SkillAssure

Jenkins Essentials

SkillAssure

Learning Objectives

- Overview of Jenkins
- Installation & Configuration
- Creating a Sample Job
- Jenkins Plugins
- The Big Picture

SkillAssure

About Jenkins

Jenkins

Open Source

- You can view source
- License is free
- Good Support
- Frequent Updates

Web Based JAVA Tool

- Built in JAVA
- Works well with other languages
- Organized as JAVA Deployable package
- Built in web server
- Very Easy to get it up and running

Highly Extensible

- Varity of Plugins Available
- Flexibility to create custom plugins


HISTORY

SkillAssure

History of Jenkins

Kohsuke Kawaguchi

Started around 2007 as Hudson

ORACLE Bought SUN

2009-2010 disturbance in the ranks

Rapid rise in popularity in 2008

Replaced Cruise Control

2011 The great division

HUDSON and JENKINS Separated

SkillAssure

The Jenkins / Hudson Controversy

SkillAssure

The Jenkins / Hudson Controversy

Hudson

Oracale/Eclipse

Jenkins

Kohsuke Kawaguchi

The Problem

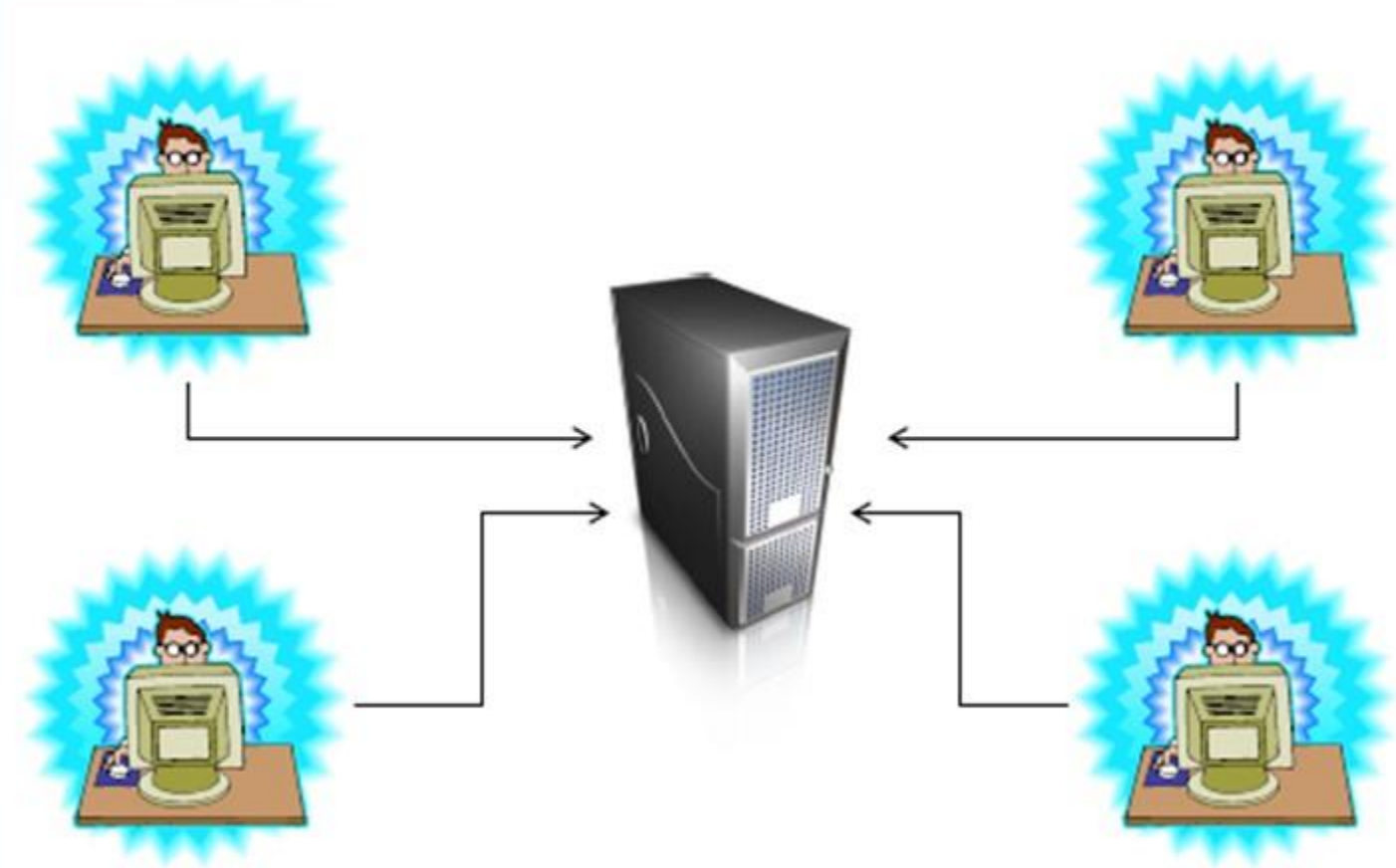
Integration is Painful

SkillAssure

The Problem

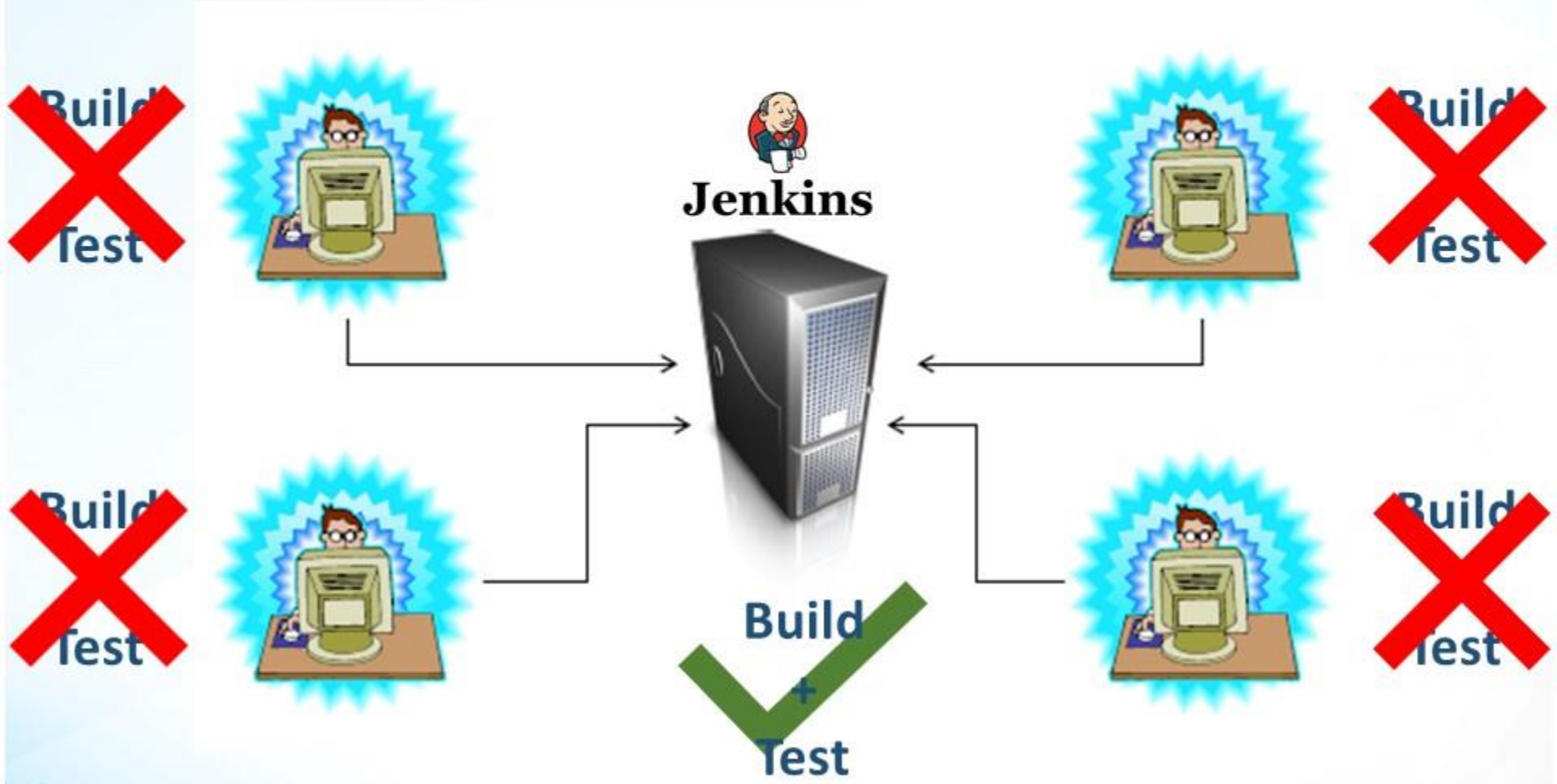
SkillAssure

What is Continuous Integration? (CI)



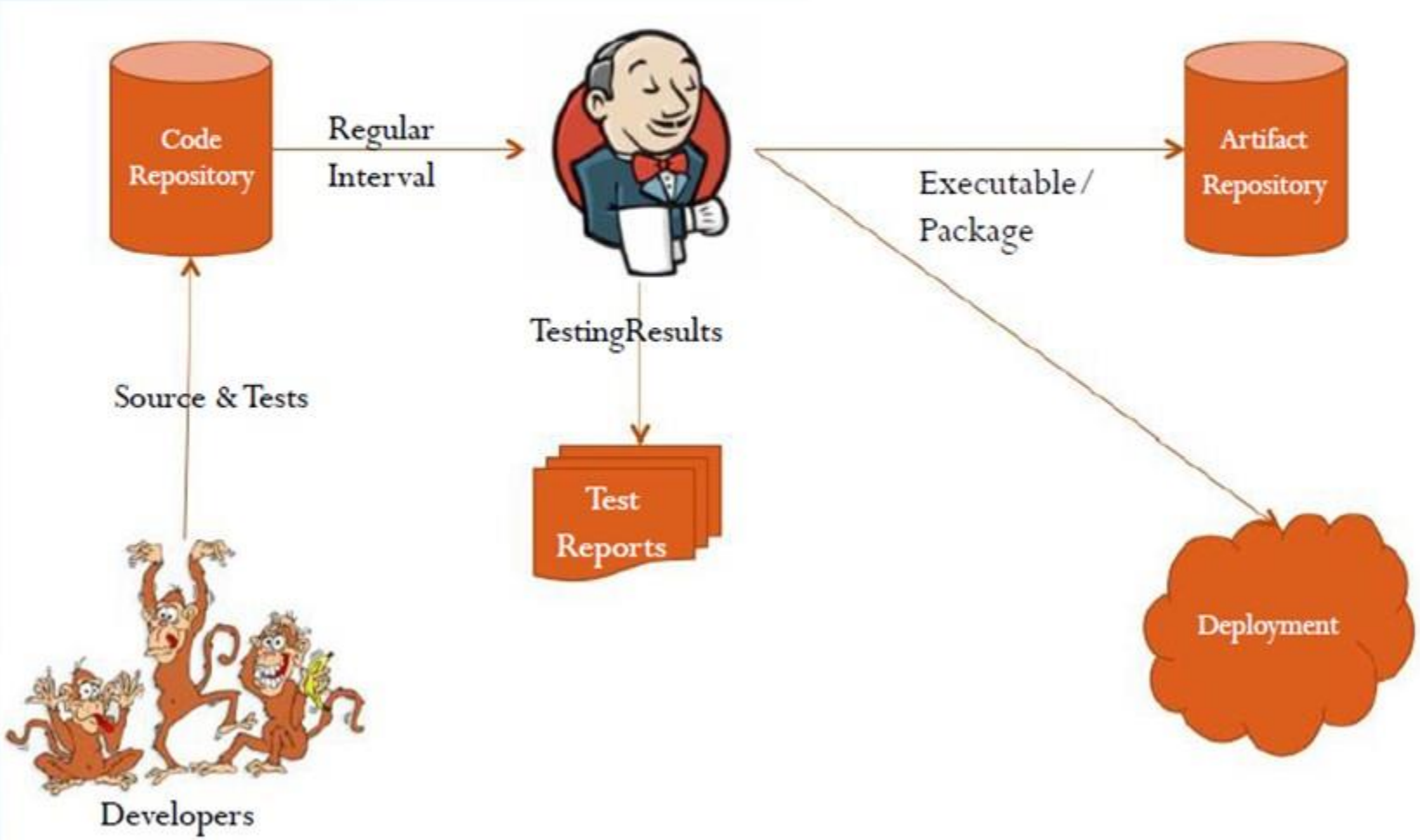
SkillAssure

Where Jenkins Fits In



SkillAssure

Where Jenkins Fits In



SkillAssure

Installation & Configuration

SkillAssure

Prerequisites

- Java 1.5 or higher
- Java in your path (optional, but preferred)
- Web server (only if you want to host outside of Jenkins)

- **Install JAVA**
  - sudo add-apt-repository ppa:webupd8team/java
  - sudo apt-get update
  - sudo apt-get install oracle-java8-installer
- **Set Environment Variable**
  - sudo update-alternatives --config java
  - sudo vim /etc/environment
  - JAVA_HOME="/usr/lib/jvm/java-8-oracle"
  - echo $JAVA_HOME

SkillAssure

Getting Jenkins

- Open the Jenkins Website https://jenkins.io/
- Got to Download
  - Weekly Release - A new release is produced weekly to deliver bug fixes and features to users and plugin developers.
  - **Long Term Support Release** – Stable release b/w every 12 weekly release.
- Download the latest stable release.

SkillAssure

Starting Jenkins

- Open Command Line
- Go to the folder where you have kept the war file.
- Type the command “java –jar Jenkins.war”

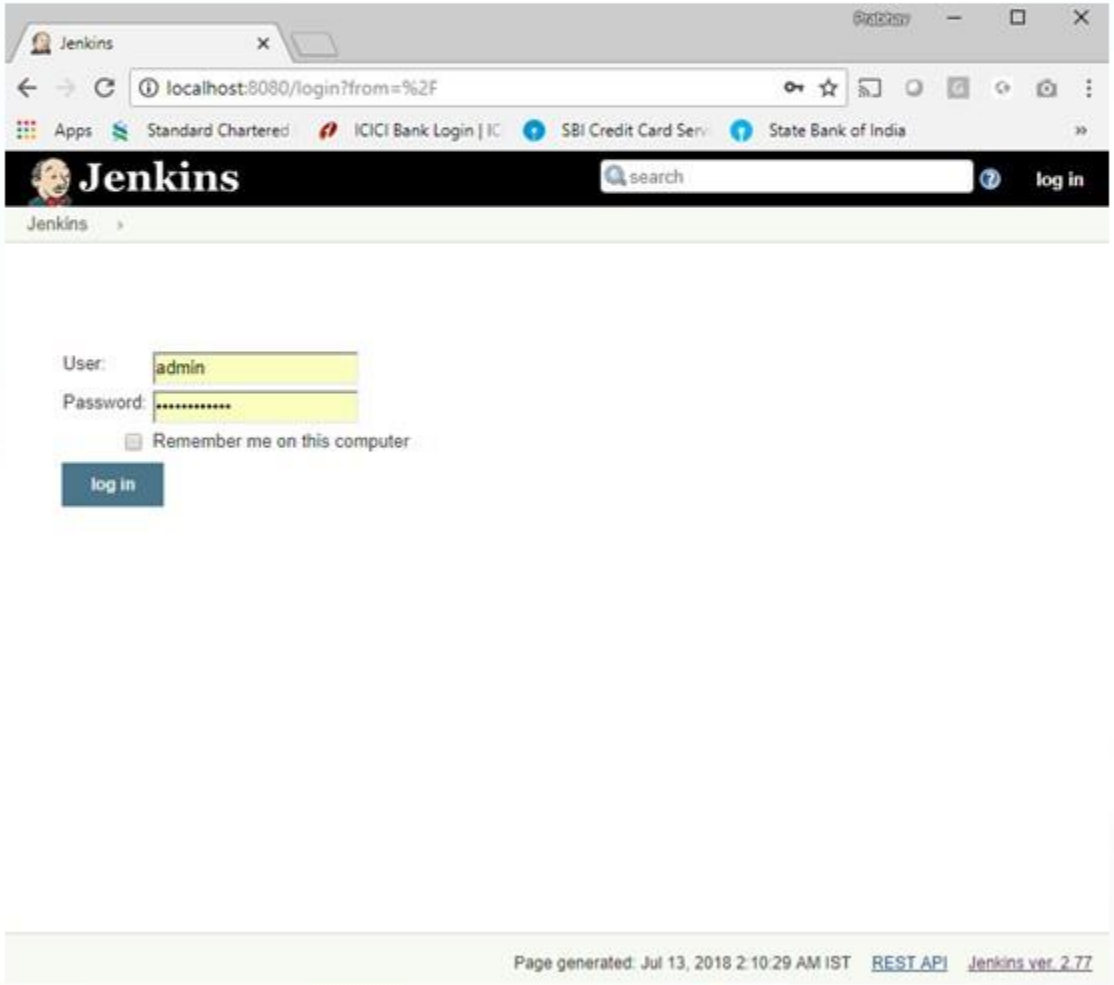
Note: You must have the JAVA Path set in the environment variables.

Jenkins

SkillAssure

Testing Jenkins

- The default port number for Jenkins is "8080"
- Open the web server http://localhost:8080



Hit Ctrl+C to stop this Jenkins.

SkillAssure

Common Command Line Options

- Run Jenkins on a different port number.
  - java –jar Jenkins.war --httpPort=8081
- Run Jenkins on HTTPS
  - java –jar Jenkins.war --httpPort=-1 --httpsPort=443
  - This is commonly used when you want to secure Jenkins
- Open the web server and see Jenkins running.

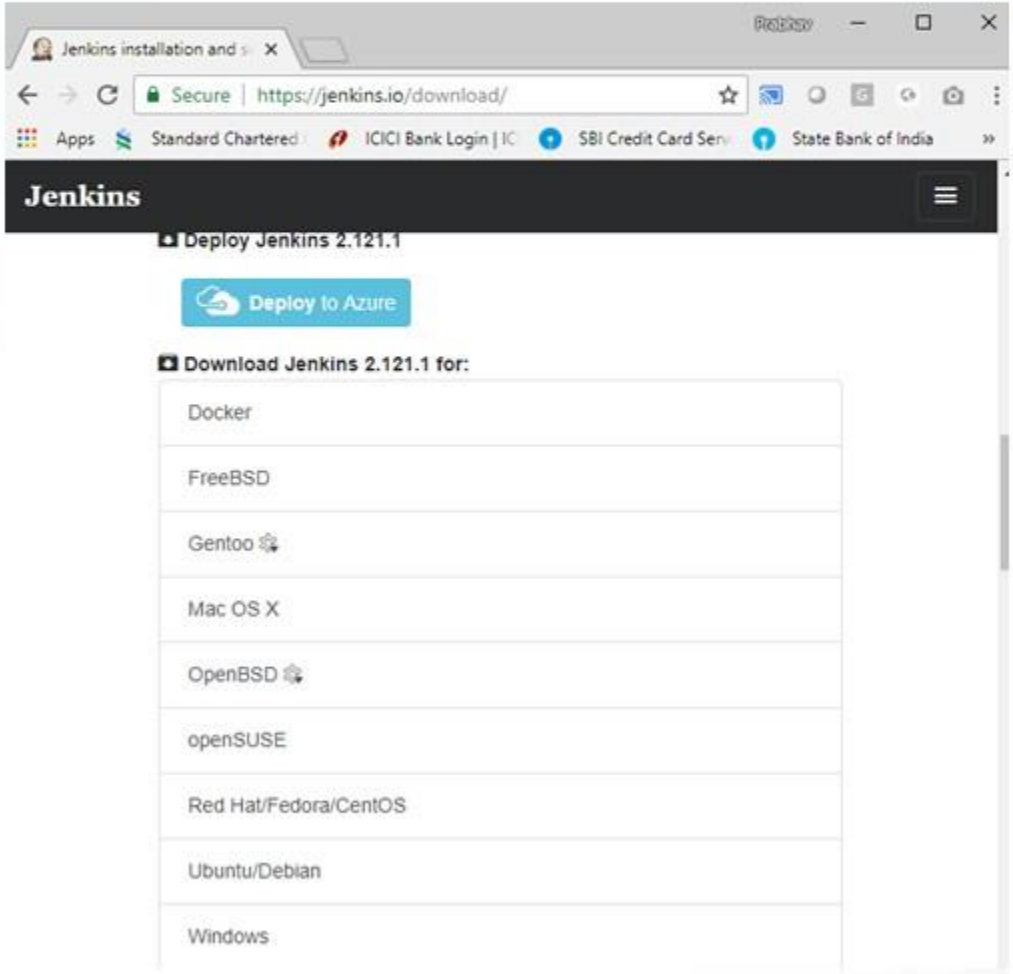
More command line options can be found here

https://wiki.jenkins.io/display/JENKINS/Starting+and+Accessing+Jenkins

SkillAssure

Getting the Service Installer

- Open the Jenkins Website https://jenkins.io/
- Got to Download
- See the various available service packages.
- Select and download the package based on the OS



SkillAssure

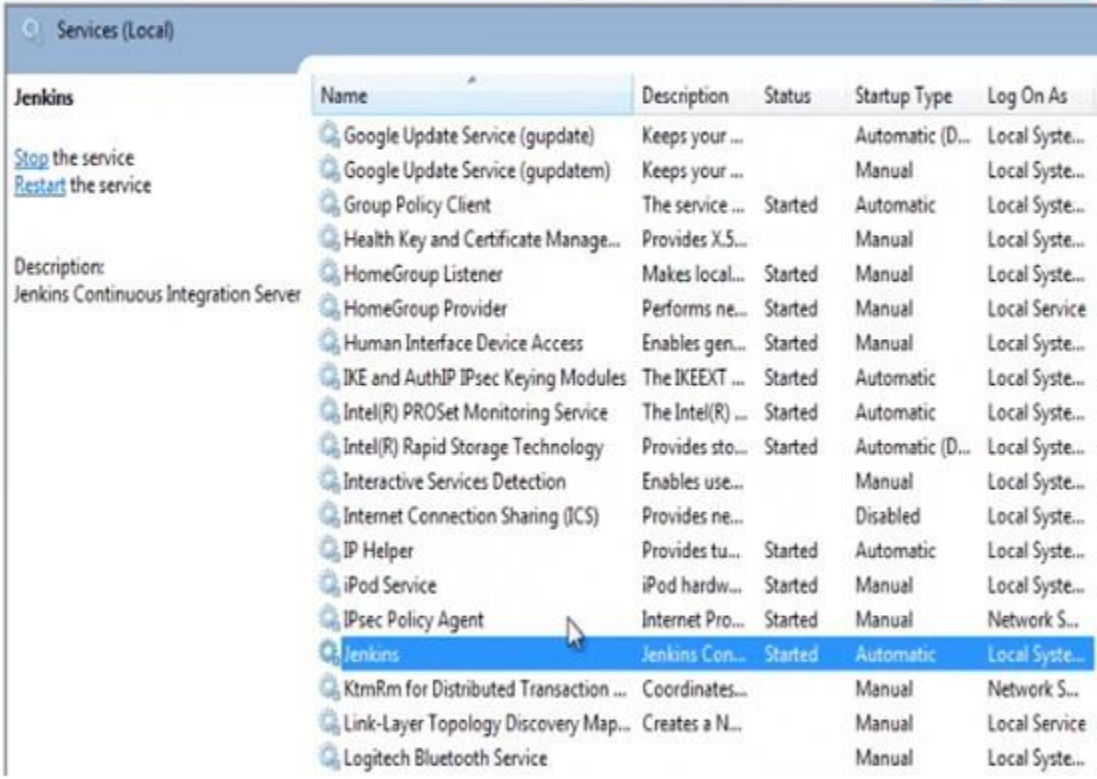
Installing as a Service - Ubuntu

- Open the terminal
- Get Jenkins
  - wget -q -O - https://pkg.jenkins.io/debian-stable/jenkins.io.key | sudo apt-key add -
- Add Library in /etc/apt/sources.list
  - deb https://pkg.jenkins.io/debian-stable binary/
- Update
  - sudo apt-get update
- Install
  - sudo apt-get install jenkins

