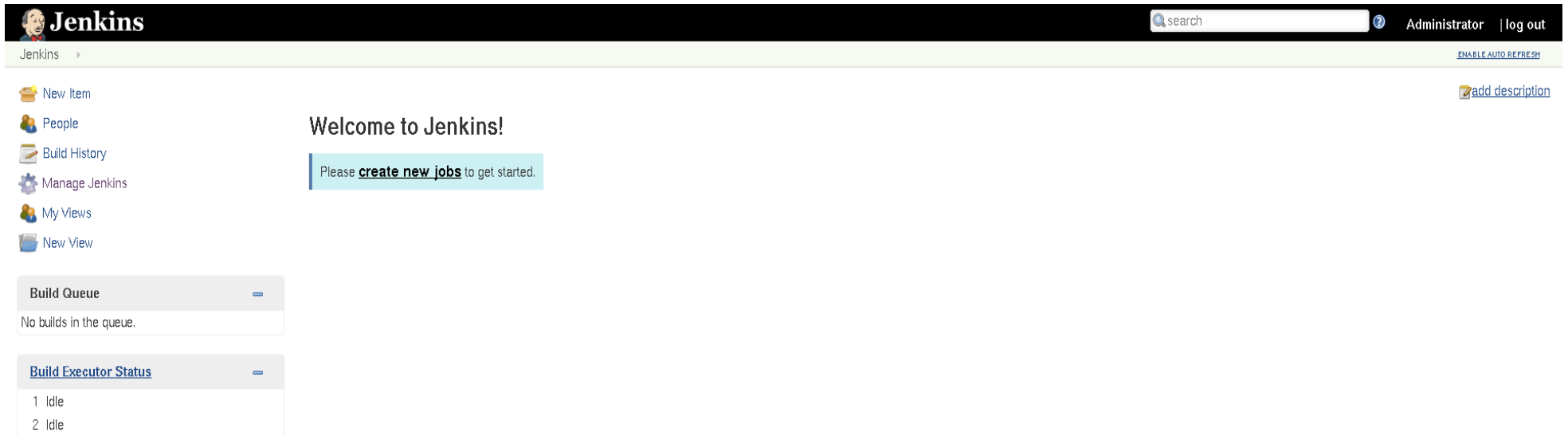
```
/usr/local/jenkins/secrets/initialAdminPassword
```

Please copy the password from either location and paste it below.