SkillAssure

Installing as a Service - Windows

- Open the Zip file
- Run Jenkins.msi file
  (windows installer Package)
- Install the file
- We can check "Jenkins running as a Service"
- Restart System



SkillAssure

Alternate Method

- Go to manage Jenkins
- Select Install as windows service
- Select directory
- Click Install
- Restart System

SkillAssure

Nodes

- Jenkins server is called as Master and its various nodes are called as Slaves
- Ability to function as distributes build server.
- Make Jenkins more powerful by distributing builds across multiple machines. Provides ability to act like a cluster
- Important to think about nodes when you are designing.
- Makes build faster
- Ability to run a build on different platforms

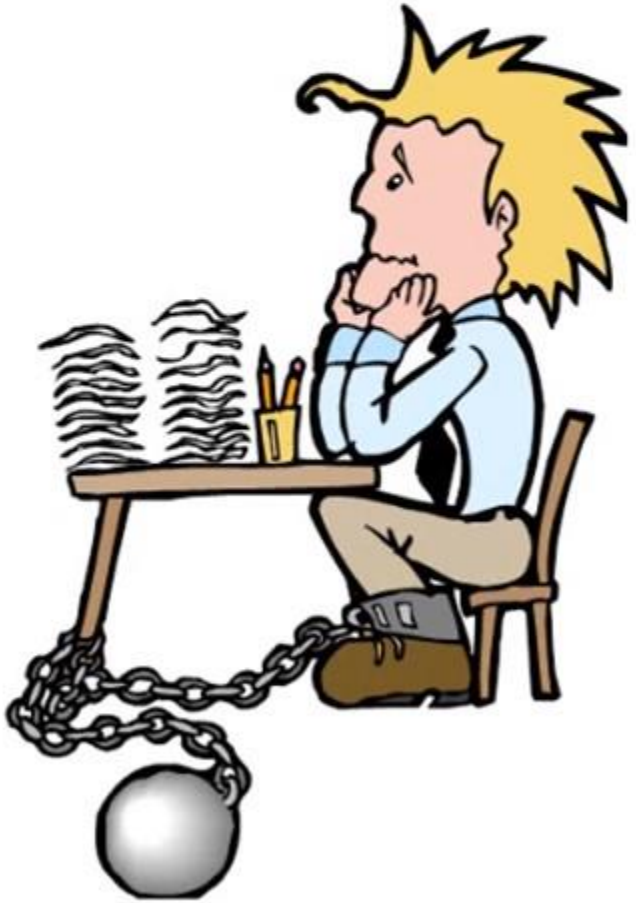
SkillAssure

Nodes

SkillAssure

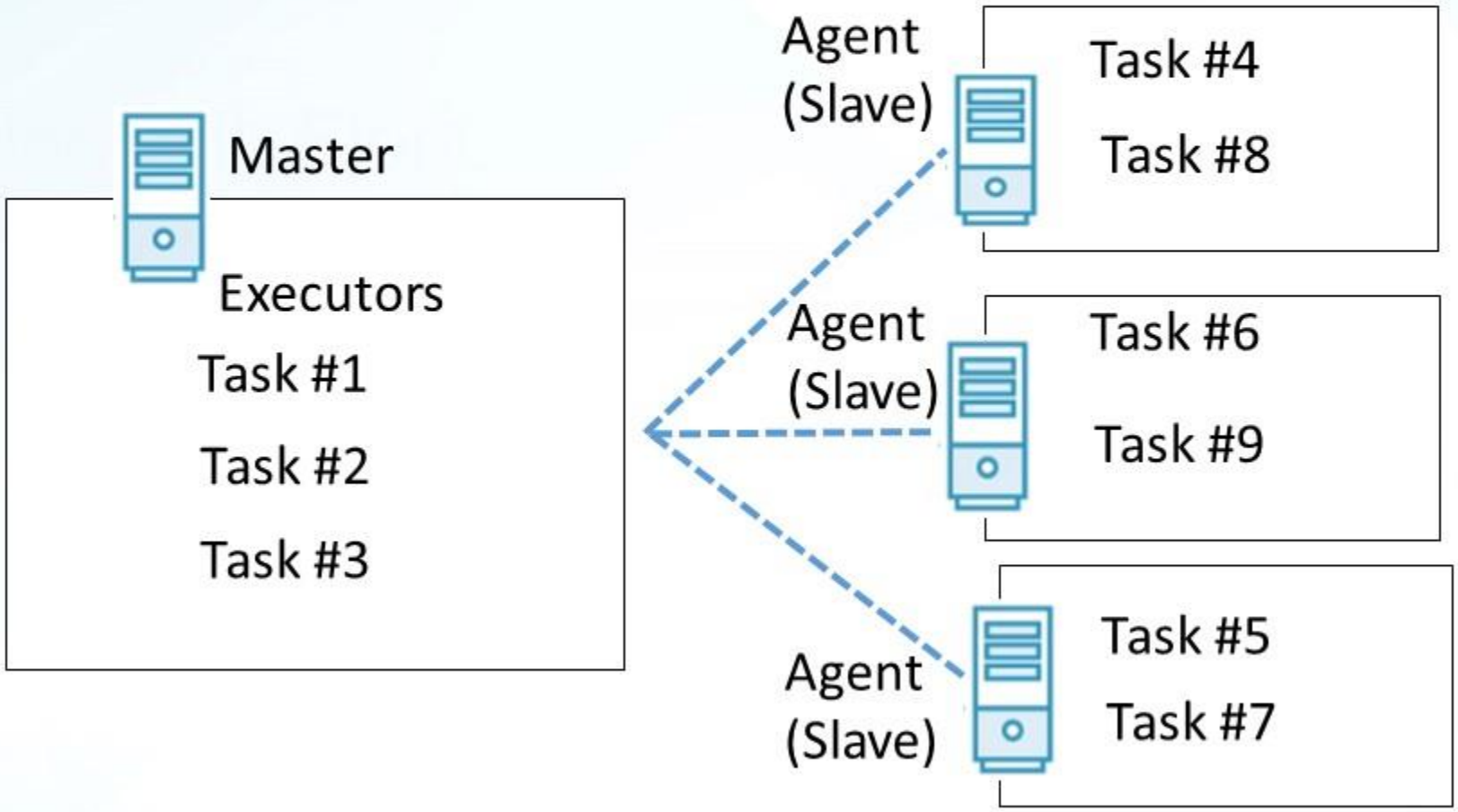
Nodes

Run my automated tests, boy!



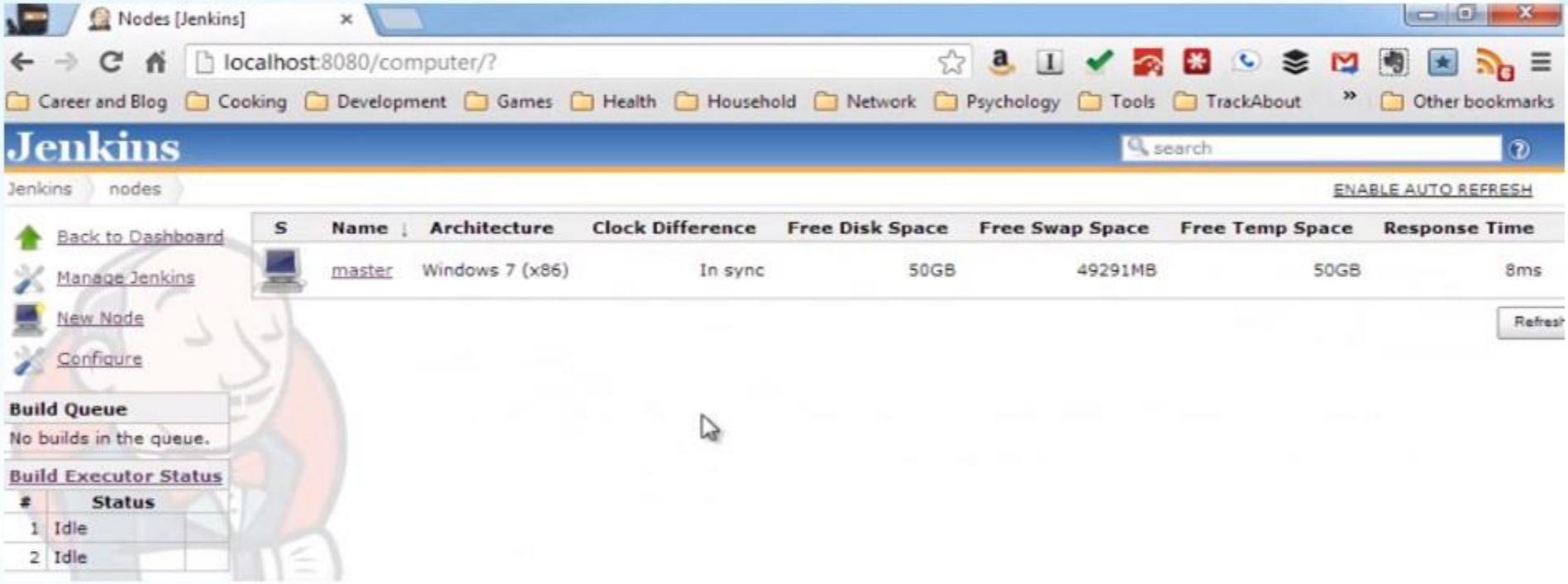
SkillAssure

The Master Agent Model



SkillAssure

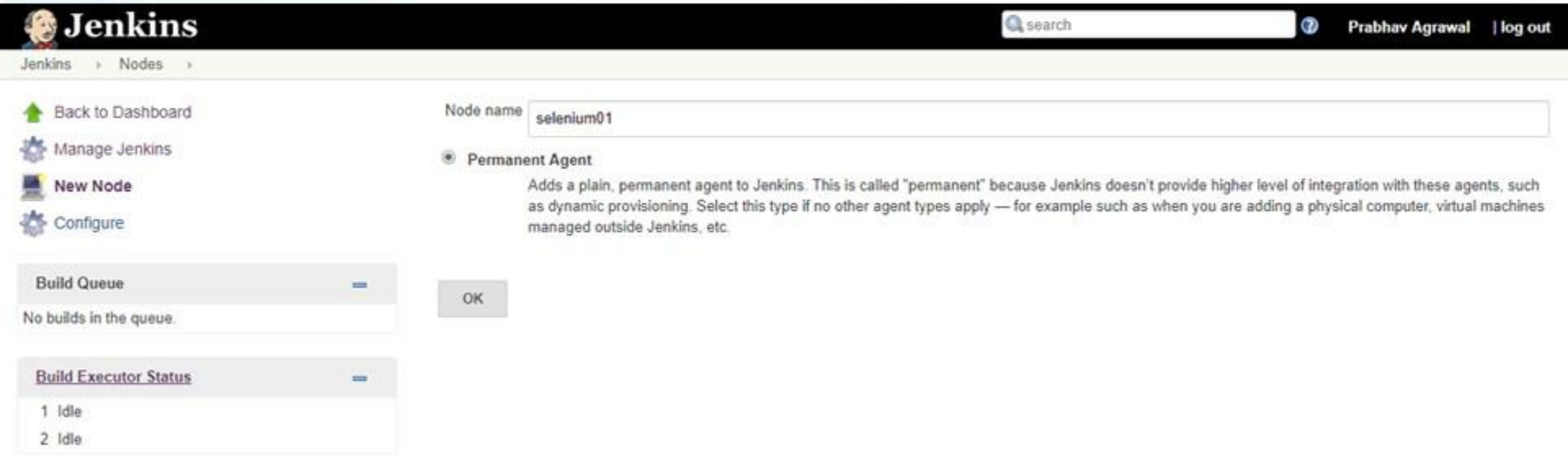
Configuring a Node



- ☞ Go to localhost:8080
- ☞ Go to Manage Jenkins
- ☞ Go to Manage nodes

SkillAssure

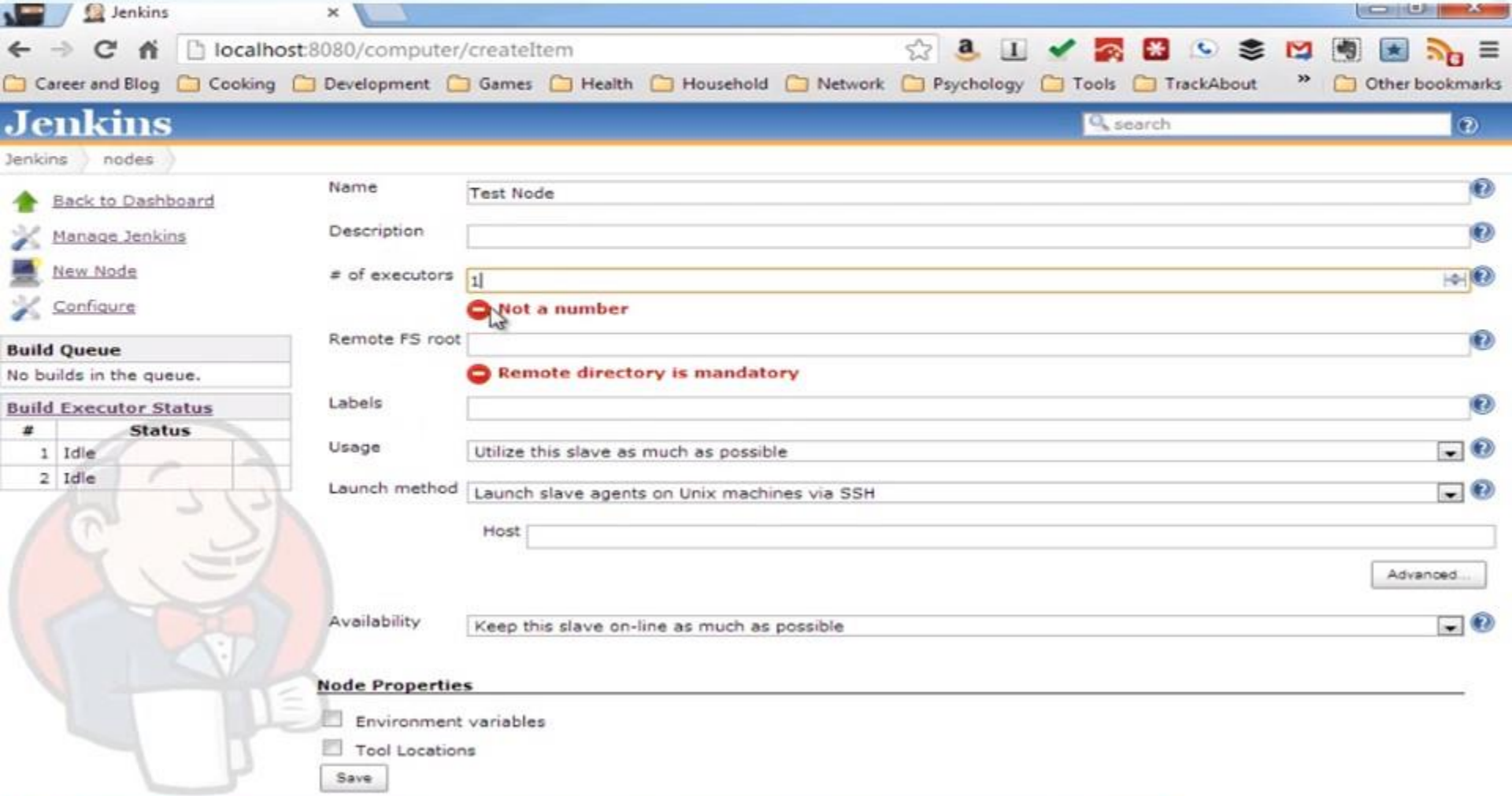
Configuring a Node



- ☞ Go to New nodes
- ☞ Give node name
- ☞ Select the node type
- ☞ Click Ok

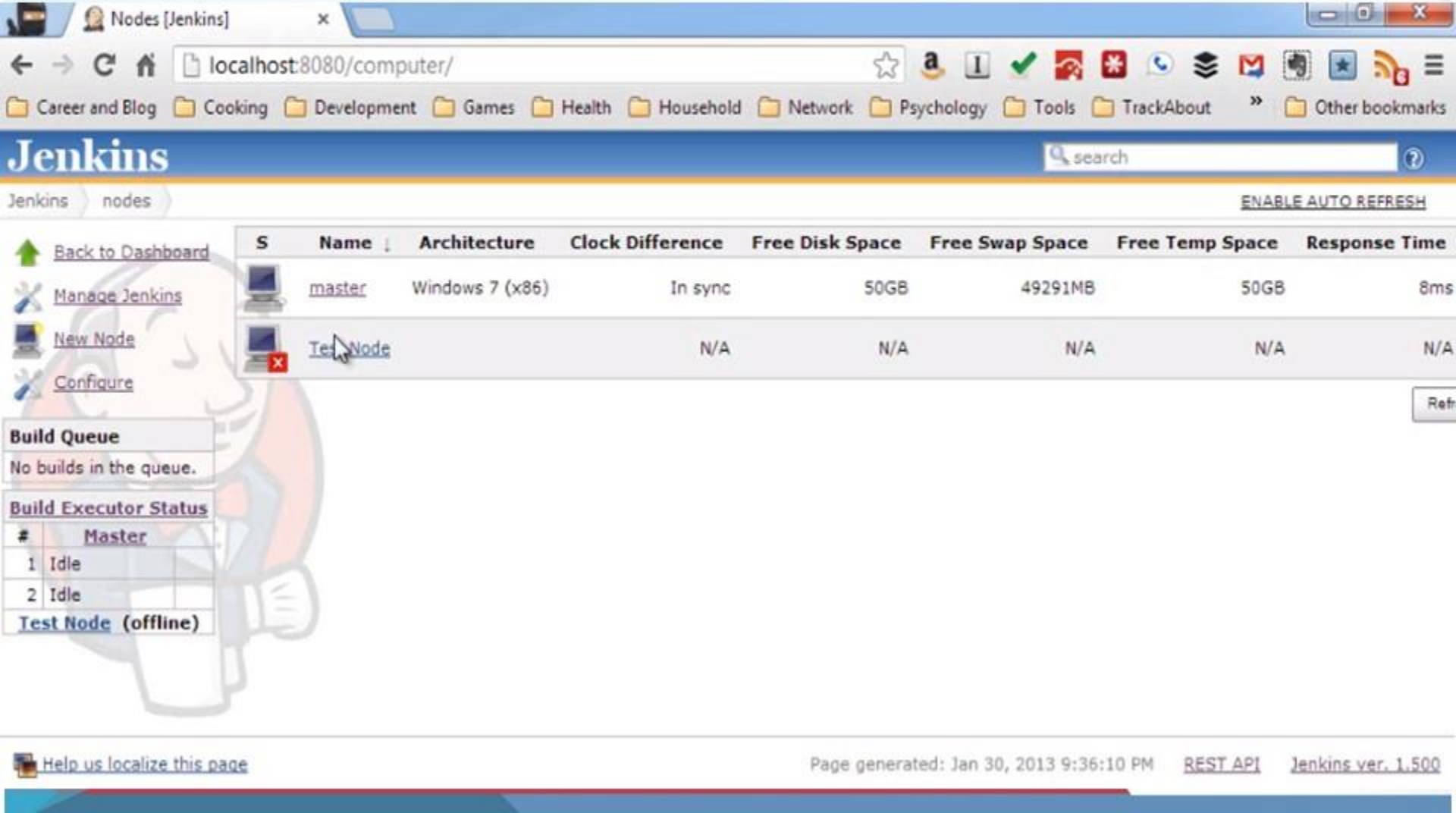
SkillAssure

Configuring a Node



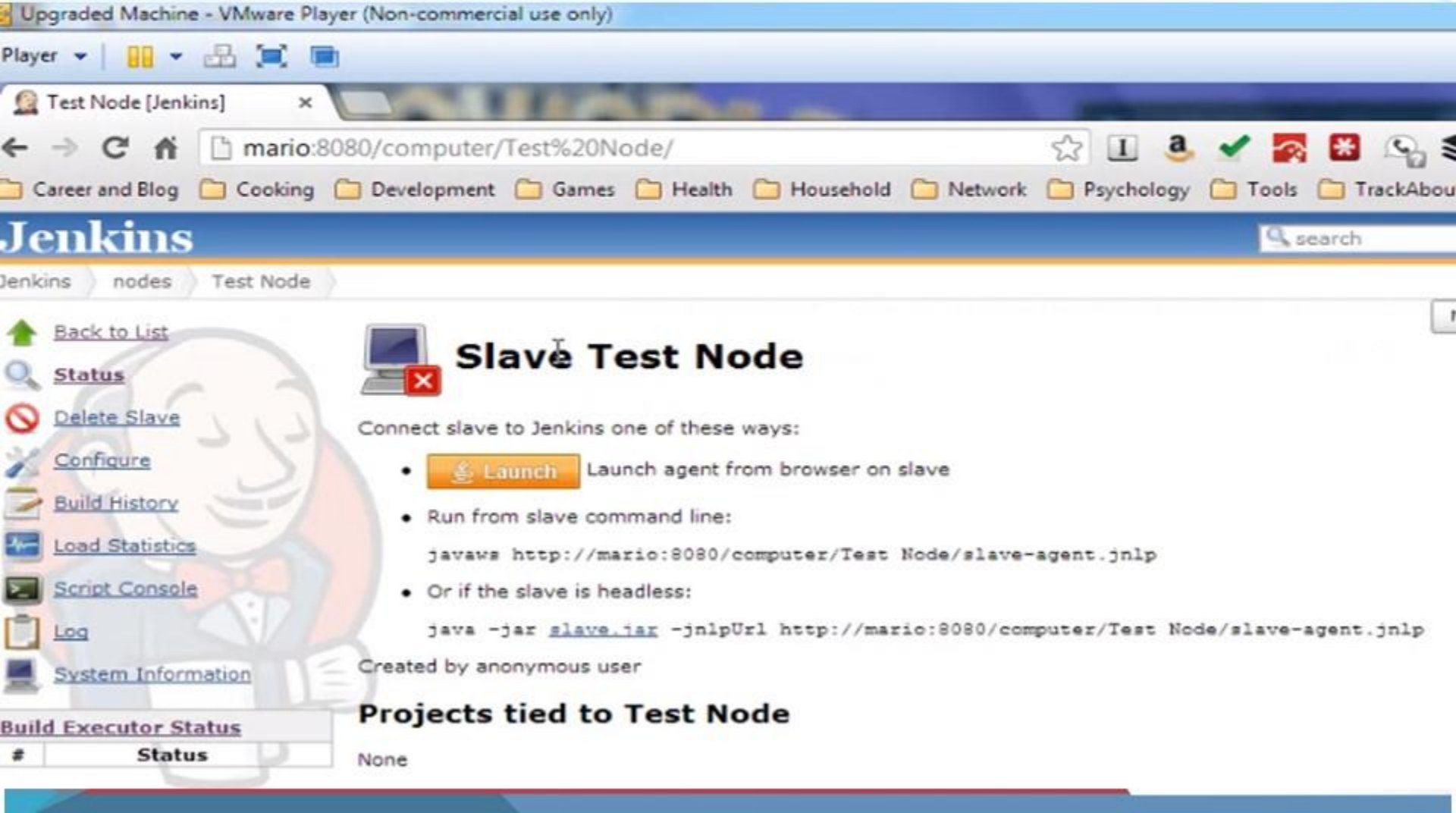
SkillAssure

Configuring a Node



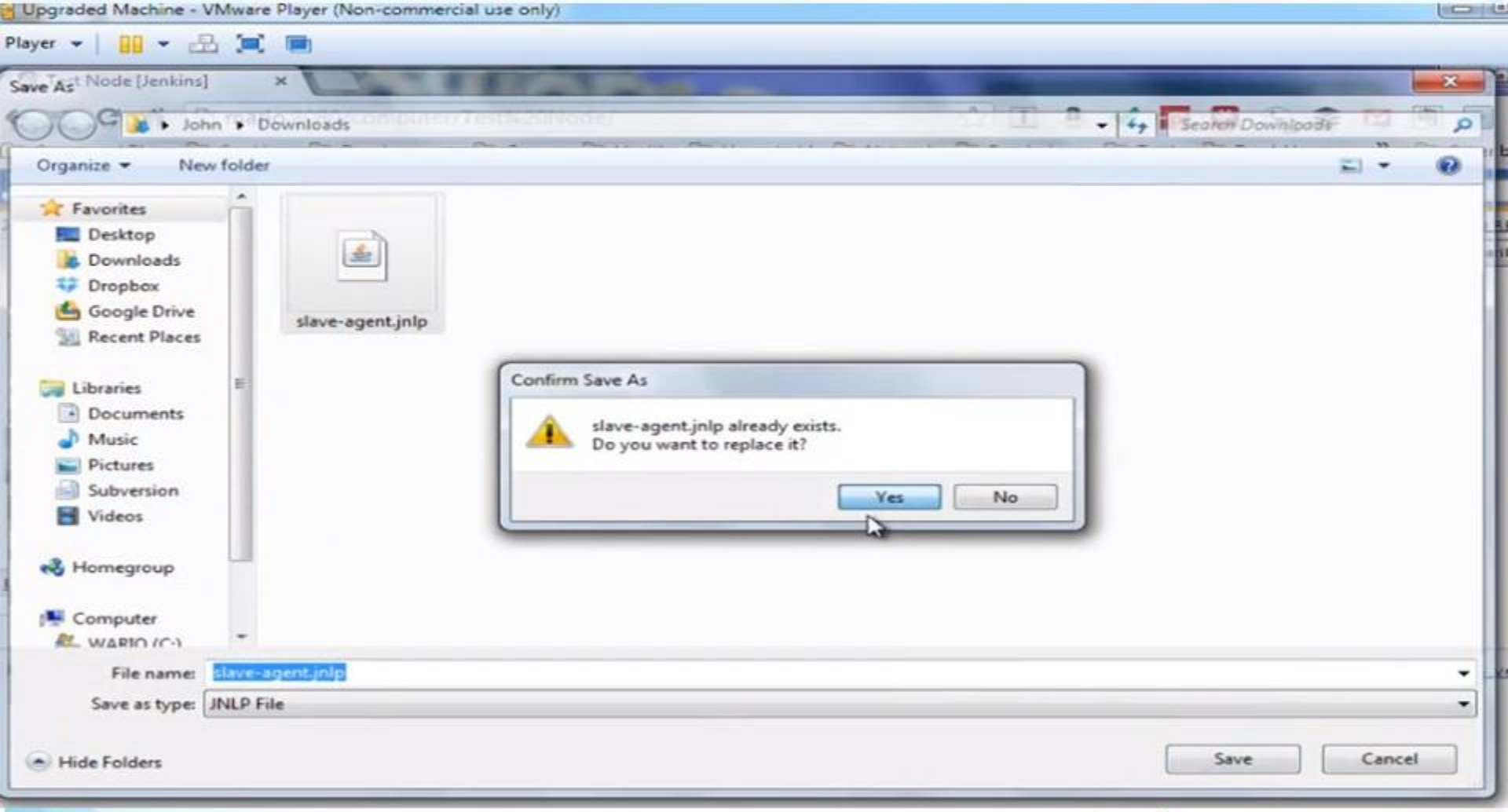
SkillAssure

Setting up a Node



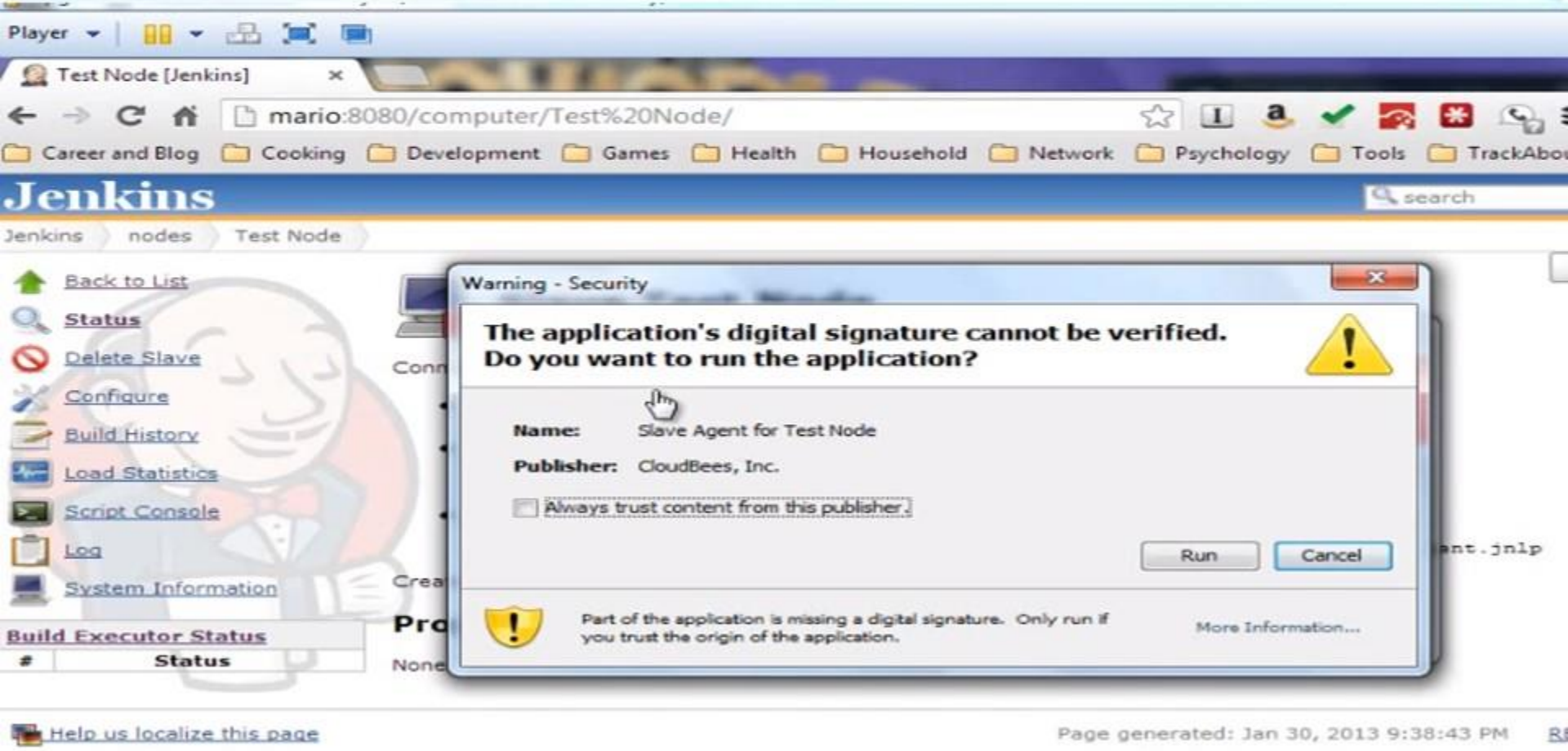
SkillAssure

Setting up a Node



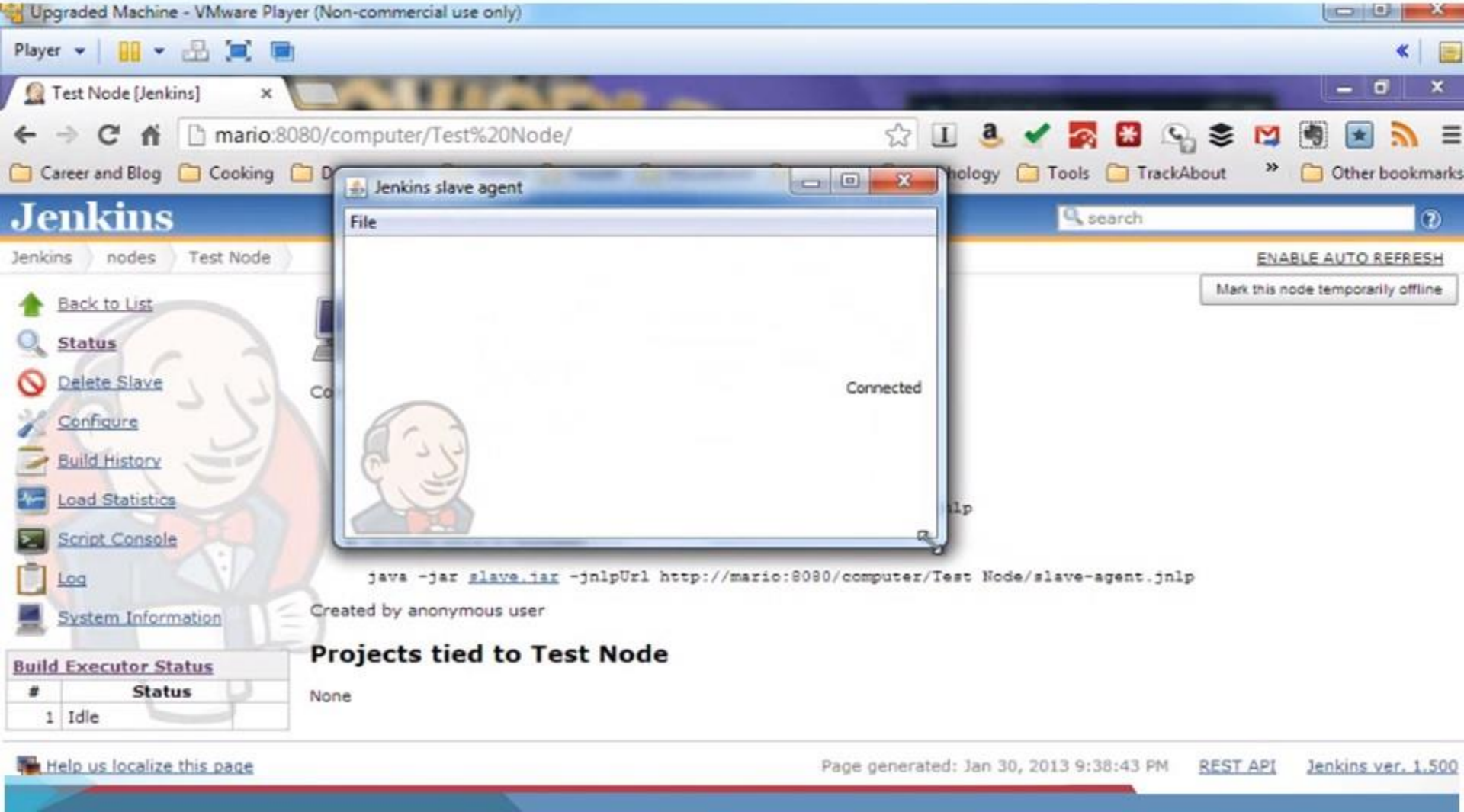
SkillAssure

Setting up a Node



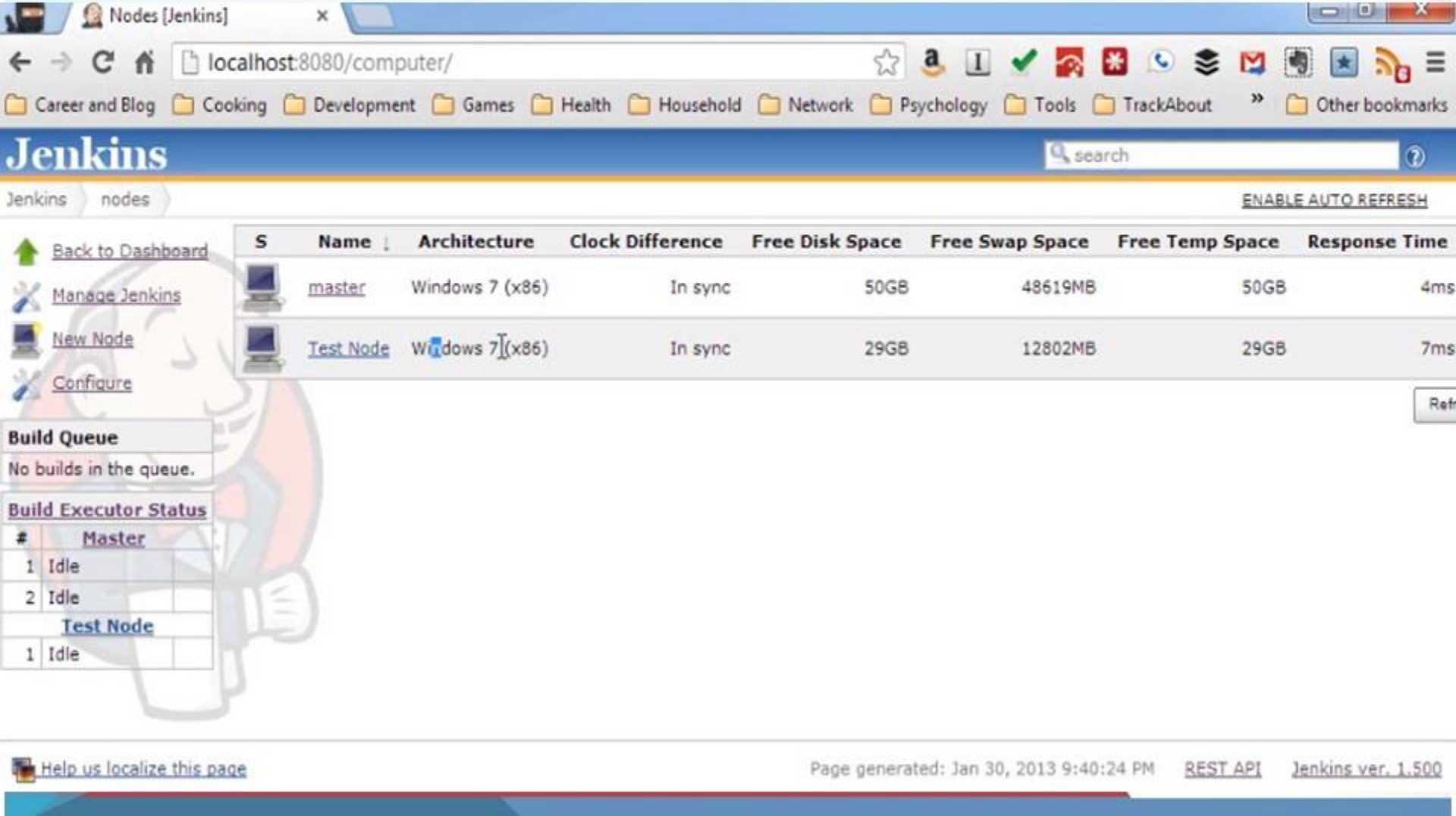
SkillAssure

Setting up a Node



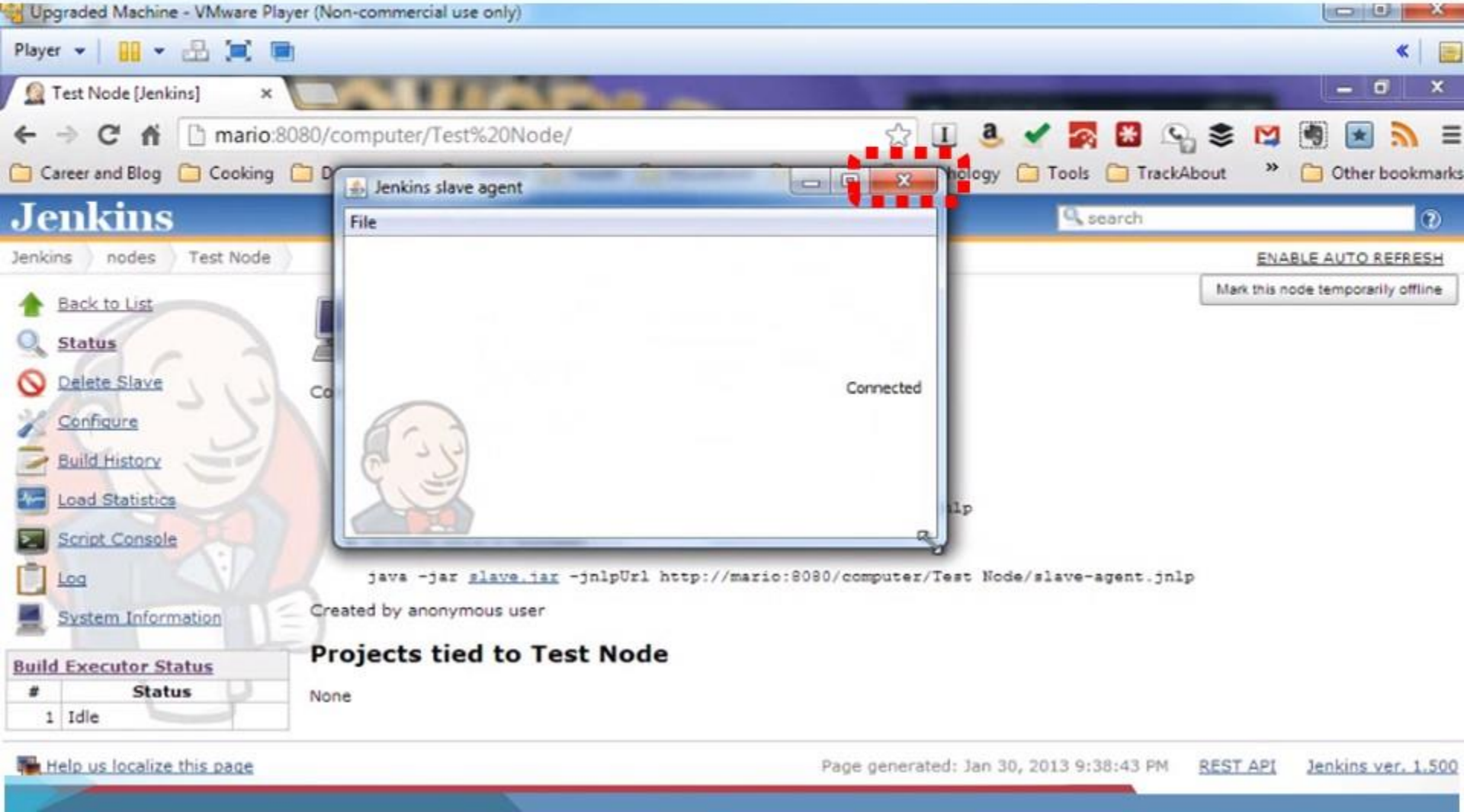
SkillAssure

Setting up a Node



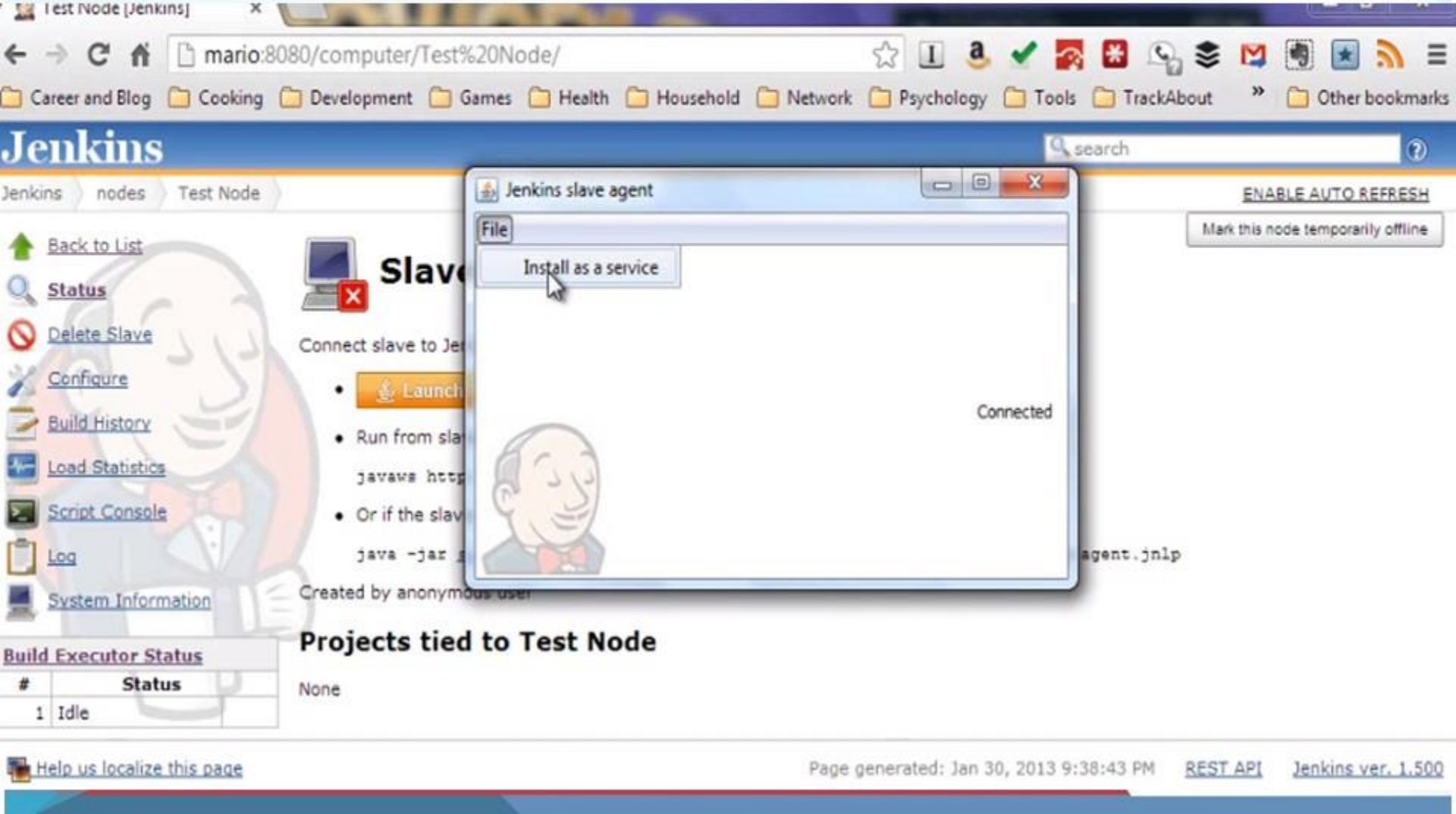
SkillAssure

Setting up a Node



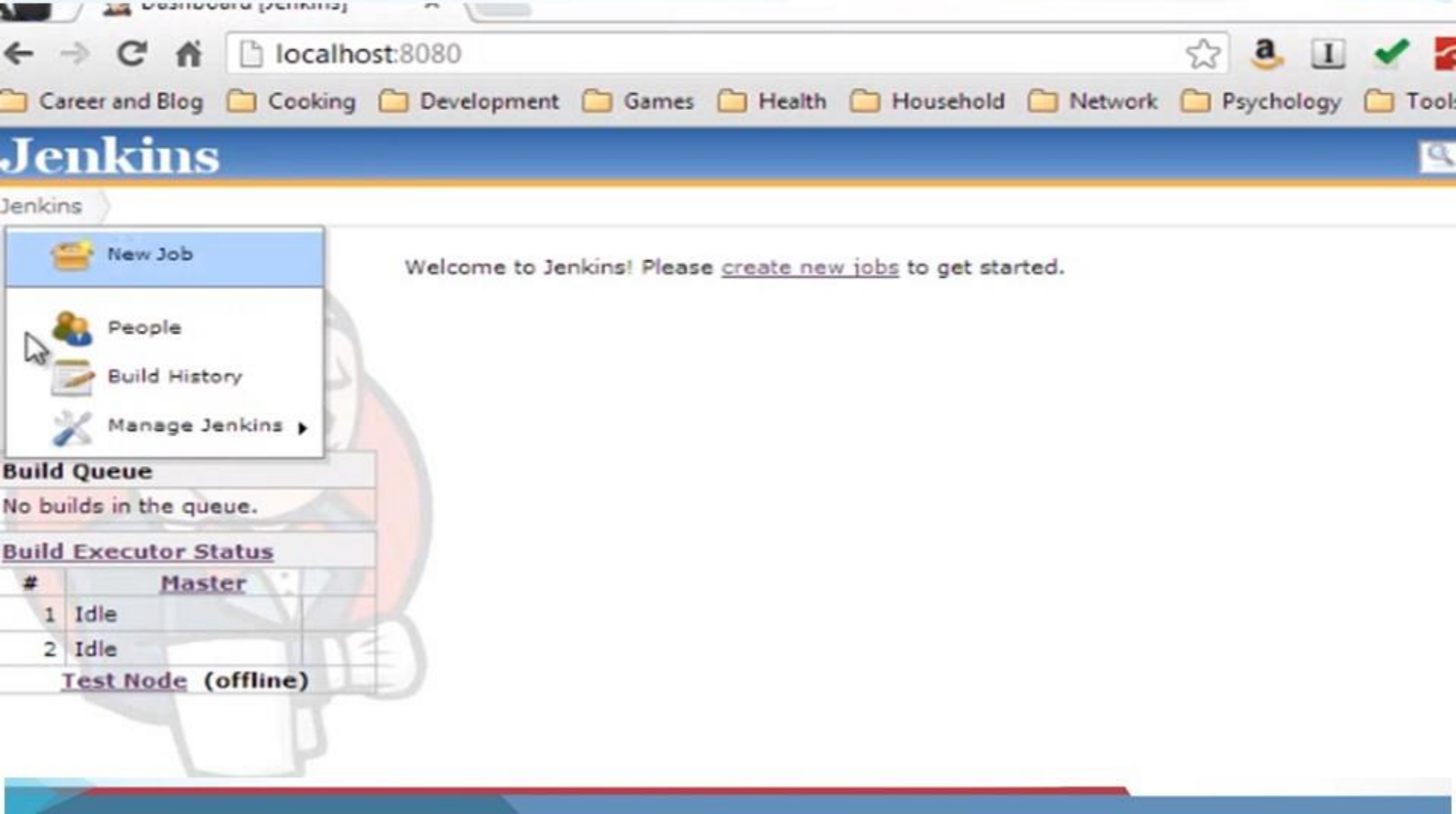
SkillAssure

Setting up a Node



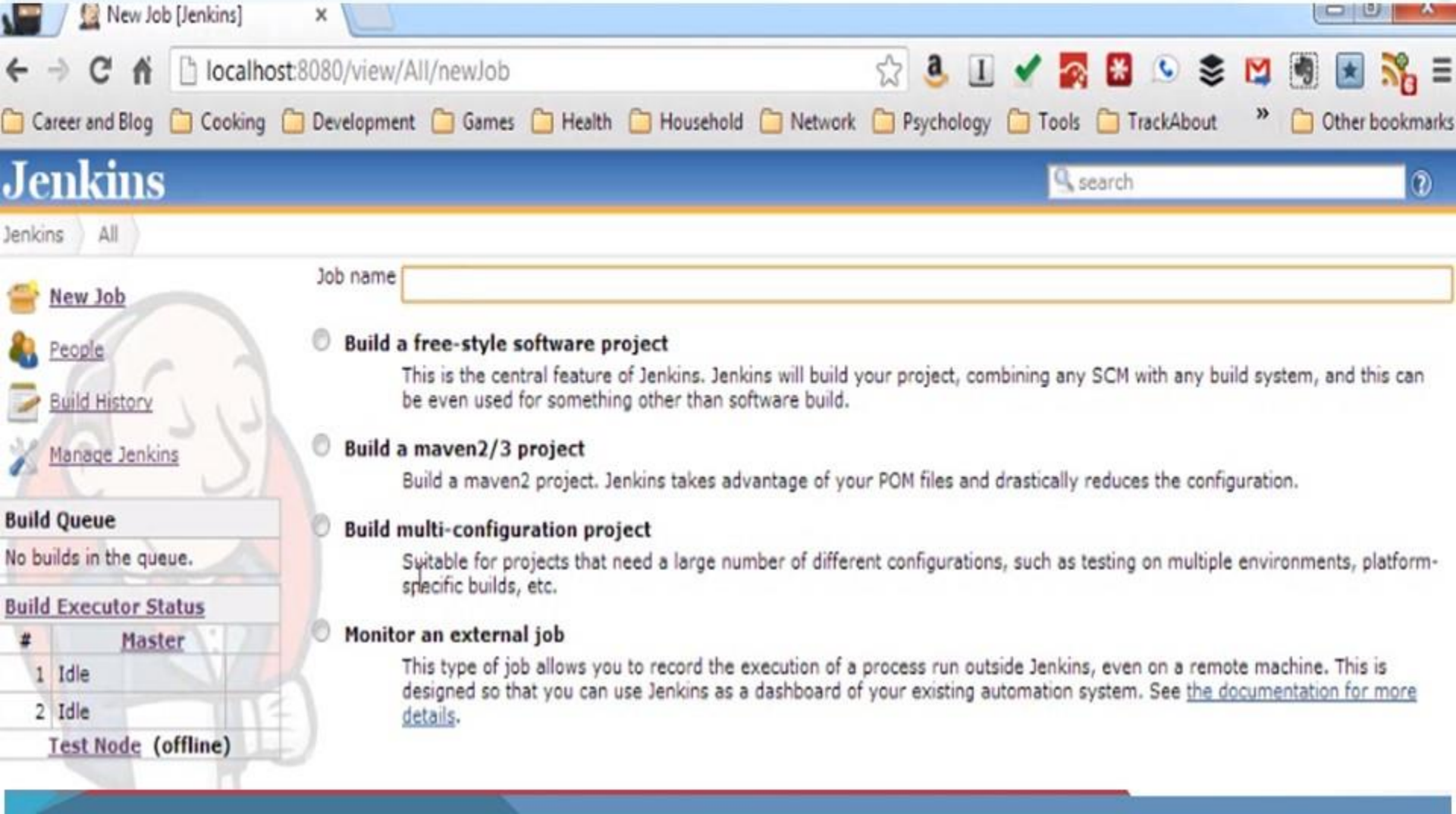
SkillAssure

UI Tour



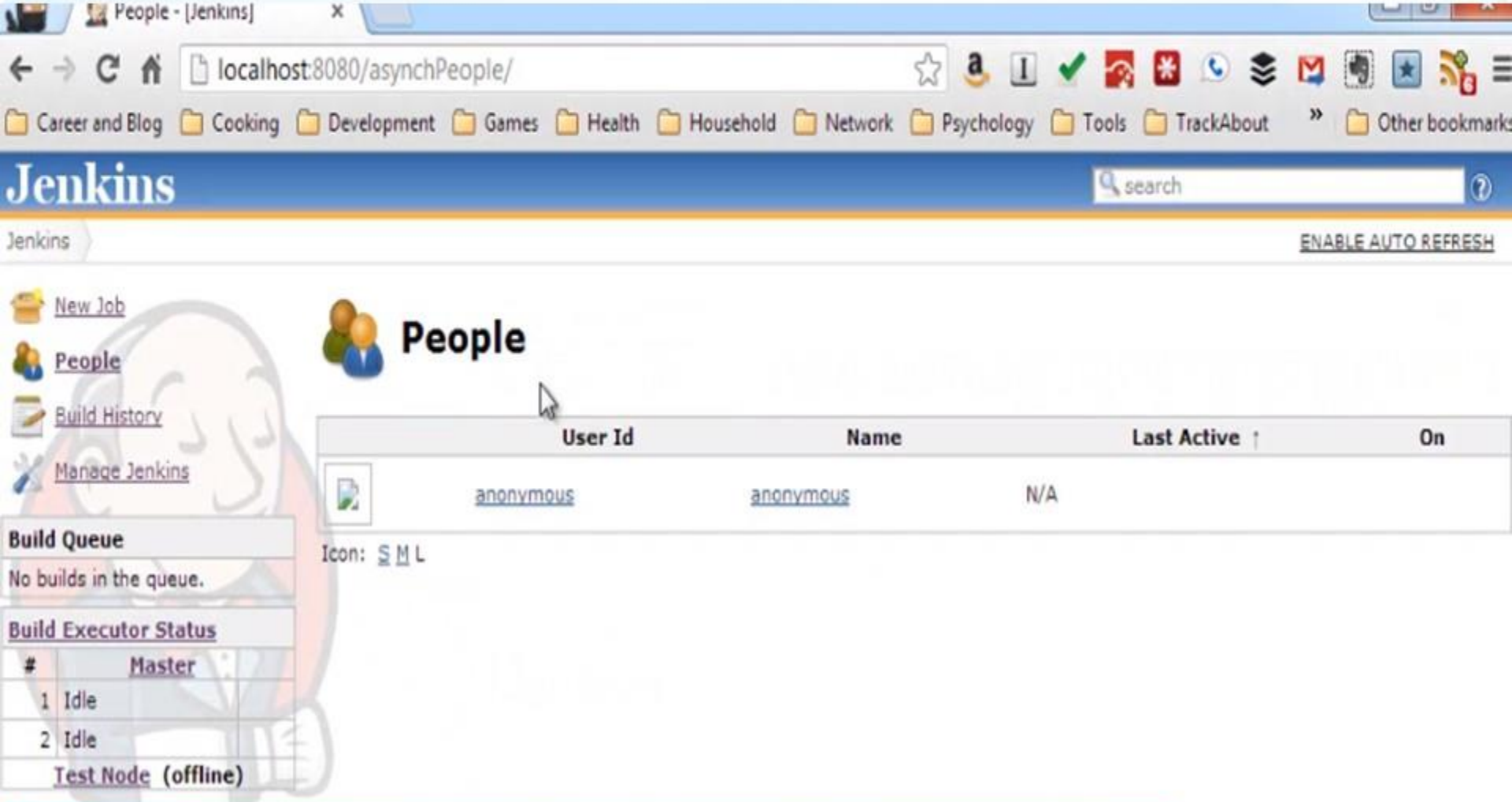
SkillAssure

UI Tour – New Job



SkillAssure

UI Tour - People



SkillAssure

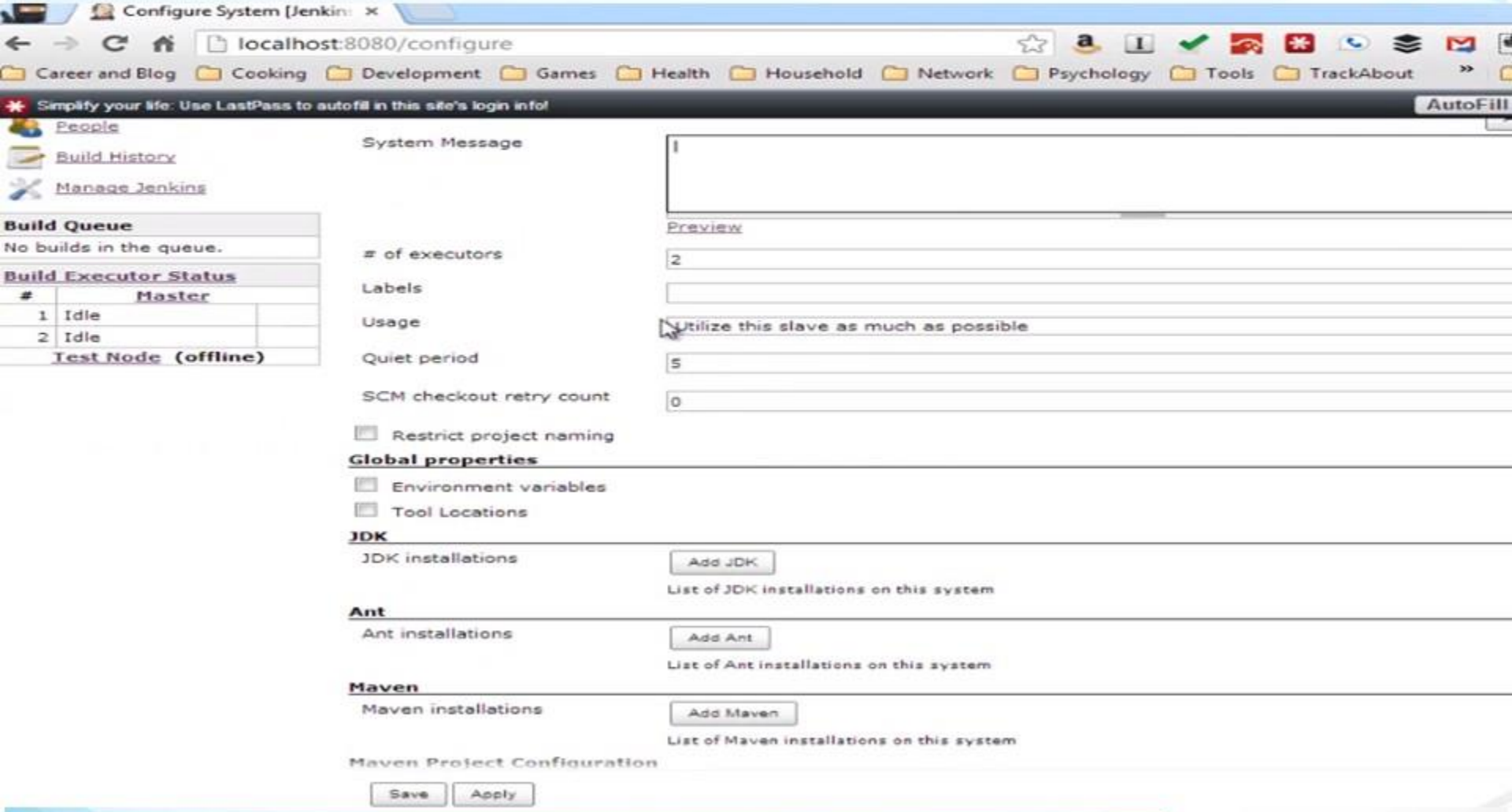
UI Tour – Build History

SkillAssure

UI Tour – Manage Jenkins

SkillAssure

UI Tour – Configure System

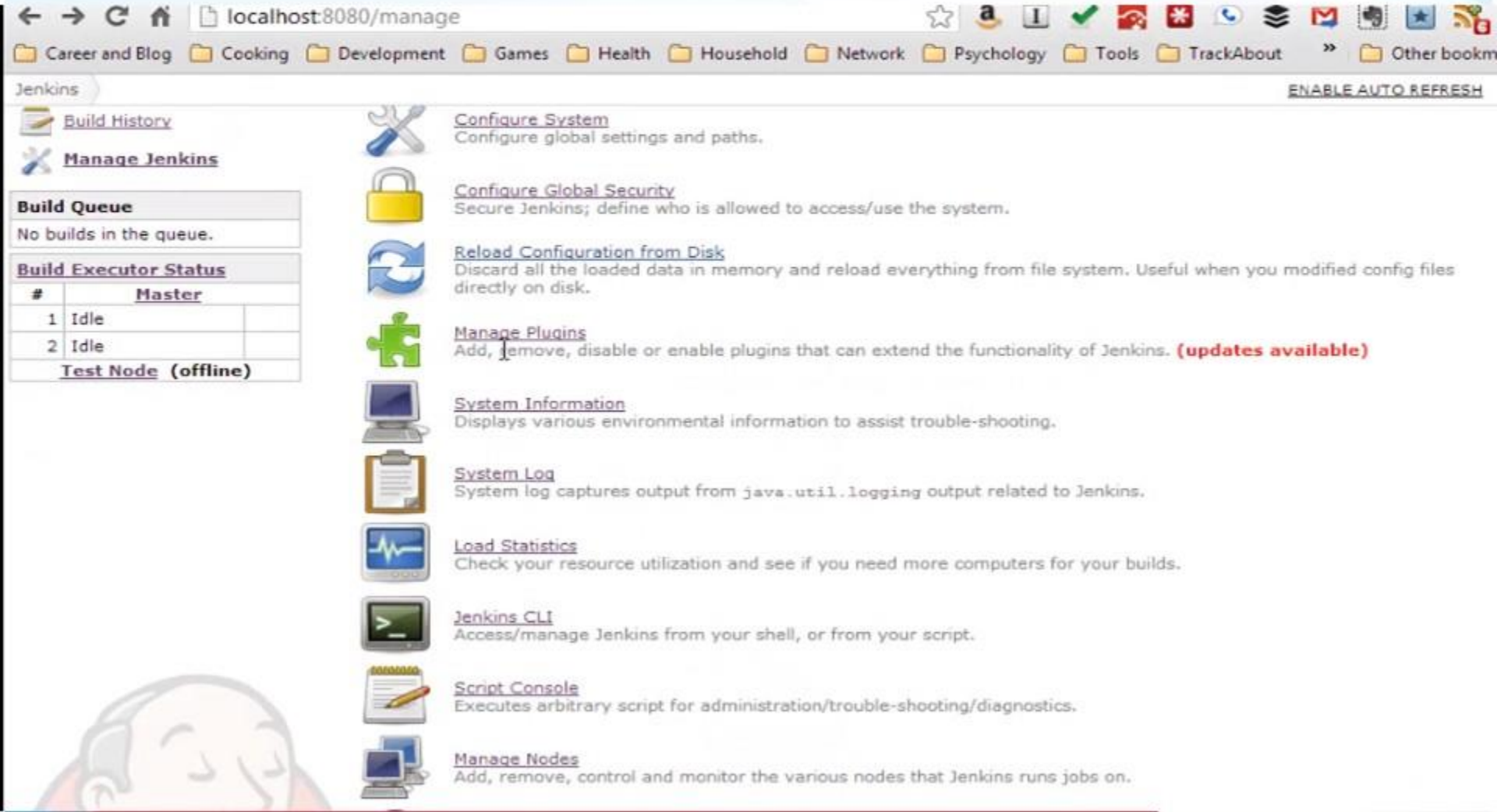


SkillAssure

UI Tour – Configure Global Security

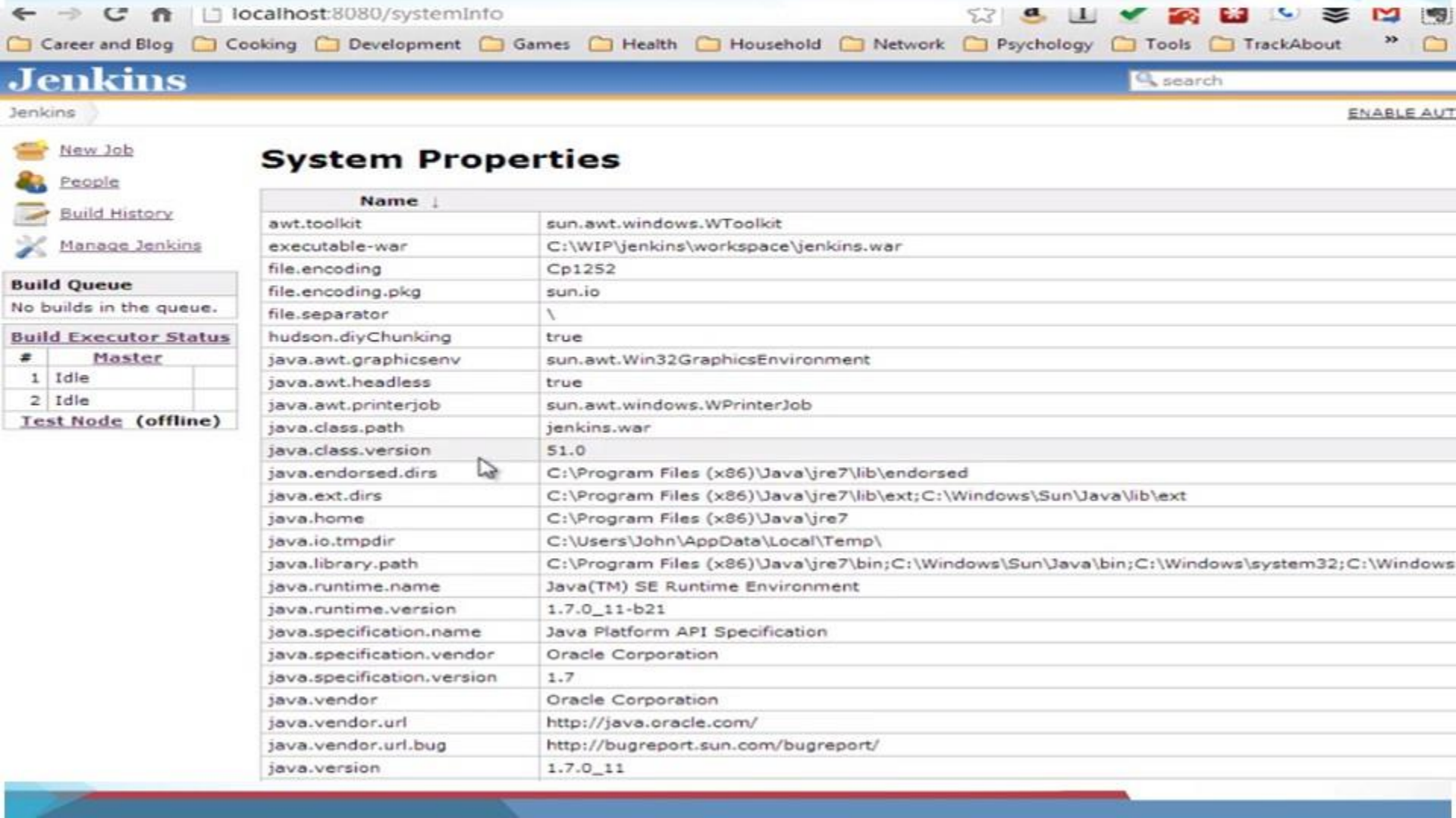
SkillAssure

UI Tour



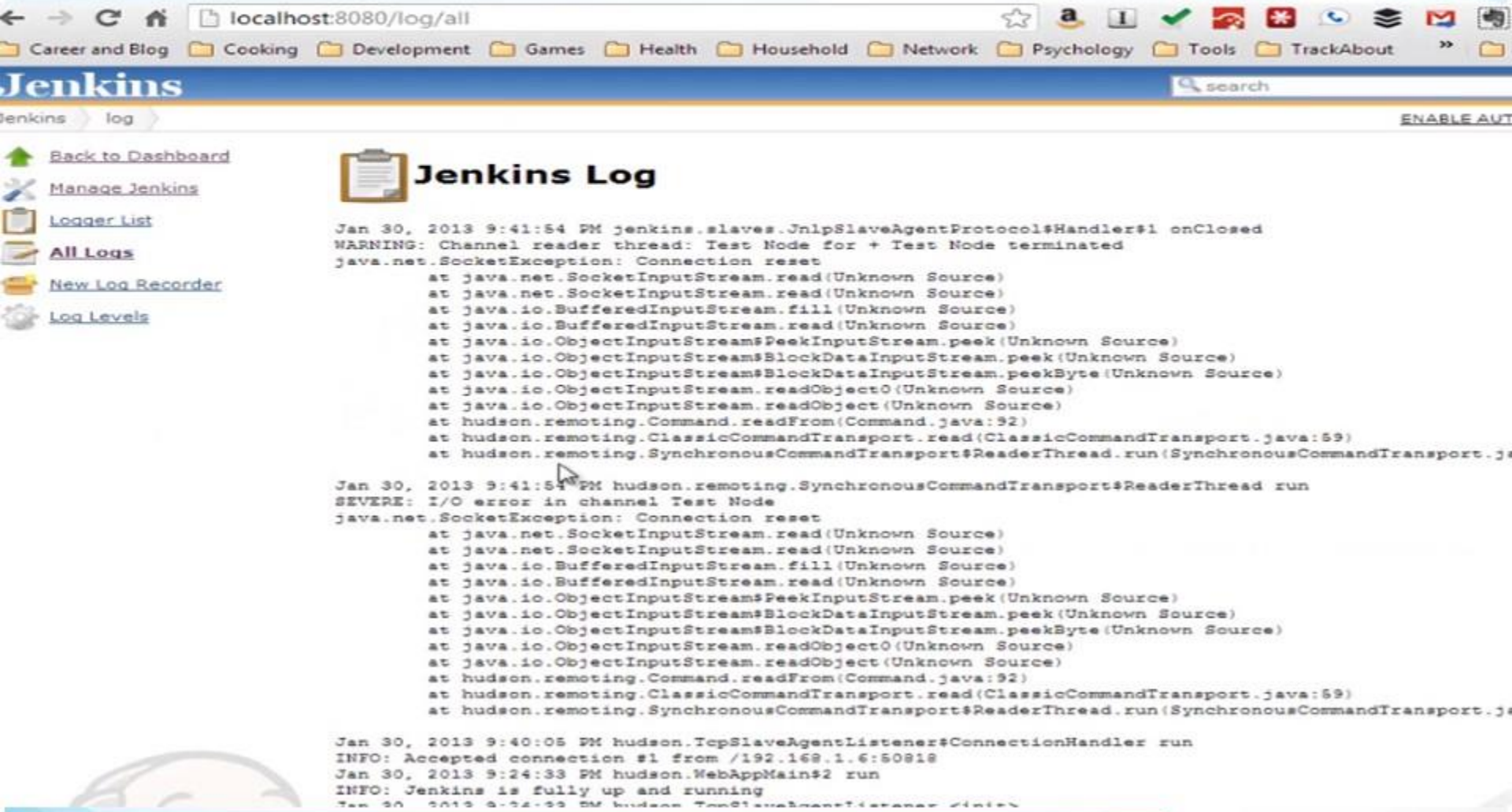
SkillAssure

UI Tour – System Properties



SkillAssure

UI Tour – System Log



SkillAssure

UI Tour – Load Statistics

Career and Blog | Cooking | Development | Games | Health | Household | Network | Psychology | Tools | TrackAbout

Jenkins

search

Jenkins

ENABLE AUTO

- New Job
- People
- Build History
- Manage Jenkins

Build Queue

No builds in the queue.

Build Executor Status

#	Master
1	Idle
2	Idle

Test Node (offline)

Load statistics: Jenkins

Timespan: Short Medium Long

Load statistics keep track of three key metrics of resource utilization:

Total number of executors
For a computer, this is the number of executors that the computer has. For a label, this is the sum of all executors acr computers in this label. For the entire Jenkins, this is the sum of all executors across all computers in this Jenkins insta Other than configuration changes, this value can also change when slaves go offline.

Number of busy executors
This line tracks the number of executors (among the executors counted above) that are carrying out builds. The ratio the total number of executors gives you the resource utilization. If all your executors are busy for a prolonged period consider adding more computers to your Jenkins cluster.