Administrator password



Installation

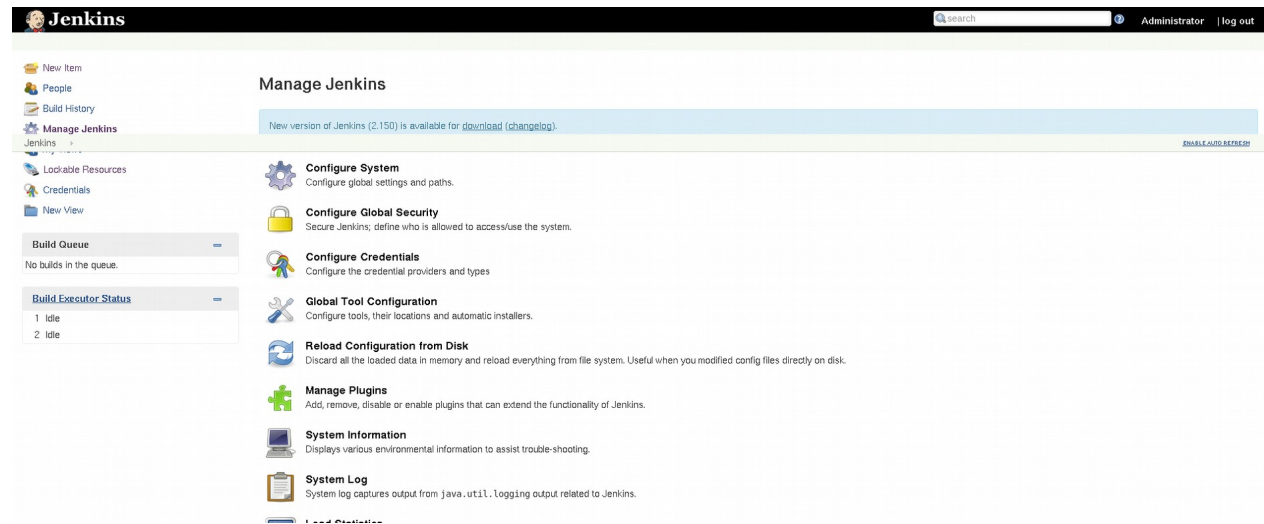


The screenshot shows the Jenkins web interface. At the top left is the Jenkins logo and name. To the right is a search bar and user information: 'Administrator | log out'. Below the search bar is a link for 'ENABLE AUTO REFRESH'. On the right side, there is a link for 'add description'. The main content area features a 'Welcome to Jenkins!' message with a light blue box containing the text: 'Please [create new jobs](#) to get started.' On the left side, there is a navigation menu with items: 'New Item', 'People', 'Build History', 'Manage Jenkins', 'My Views', and 'New View'. Below the menu are two panels: 'Build Queue' showing 'No builds in the queue.' and 'Build Executor Status' showing '1 Idle' and '2 Idle'.



Manage Jenkins

- Configuring System – plugin settings, notifications, workspaces, environment variables.
- Users, security settings.
- System logs.
- Nodes.
- Credentials.
- Plugins.



The screenshot shows the Jenkins web interface. At the top, there's a search bar and user information (Administrator | log out). The main content area is titled "Manage Jenkins" and features a notification: "New version of Jenkins (2.150) is available for download (changelog)." Below this, there are several configuration options, each with an icon and a brief description:

- Configure System**: Configure global settings and paths.
- Configure Global Security**: Secure Jenkins; define who is allowed to access/use the system.
- Configure Credentials**: Configure the credential providers and types.
- Global Tool Configuration**: Configure tools, their locations and automatic installers.
- Reload Configuration from Disk**: Discard all the loaded data in memory and reload everything from file system. Useful when you modified config files directly on disk.
- Manage Plugins**: Add, remove, disable or enable plugins that can extend the functionality of Jenkins.
- System Information**: Displays various environmental information to assist trouble-shooting.
- System Log**: System log captures output from java.util logging output related to Jenkins.

On the left side of the interface, there's a sidebar with navigation links: New Item, People, Build History, Manage Jenkins, Lockable Resources, Credentials, New View, Build Queue (No builds in the queue), and Build Executor Status (1 Idle, 2 Idle).



Plugins

- Support for many version control systems: Perforce, Subversion, Git, Mercurial, CVS...
- Bug tracking – JIRA, Redmine...
- Reporting – XML, HTML...
- Notifications – Slack, Gmail, Discord...
- Dashboard plugins to edit views.



Managing plugins

- All plugins are available from Manage Plugins view where you can update and install them.

Filter:

Updates **Available** Installed Advanced

Install	Name	Version
.NET Development		
<input type="checkbox"/>	CCM This plug-in collects the CCM analysis results of the project modules and visualizes the found warnings.	3.2
<input type="checkbox"/>	change-assembly-version-plugin	1.10
<input type="checkbox"/>	ExCop Runner FxCopCmd.exe support plugin.	1.1
<input type="checkbox"/>	MSBuild This plugin makes it possible to build a Visual Studio project (.proj) and solution files (.sln).	1.29
<input type="checkbox"/>	MSTest This plugin converts MSTest TRX test reports into JUnit XML reports so it can be integrated with Hudson's JUnit features.	0.23
<input type="checkbox"/>	MSTestRunner This plugin run MSTest command line tool to execute unit tests for .NET projects.	1.3.0
<input type="checkbox"/>	NAnt This plugin is a NAnt Builder for building .NET projects.	1.4.3
<input type="checkbox"/>	PowerShell This plugin allows Jenkins to invoke Windows PowerShell as build scripts.	1.3





Creating jobs


- Jobs – executing shell locally/remote.
- Pipelines – multiple step build.
- GitHub Organization – build on repository.


Enter an item name


» This field cannot be empty, please enter a valid name


 **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 agents. 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 Git-Hub 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.