Queue length
This is the number of jobs that are in the build queue, waiting for an available executor (of this computer, of this label,

SkillAssure

UI Tour – Jenkins CLI

Career and Blog | Cooking | Development | Games | Health | Household | Network | Psychology | Tools | TrackAbout

Jenkins

search

Jenkins

New Job
People
Build History
Manage Jenkins

Build Queue

No builds in the queue.

Build Executor Status

#	Master	
1	Idle	
2	Idle	

Test Node (offline)

Jenkins CLI

You can access various features in Jenkins through a command-line tool. See the Wiki for more details of this feature.To get s
download jenkins-cli.jar, and run it as follows:

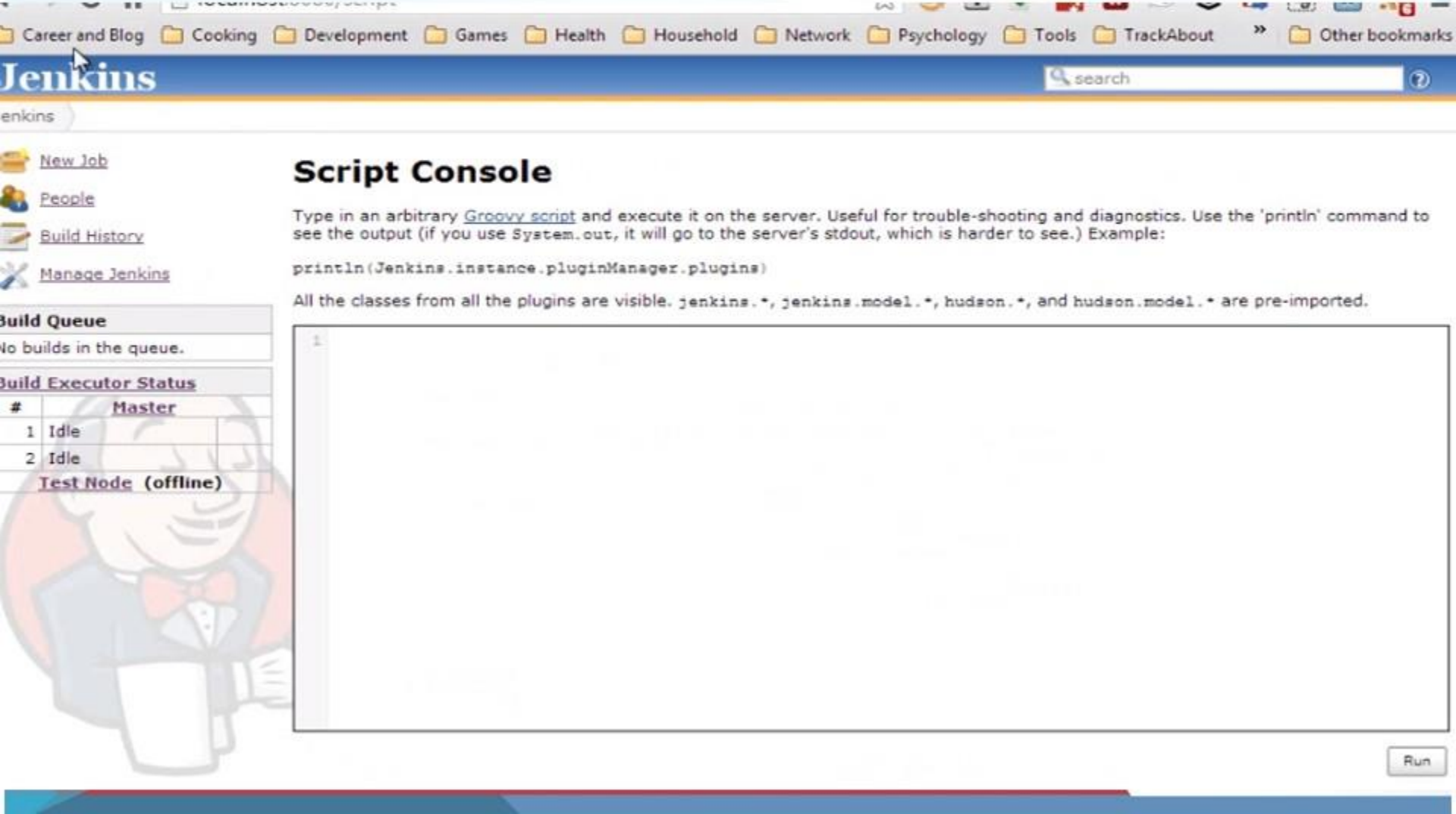
```
java -jar jenkins-cli.jar -s http://localhost:8080/ help
```

Available Commands

- **build**: Builds a job, and optionally waits until its completion.
- **cancel-quiet-down**: Cancel the effect of the "quiet-down" command.
- **clear-queue**: Clears the build queue
- **connect-node**: Reconnect to a node
- **console**: Retrieves console output of a build
- **copy-job**: Copies a job.
- **create-job**: Creates a new job by reading stdin as a configuration XML file.
- **delete-builds**: Deletes build record(s).
- **delete-job**: Deletes a job
- **delete-node**: Deletes a node
- **disable-job**: Disables a job
- **disconnect-node**: Disconnects from a node
- **enable-job**: Enables a job
- **get-job**: Dumps the job definition XML to stdout
- **groovy**: Executes the specified Groovy script.
- **groovysh**: Runs an interactive groovy shell.
- **help**: Lists all the available commands.
- **install-plugin**: Installs a plugin either from a file, an URL, or from update center.
- **install-tool**: Performs automatic tool installation, and print its location to stdout. Can be only called from inside a buil
- **keep-build**: Mark the build to keep the build forever.
- **list-changes**: Dumps the changelog for the specified build(s).
- **list-jobs**: Lists all jobs in a specific view or item group.
- **list-plugins**: Outputs a list of installed plugins.
- **login**: Saves the current credential to allow future commands to run without explicit credential information.
- **logout**: Deletes the credential stored with the login command.
- **mail**: Reads stdin and sends that out as an e-mail.
- **offline-node**: Stop using a node for performing builds temporarily, until the next "online-node" command.
- **online-node**: Resume using a node for performing builds, to cancel out the earlier "offline-node" command.
- **quiet-down**: Quiet down Jenkins, in preparation for a restart. Don't start any builds.
- **reload-configuration**: Discard all the loaded data in memory and reload everything from file system. Useful when

SkillAssure

UI Tour – Script Console



SkillAssure

UI Tour – Script Console

Get-current-queue-state

```
/*** BEGIN META {
 "name" : "Get Queue State",
 "comment" : "Get current Queue/Executors state. Useful to troubleshoot queue and check that
tasks are correctly handled",
 "parameters" : [ ],
 "core": "1.642",
 "authors" : [
 { name : "Allan Burdajewicz" }
 ]
 } END META**/

def all = [ Jenkins.instance ]
all.addAll(Jenkins.instance.nodes)
all.each {
    def c = it.toComputer()
    println "[$it.nodeName] - $it.numExecutors - $it.assignedLabels - $it.acceptingTasks -
$it.nodeProperties - $c.offline"
}
println "----"
Jenkins.instance.queue.items.each {
    println "$it.id $it.blocked $it.buildable $it.stuck $it.assignedLabel $it.causes"
}
```

SkillAssure

UI Tour – About Jenkins

Jenkins

search

Jenkins › About Jenkins

About Jenkins 1.500

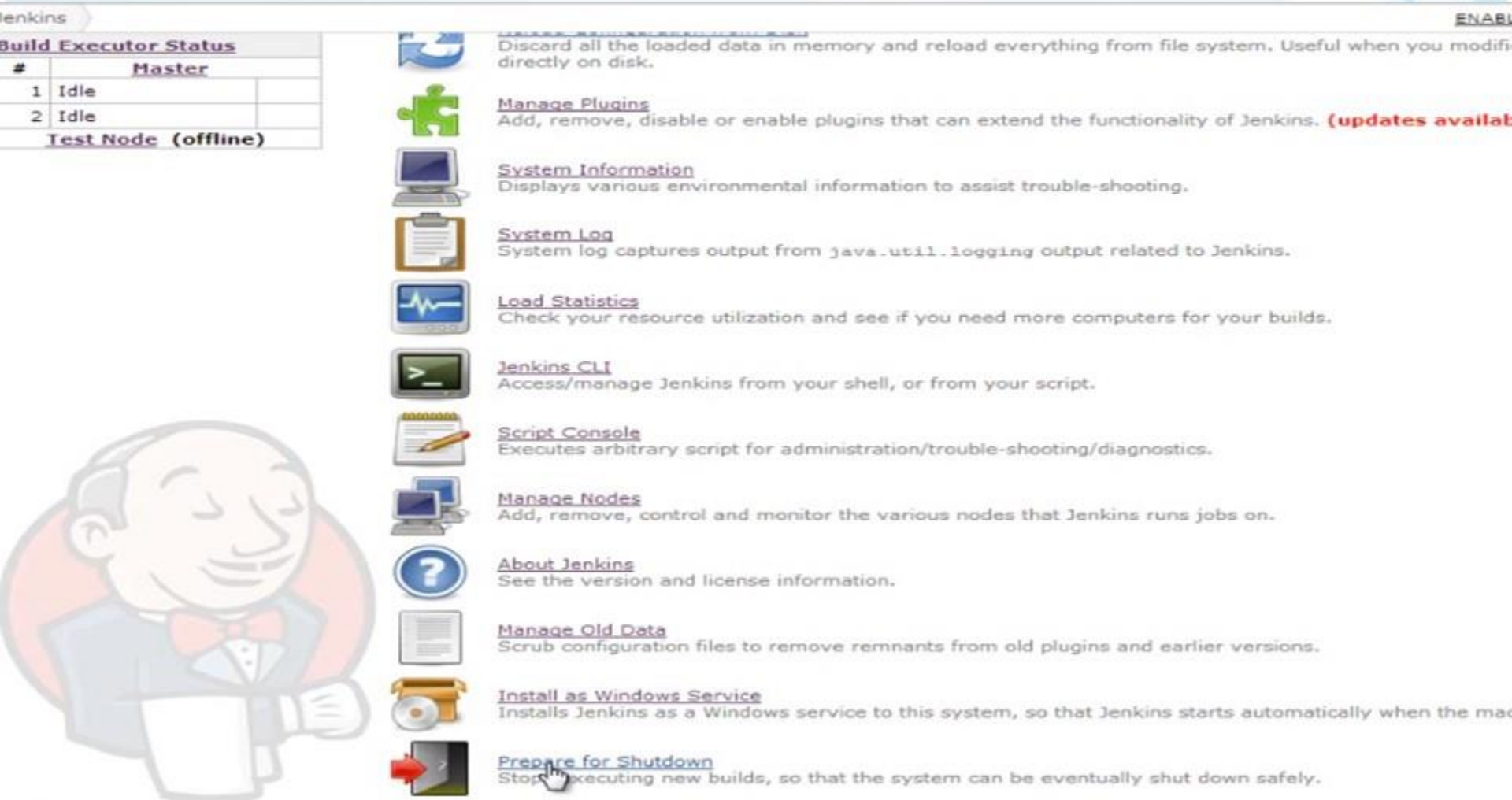
Jenkins is a community-developed open-source continuous integration server.

Jenkins depends on the following 3rd party libraries.

Name	Maven ID	
Jenkins war	org.jenkins-ci.main:jenkins-war:1.500	The MIT licen
Commons BeanUtils	commons-beanutils:commons-beanutils:1.8.3	The Apache S
Apache Ant Launcher	org.apache.ant:ant-launcher:1.8.3	The Apache S
Bouncy Castle Provider	org.bouncycastle:bcprov-jdk15on:1.47	Bouncy Castl
oro	oro:oro:2.0.8	The Apache S
Jenkins CLI	org.jenkins-ci.main:cli:1.500	The MIT licen
StAX API	stax:stax-api:1.0.1	The Apache S
jinterop-proxy	org.kohsuke.jinterop:jinterop-proxy:1.1	MIT license
Stapler	org.kohsuke.stapler:stapler:1.199	2-clause BSD
SSH CLI client authenticator	org.jenkins-ci.modules:ssh-cli-auth:1.2	MIT License
memory-monitor	org.jenkins-ci:memory-monitor:1.7	MIT License
commons-jelly-tags-xml	commons-jelly:commons-jelly-tags-xml:1.1	The Apache S
CodeMirror library from http://codemirror.net/	org.kohsuke.stapler:stapler-adjunct-codemirror:1.1	MIT License
Spring Framework: DAO	org.springframework:spring-dao:1.2.9	The Apache S
JLine	jline:jline:0.9.94	BSD
Java binding for libpam.so	org.kohsuke:libpam4j:1.6	The MIT licen
Spring Framework: Context	org.springframework:spring-context:2.5	The Apache S
HttpClient	commons-httpclient:commons-httpclient:3.1	Apache Licen
Woodstox	org.codehaus.woodstox:wstx-asl:3.2.9	The Apache S
jbcrypt	org.mindrot:jbcrypt:0.3m	ISC/BSD Lic
Apache Ant Core	org.apache.ant:ant:1.8.3	The Apache S
libzfs-java	org.jvnet.libzfs:libzfs:0.5	COMMON DE
Ganymed SSH2 for Java	org.jenkins-ci:trilead-ssh2:build214-jenkins-1	BSD style lic
Stapler Jelly module	org.kohsuke.stapler:stapler-jelly:1.199	2-clause BSD