If you want to create a new item from other existing, you can use this option:

 Copy from



Job

General Source Code Management Build Triggers Build Environment Build Post-build Actions

Description

[Plain text] [Preview](#)

- Discard old builds ?
- GitHub project
- This build requires lockable resources
- This project is parameterized ?
- Throttle builds ?
- Disable this project ?
- Execute concurrent builds if necessary ?

Source Code Management

- None
- Git

[Advanced...](#)



Pipeline

- Write in Groovy.
- Jenkinsfile.
- Declare stages and steps.
- Parallel steps.
- Running other jobs...

```
pipeline {
  agent any

  stages {
    stage('Build') {
      steps {
        echo 'Building..'
      }
    }
    stage('Test') {
      steps {
        echo 'Testing..'
      }
    }
    stage('Deploy') {
      steps {
        echo 'Deploying....'
      }
    }
  }
}
```



Pipeline

- Back to Dashboard
- Status
- Changes
- Build Now
- Delete Pipeline
- Configure
- Full Stage View
- Rename
- Pipeline Syntax

Build History [trend](#)

- #3** Nov 12, 2018 1:26 PM
- #2** Nov 12, 2018 1:11 PM
- #1** Nov 12, 2018 12:22 PM

[RSS for all](#) [RSS for failures](#)

Pipeline test-pipeline



Stage View

Average stage times:
(Average full run time: ~2s)

	build	tests	deploy	reports
#3 Nov 12 13:26 No Changes	444ms	350ms	345ms	348ms
#2 Nov 12 13:11 No Changes				
#1 Nov 12 12:22 No Changes				



Results

- Back to Dashboard
- Status
- Changes
- Workspace
- Build with Parameters
- Delete Project
- Configure
- Favorite
- HTML Report
- Authorization
- Open Blue Ocean
- Rename

Project fudo-head-gui-all_pages

- [HTML Report](#)
- [Workspace](#)
- [Recent Changes](#)
- [Latest Test Result](#) (4 failures / ±0)

Build History [trend](#)

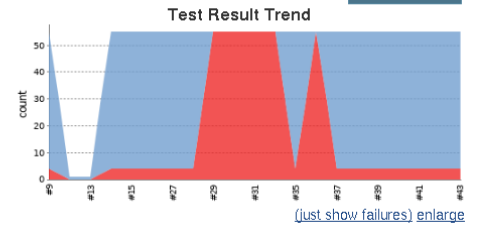
#43	Nov 14, 2018 3:44 PM
#42	Nov 14, 2018 1:40 PM

Permalinks

- [Last build \(#43\), 5 hr 26 min ago](#)
- [Last stable build \(#13\), 10 days ago](#)
- [Last successful build \(#13\), 10 days ago](#)
- [Last failed build \(#43\), 5 hr 26 min ago](#)
- [Last unsuccessful build \(#43\), 5 hr 26 min ago](#)
- [Last completed build \(#43\), 5 hr 26 min ago](#)

[add description](#)

[Disable Project](#)



Results Blue Ocean

✕ fudo-head-gui-all_pages < 43 Pipeline Changes Tests Artifacts Logout

Branch: — 11m 0s No changes
Commit: — 5 hours ago Started by upstream pipeline "fudo-head-pipeline" build #261



4 tests have failed

There are 0 new tests failing, 4 existing failing and 0 skipped.

Existing failures - 4

✕	> test_check_routing_net_conf_c765 - test_check_all_pages	<1s
✕	> test_server_add_page_c762 - test_check_all_pages	<1s
✕	> test_safe_add_page_c772 - test_check_all_pages	<1s
✕	> test_account_add_page_c773 - test_check_all_pages	<1s

Passed - 51

✓	> test_check_dashboard_c717 - test_check_all_pages	<1s
✓	> test_check_sessions_c718 - test_check_all_pages	<1s
✓	> test_check_users_c719 - test_check_all_pages	<1s
✓	> test_check_servers_c720 - test_check_all_pages	<1s
✓	> test_check_accounts_c721 - test_check_all_pages	<1s
✓	> test_check_listeners_c722 - test_check_all_pages	<1s
✓	> test_check_safes_c723 - test_check_all_pages	<1s
✓	> test_password_policies_c724 - test_check_all_pages	<1s
✓	> test_password_changers_c725 - test_check_all_pages	<1s



Questions ?