SkillAssure

UI Tour

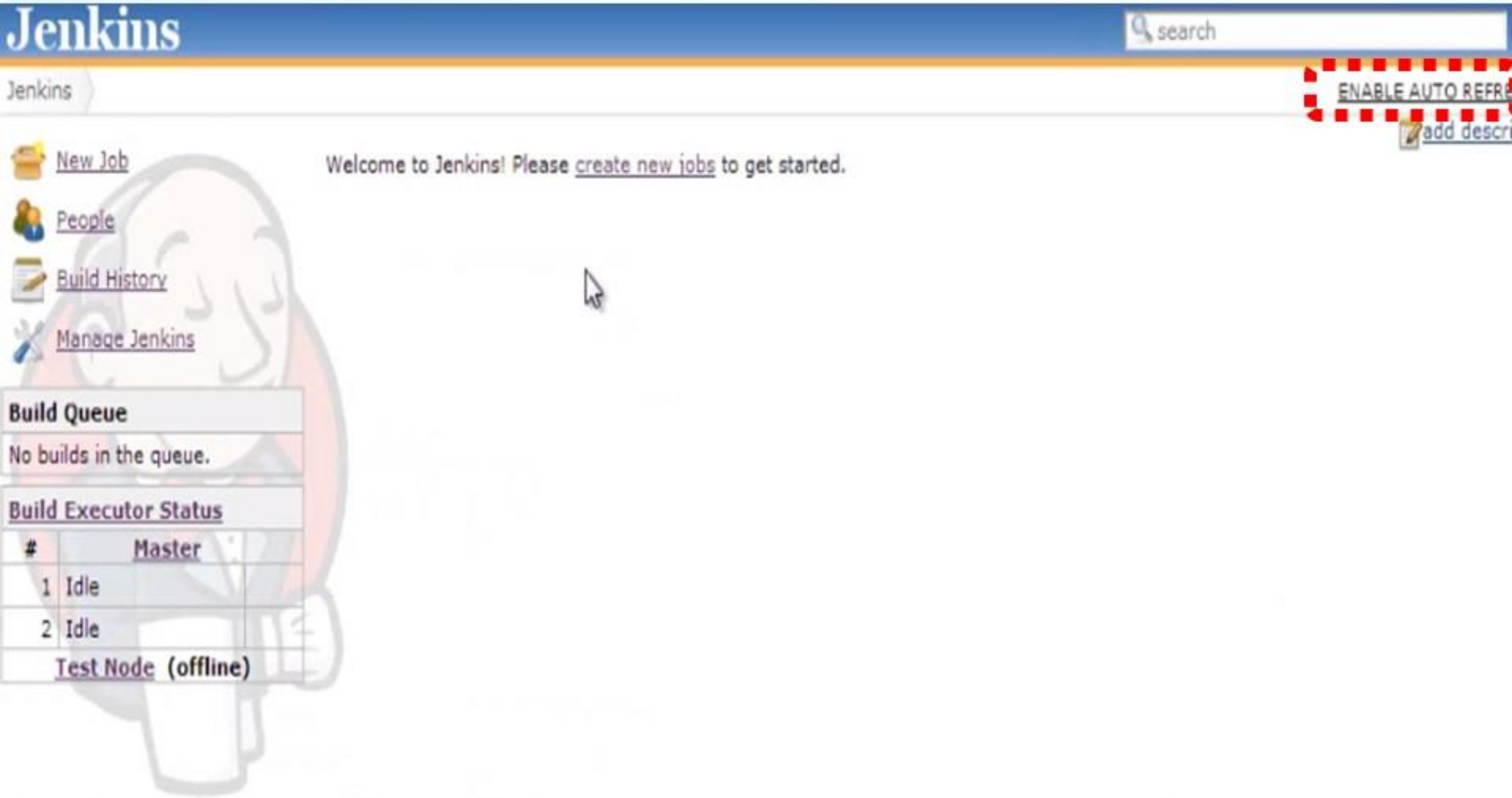


SkillAssure

UI Tour – Build Queue

SkillAssure

UI Tour – Home Page Auto Refresh

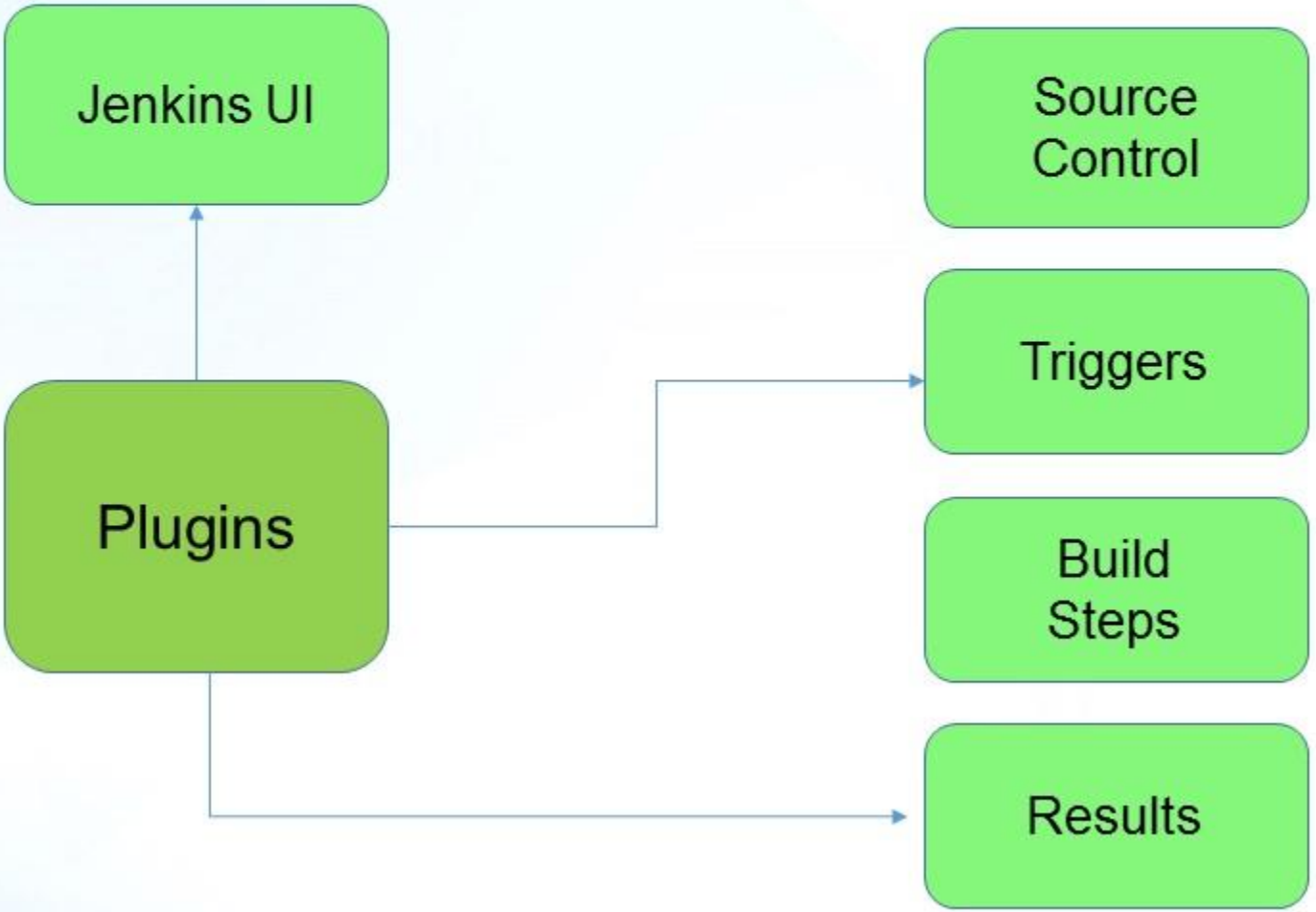


SkillAssure

Jenkins Plugins

SkillAssure

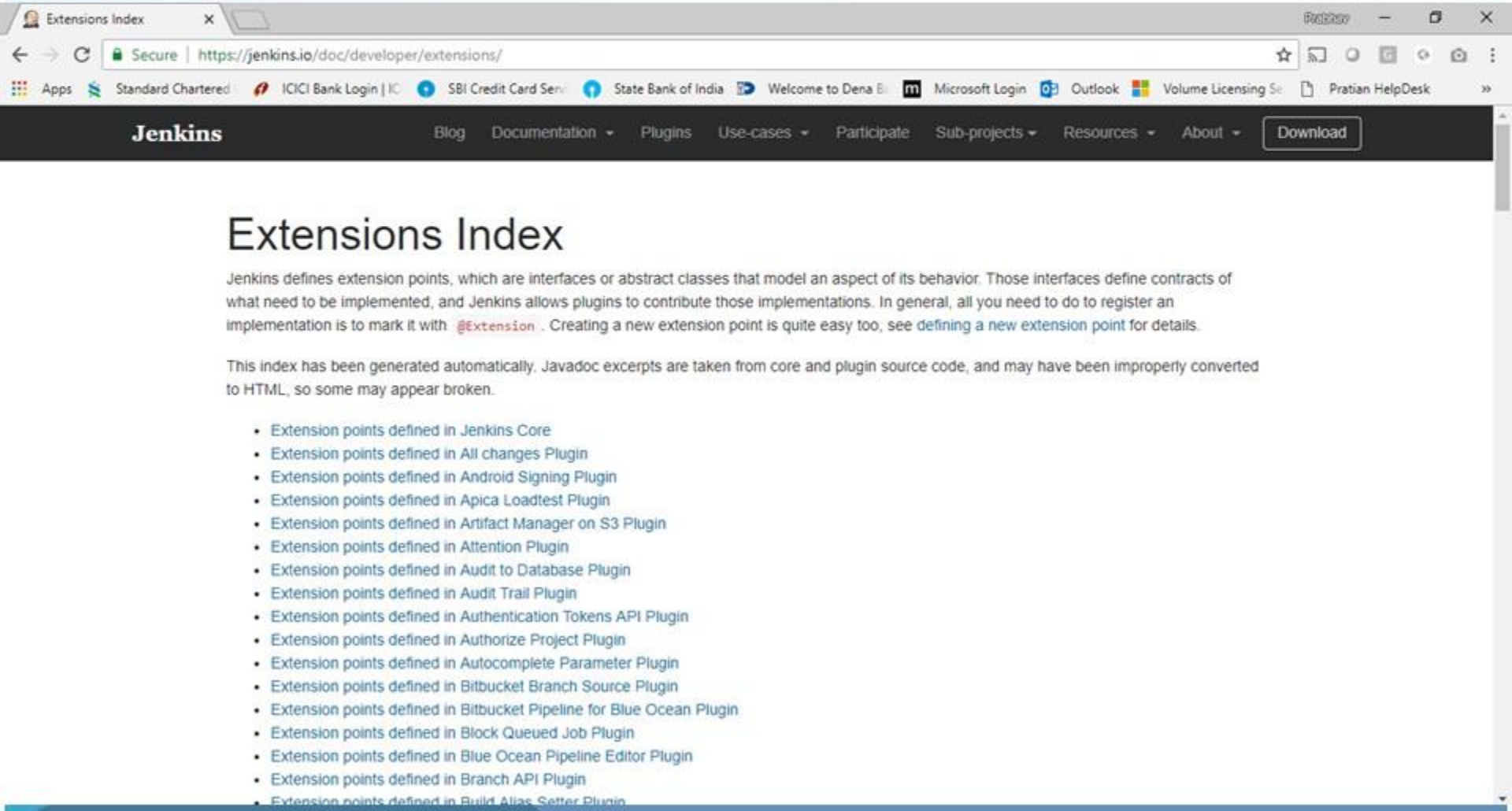
Plugin Architecture



SkillAssure

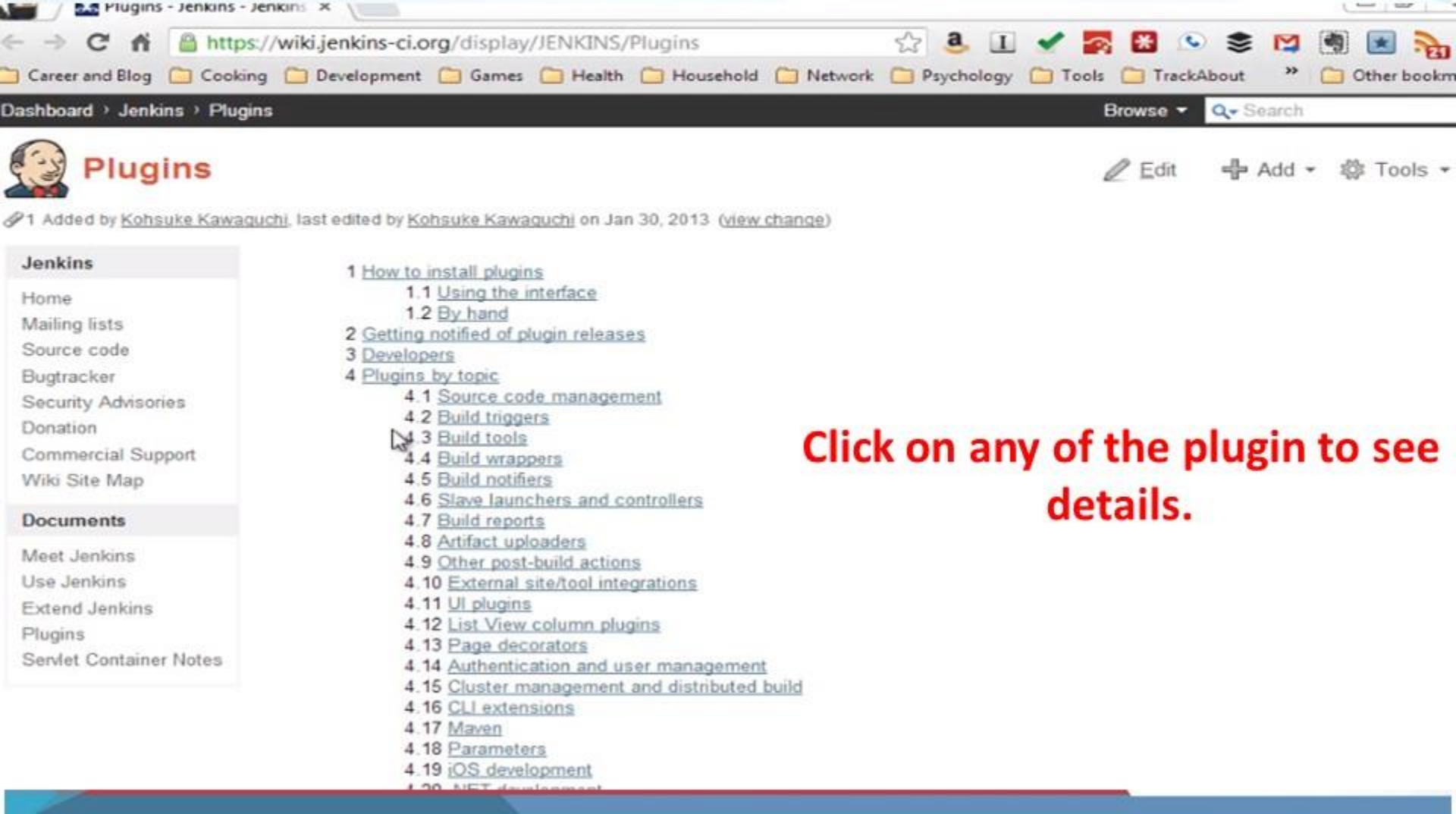
Extension Points

https://jenkins.io/doc/developer/extensions/



SkillAssure

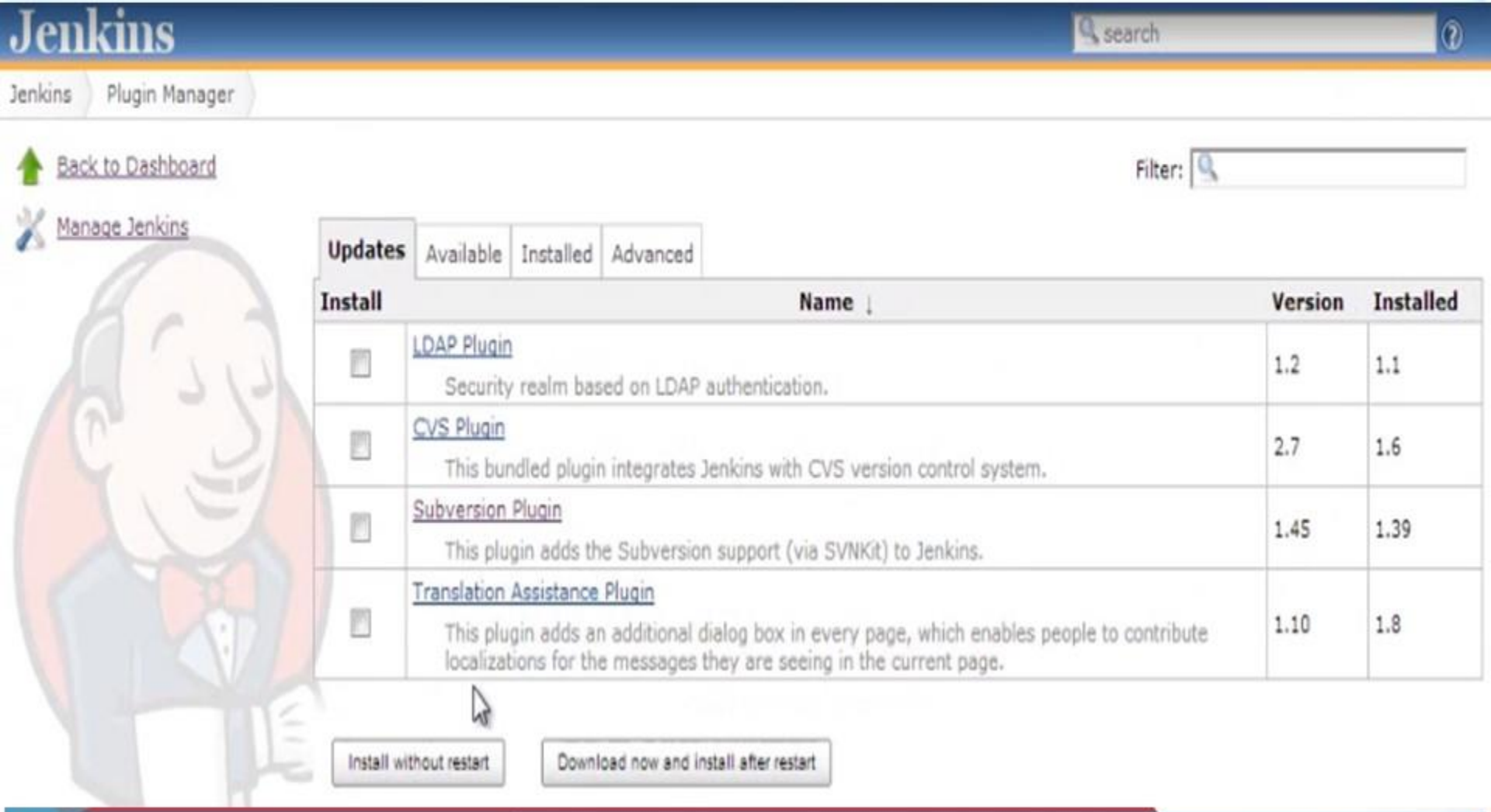
Plugin Wiki



Click on any of the plugin to see details.

SkillAssure

Getting Plugins



SkillAssure

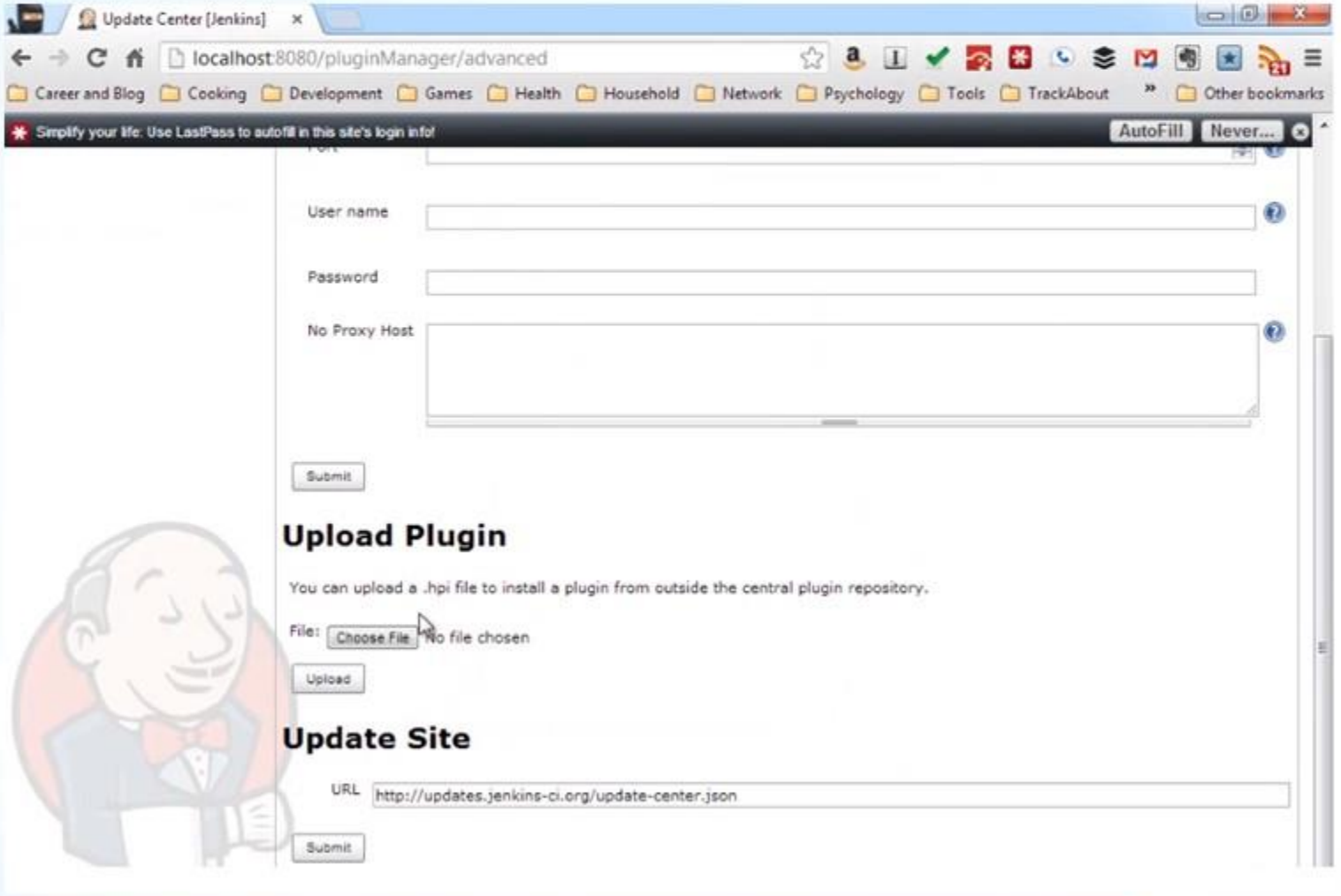
Getting Plugins

Updates | **Available** | Installed | Advanced

Install	Name	Vers
Artifact Uploaders		
Authentication and User Management		
Build Notifiers		
Build Reports		
Build Tools		
Build Triggers		
Build Wrappers		
Cluster Management and Distributed Build		
Command Line Interface		
External Site/Tool Integrations		
List view columns		
Maven		
Misc (android)		
☐	Android Emulator Plugin Lets you automatically generate, launch and interact with an Android emulator during a build, with the emulator logs being captured as artifacts.	2.8
☐	Android Lint Plugin Parses output from the Android lint tool and displays the results for analysis.	2.0.2
☐	Appaloosa Plugin Publish your mobile applications (Android, iOS, ...) to the appaloosa-store.com platform.	1.3.1
☐	External Resource Dispatcher This plugin adds support for external resources in Jenkins. An external resource is something external attached to a Jenkins slave and can be locked by a build, which thus gets exclusive access to it, then	1.0b

SkillAssure

Upload Plugin



SkillAssure

Useful Plugins Overview

SkillAssure

Source Code Plugins

- Git
- Mecurial
- BitBucket
- TFS

SkillAssure

Triggers Plugins

- Github Pull Request
- Join
- Locks & Latches

SkillAssure

Build Tool Plugins

- Copy artifacts
- Fitness
- MSBuild
- MSTest Runner
- Nant
- PowerShell
- Promoted Builds

SkillAssure

Wrapper Plugins

- Selenium RC
- VMWare
- VirtualBox

SkillAssure

Notifier Plugins

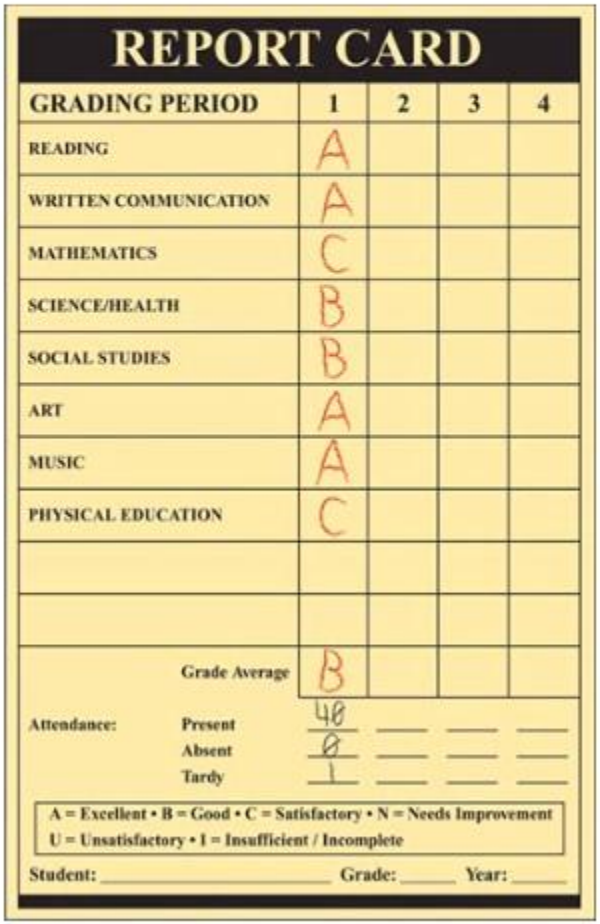
- Hipchat
- IRC
- Twitter
- Jabber

Calling to let you know, you broke the build!

SkillAssure

Reporting Plugins

- Cobertura – Java Code Coverage
- Findbugs – Java Static Analysis
- MSTest - MSTest reporting
- Ncover - .Net code coverage
- PMD – Static analysis Java code



REPORT CARD

GRADING PERIOD	1	2	3	4
READING	A			
WRITTEN COMMUNICATION	A			
MATHEMATICS	C			
SCIENCE/HEALTH	B			
SOCIAL STUDIES	B			
ART	A			
MUSIC	A			
PHYSICAL EDUCATION	C			
Grade Average	B			
Attendance: Present	40			
Absent	0			
Tardy	1			

A = Excellent • B = Good • C = Satisfactory • N = Needs Improvement
U = Unsatisfactory • I = Insufficient / Incomplete

Student: _____ Grade: _____ Year: _____

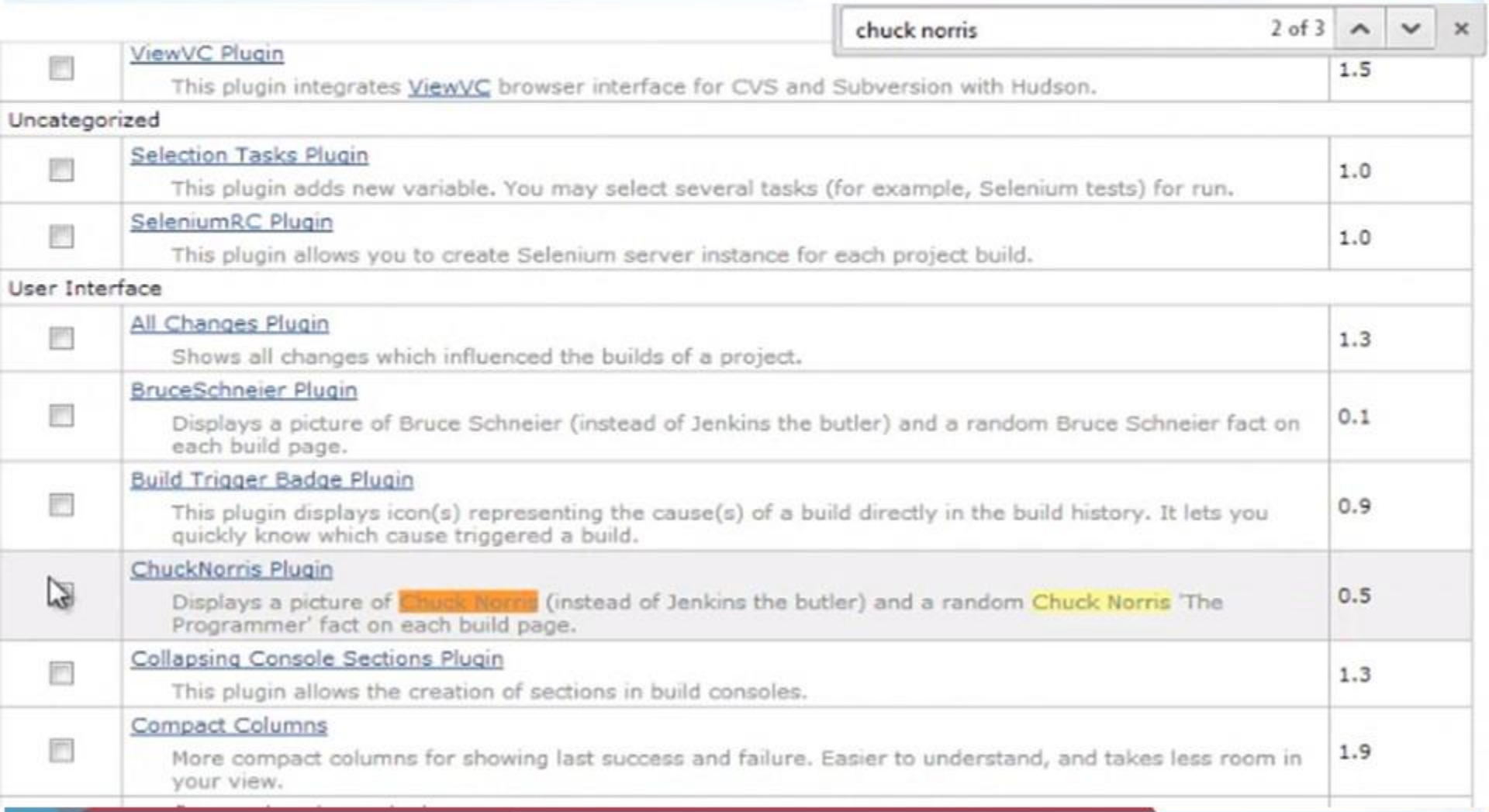
SkillAssure

Artifacts / UI

- Artifact deployer
- Deploy
- Chunk Norris
- Emotional Jenkins
- Claim
- CI Game

SkillAssure

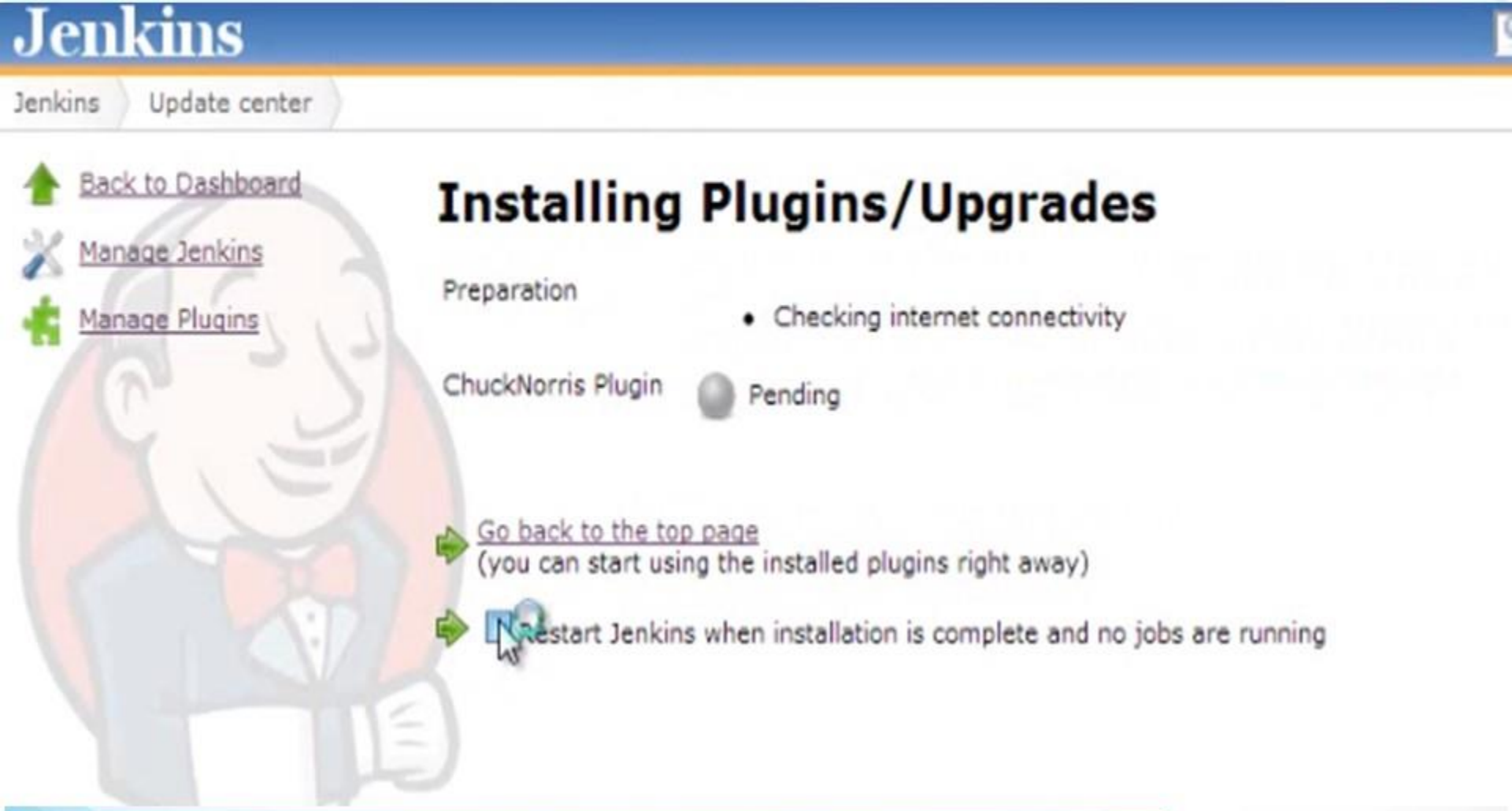
Installing a Plugin



☐	ViewVC Plugin This plugin integrates ViewVC browser interface for CVS and Subversion with Hudson.	1.5
Uncategorized		
☐	Selection Tasks Plugin This plugin adds new variable. You may select several tasks (for example, Selenium tests) for run.	1.0
☐	SeleniumRC Plugin This plugin allows you to create Selenium server instance for each project build.	1.0
User Interface		
☐	All Changes Plugin Shows all changes which influenced the builds of a project.	1.3
☐	BruceSchneier Plugin Displays a picture of Bruce Schneier (instead of Jenkins the butler) and a random Bruce Schneier fact on each build page.	0.1
☐	Build Trigger Badge Plugin This plugin displays icon(s) representing the cause(s) of a build directly in the build history. It lets you quickly know which cause triggered a build.	0.9
☐	ChuckNorris Plugin Displays a picture of Chuck Norris (instead of Jenkins the butler) and a random Chuck Norris 'The Programmer' fact on each build page.	0.5
☐	Collapsing Console Sections Plugin This plugin allows the creation of sections in build consoles.	1.3
☐	Compact Columns More compact columns for showing last success and failure. Easier to understand, and takes less room in your view.	1.9

SkillAssure

Install Plugins



SkillAssure

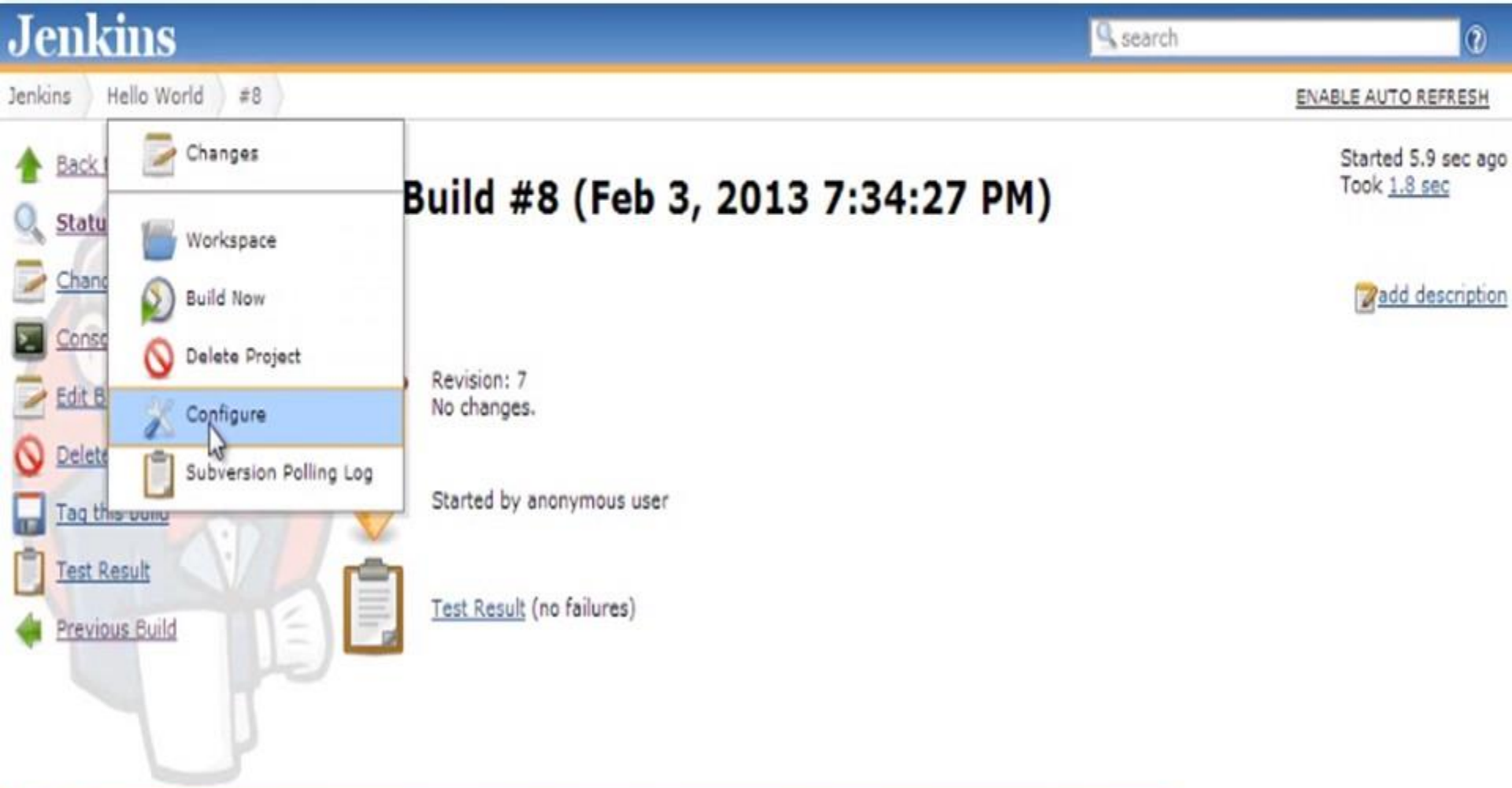
Install Plugins

Updates | Available | **Installed** | Advanced

Enabled	Name ↓	Version	Previously installed version	Pinned	Uninstall
☑	Jenkins Mailer Plugin	1.4			
☑	External Monitor Job Type Plugin	1.1			
☑	LDAP Plugin	1.1			
☑	pam-auth	1.0			
☑	ant	1.1			
☑	ChuckNorris Plugin ChuckNorris plugin displays a picture of Chuck Norris (instead of Jenkins the butler) and a random Chuck Norris 'The Programmer' fact on each build page.	0.5			Uninstall
☑	javadoc	1.0			
☑	CVS Plugin	1.6			

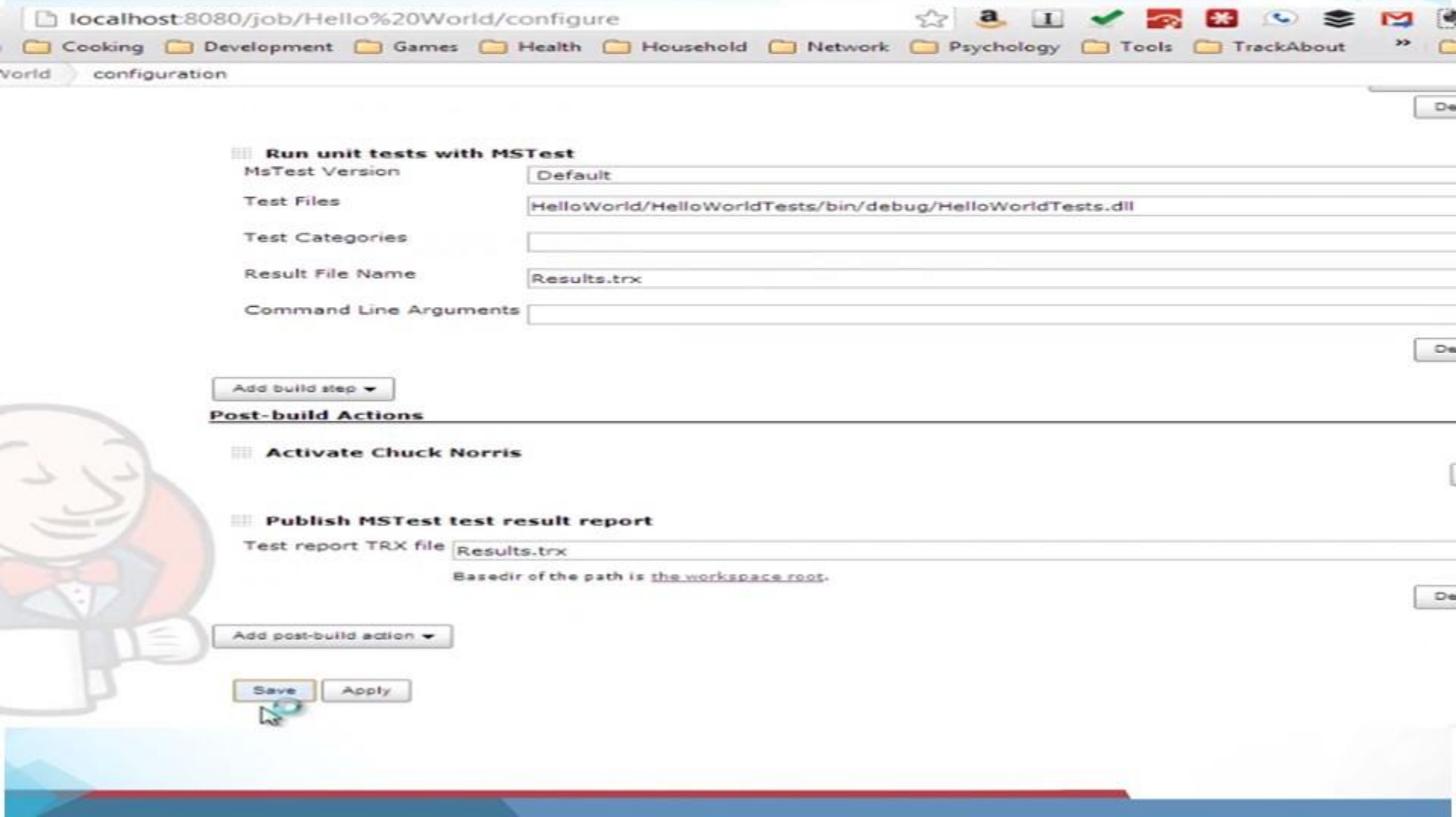
SkillAssure

Installing a Plugin



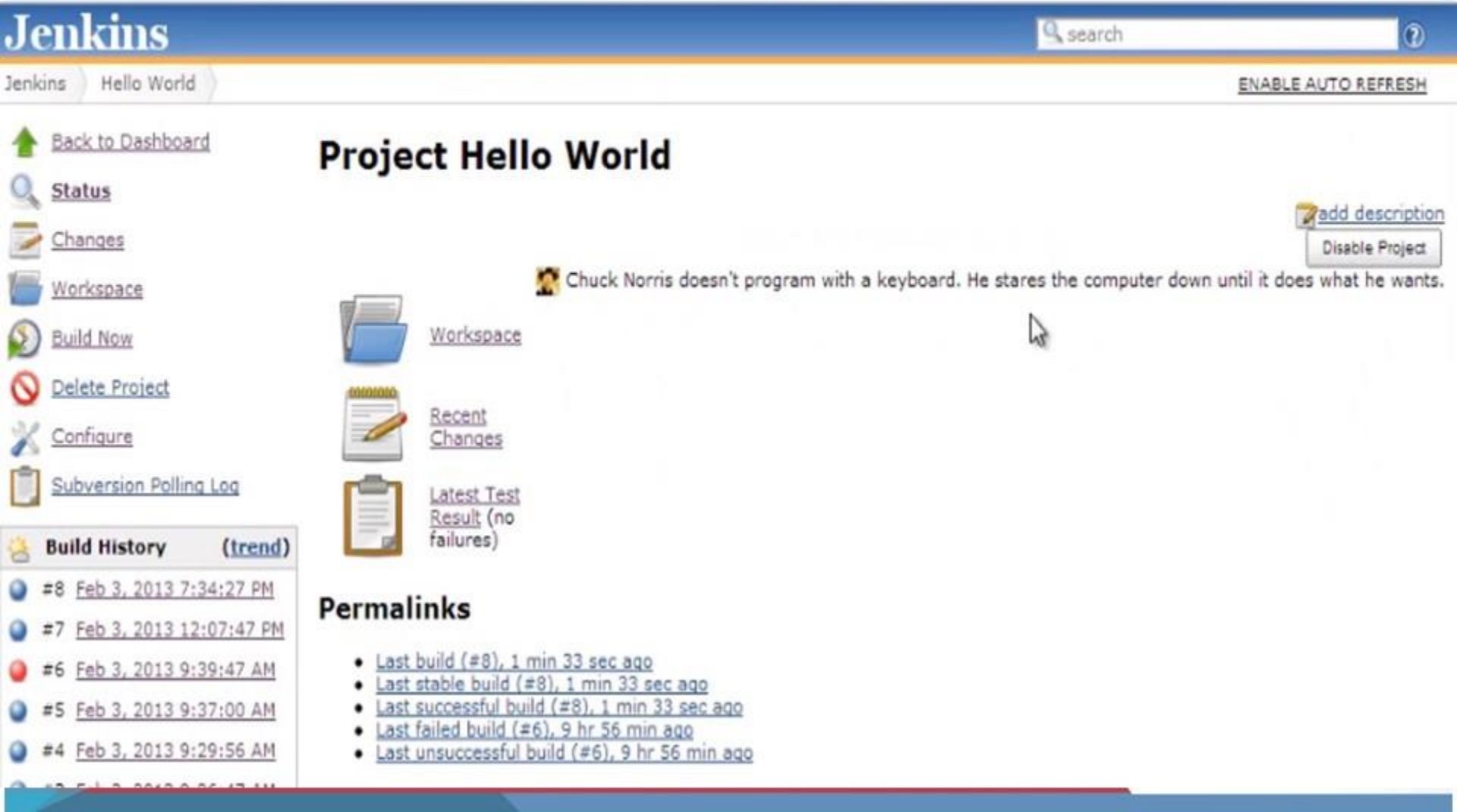
SkillAssure

Installing a Plugins



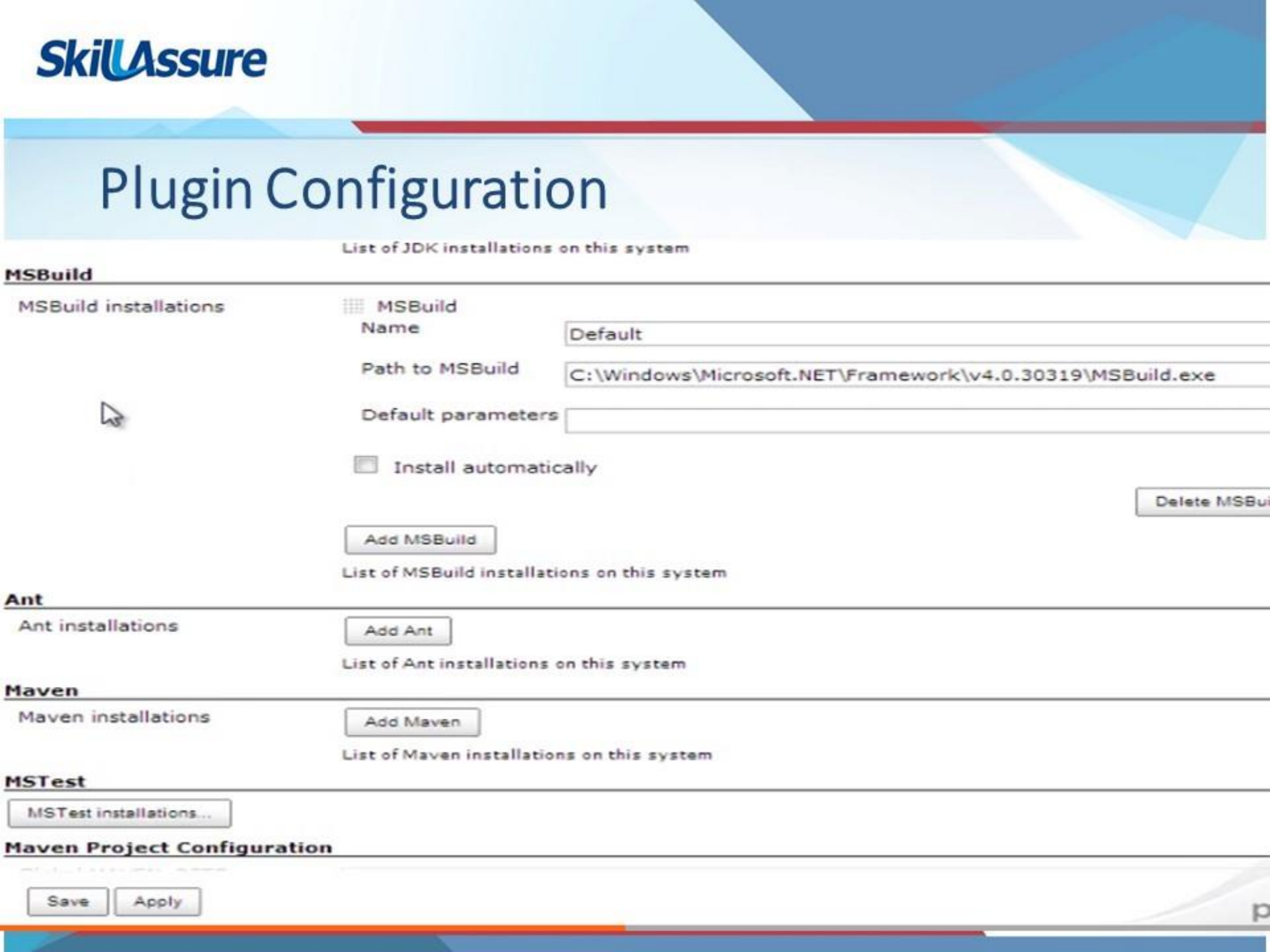
SkillAssure

Installing a Plugins



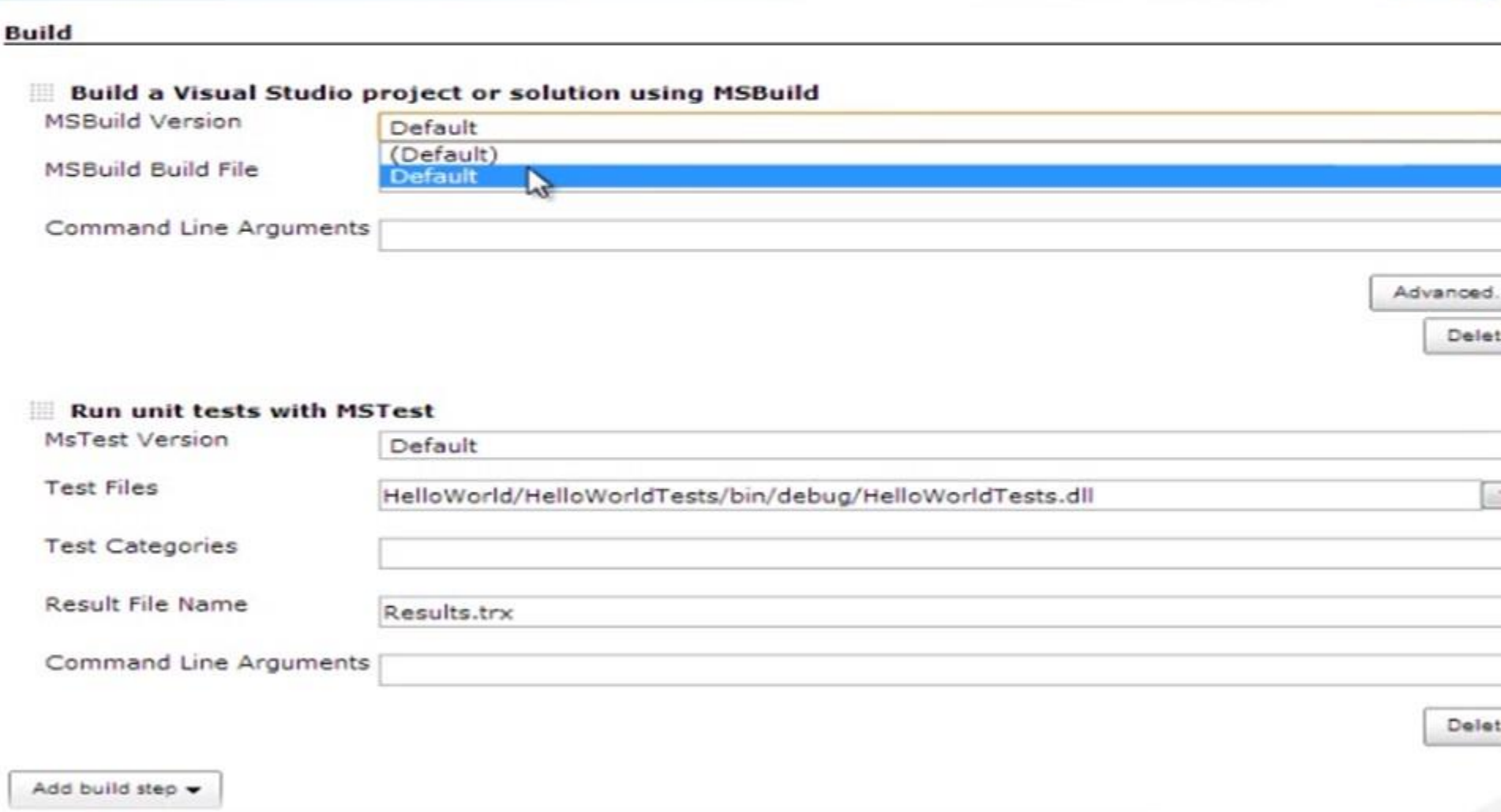
SkillAssure

Plugin Configuration



SkillAssure

Plugin Configuration



SkillAssure

Security Overview

- Go to Manage Jenkins
- Go to Global Security
- Check Enable Security

SkillAssure

Security Overview

Configure Global Security

Enable security

TCP port for JNLP slave agents: Fixed : Random Disable

Markup Formatter: Raw HTML

Treat the text as HTML and use it as is without any translation

Disable syntax highlighting

Access Control

Security Realm

- Delegate to servlet container
- Jenkins's own user database
  - Allow users to sign up
- LDAP

Authorization

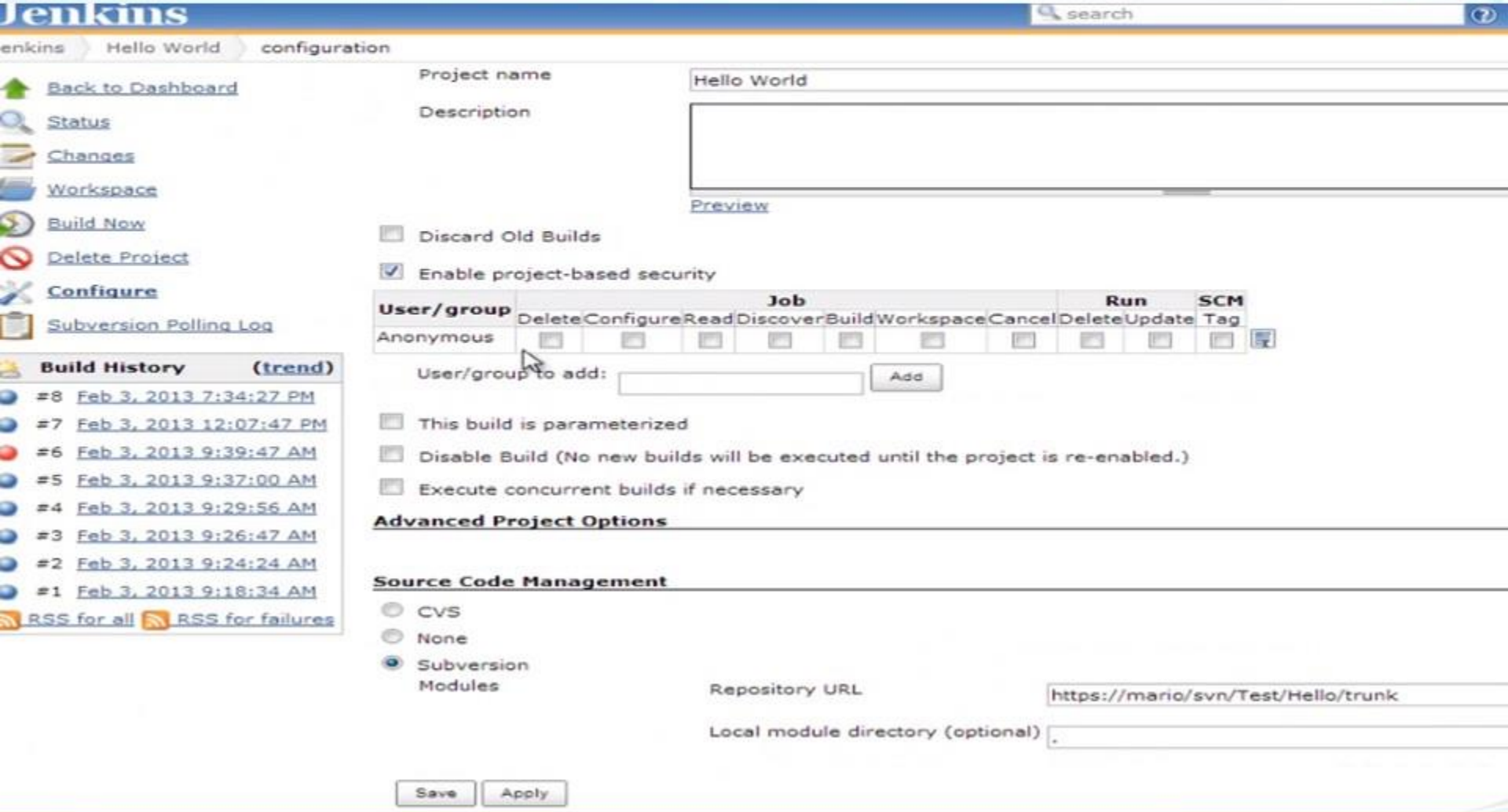
- Anyone can do anything
- Legacy mode
- Logged-in users can do anything
- Matrix-based security
- Project-based Matrix Authorization Strategy

User/group	Overall						Slave				
	Administer	Read	RunScripts	UploadPlugins	ConfigureUpdateCenter		Configure	Delete	Create	Disconnect	Connect
Anonymous											

User/group to add: Add

SkillAssure

Security Overview



SkillAssure

Exercise

- Create a Jenkins Job.
- Restart Jenkins Server.

SkillAssure

Exercise

- **LDAP User**
  - Integrate Jenkins with LDAP.
  - Create two users in LDAP (e.g. adminJ, readJ)
  - Give administrator access to **adminJ** and read only access to **readJ**.
- **Jenkins Security**
  - Manage Jenkins
  - Configure Global Security
  - Under Authorization select "Matrix-based security"
  - Click Add User/Group.
  - Provide User or group name as **Pratian**.
  - Check Administer permission for **Pratian**
  - Uncheck Administer permission for **adminJ.**
- **Switch back to Jenkins User Authentication**
  - Create a Jenkins user.
  - Give read only access to that user.

Did you just lock out yourself in Jenkins.

How will you fix it?

SkillAssure

Exercise

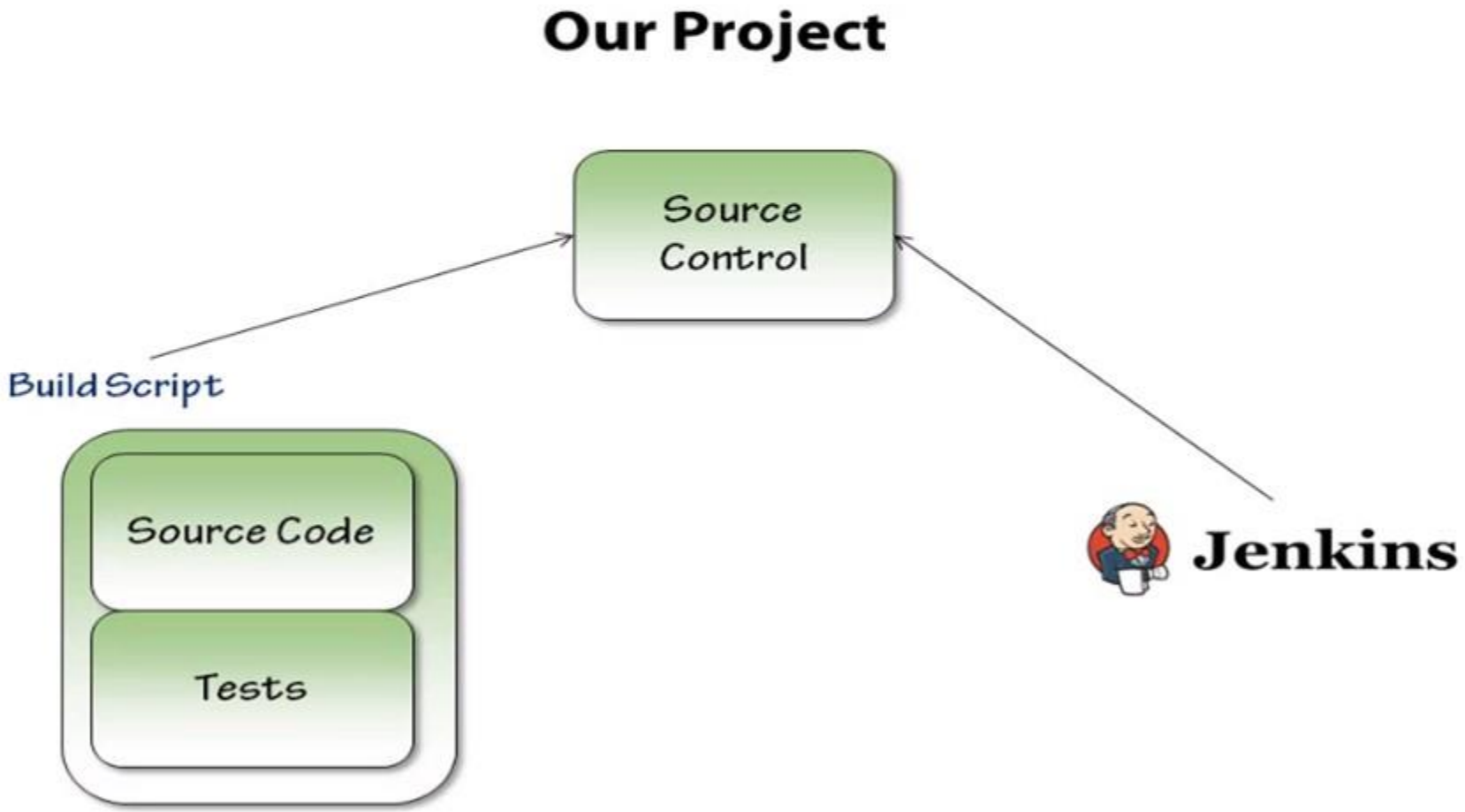
- Change the Jenkins status balls from “BLUE” to “GREEN” for successful builds.
- Integrate MSBuild with Jenkins.

SkillAssure

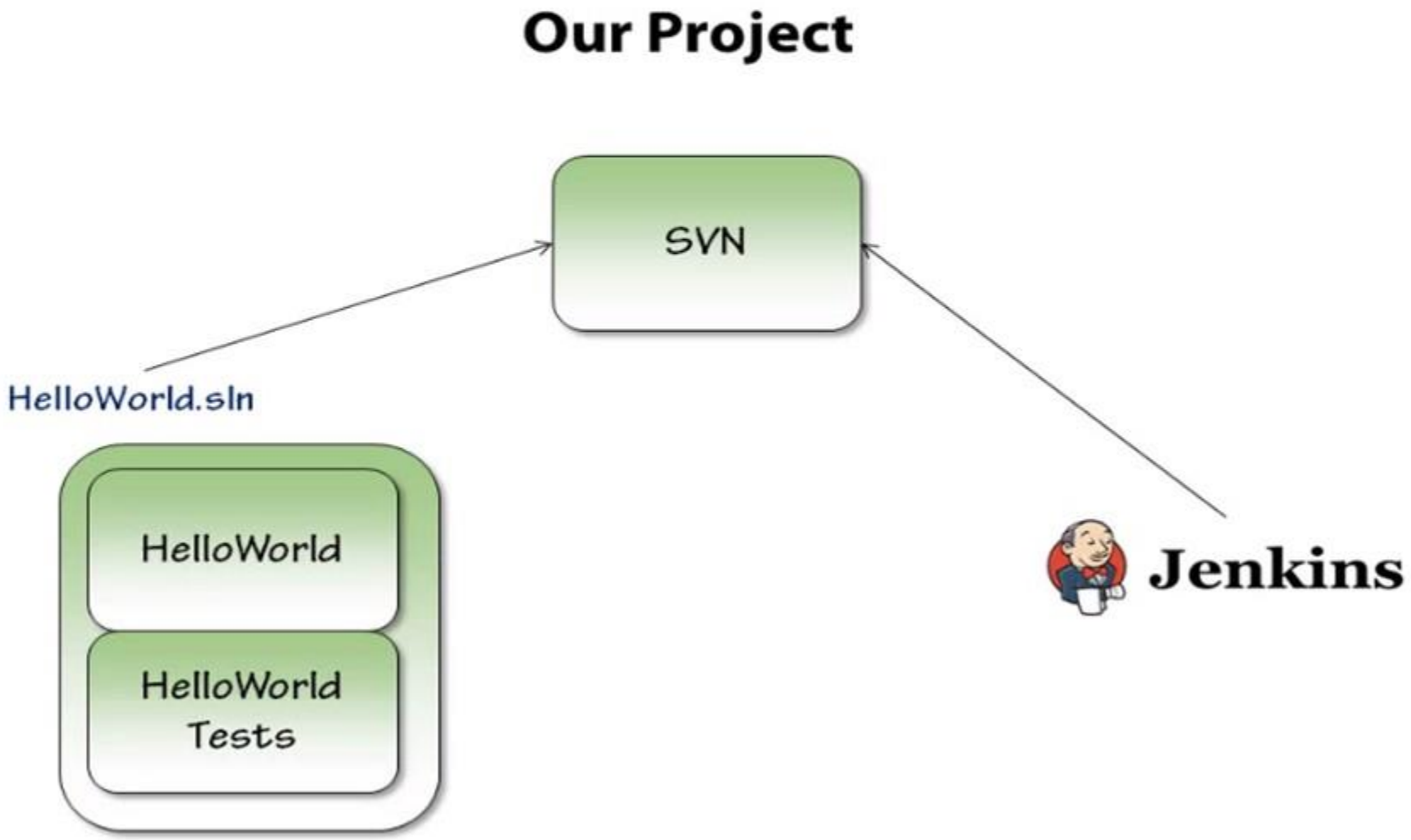
Creating a Simple Job

SkillAssure

Basic Project

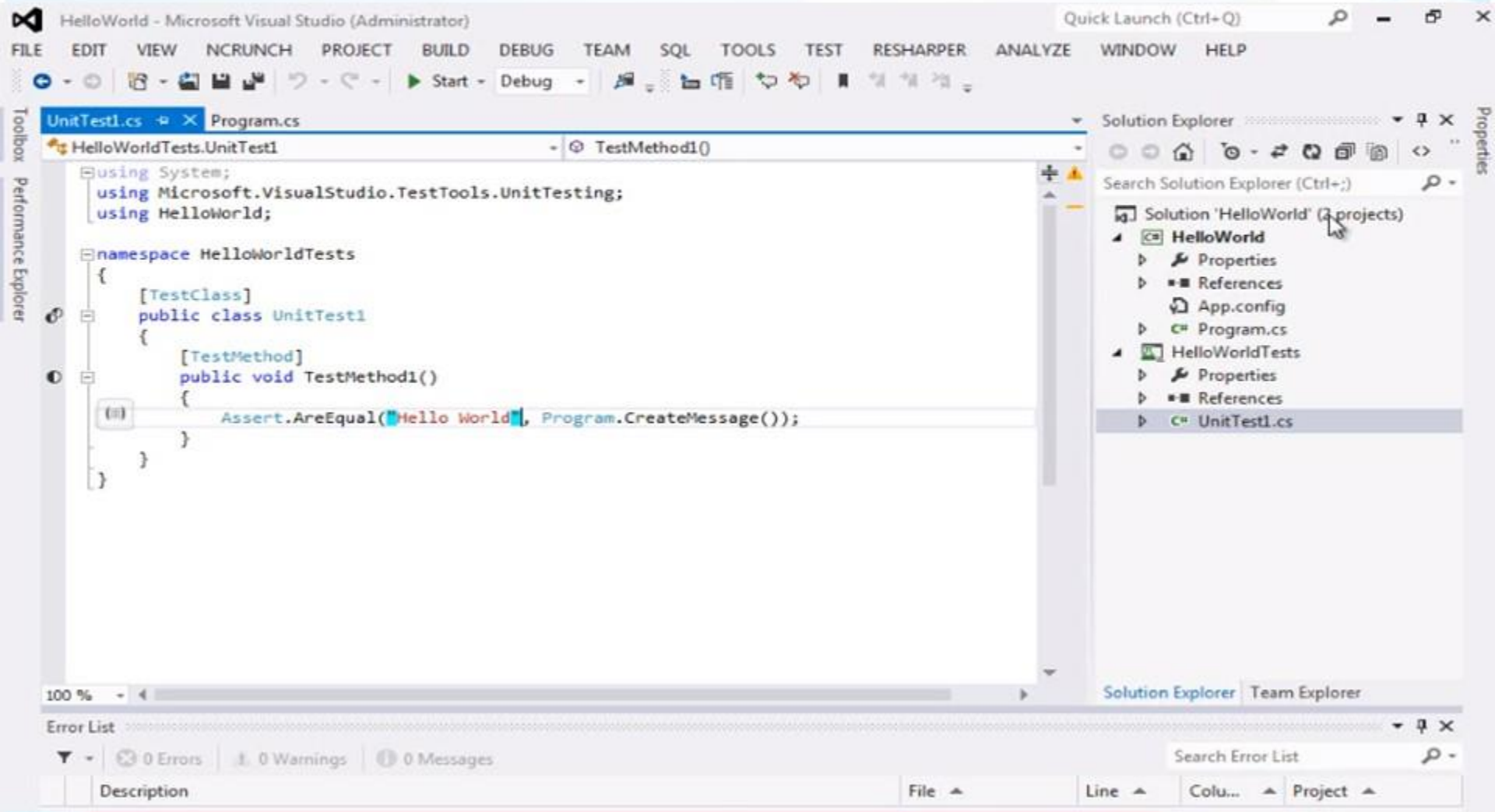


Our Project


Our Project
SVN
HelloWorld.sln
HelloWorld
HelloWorld Tests
Jenkins

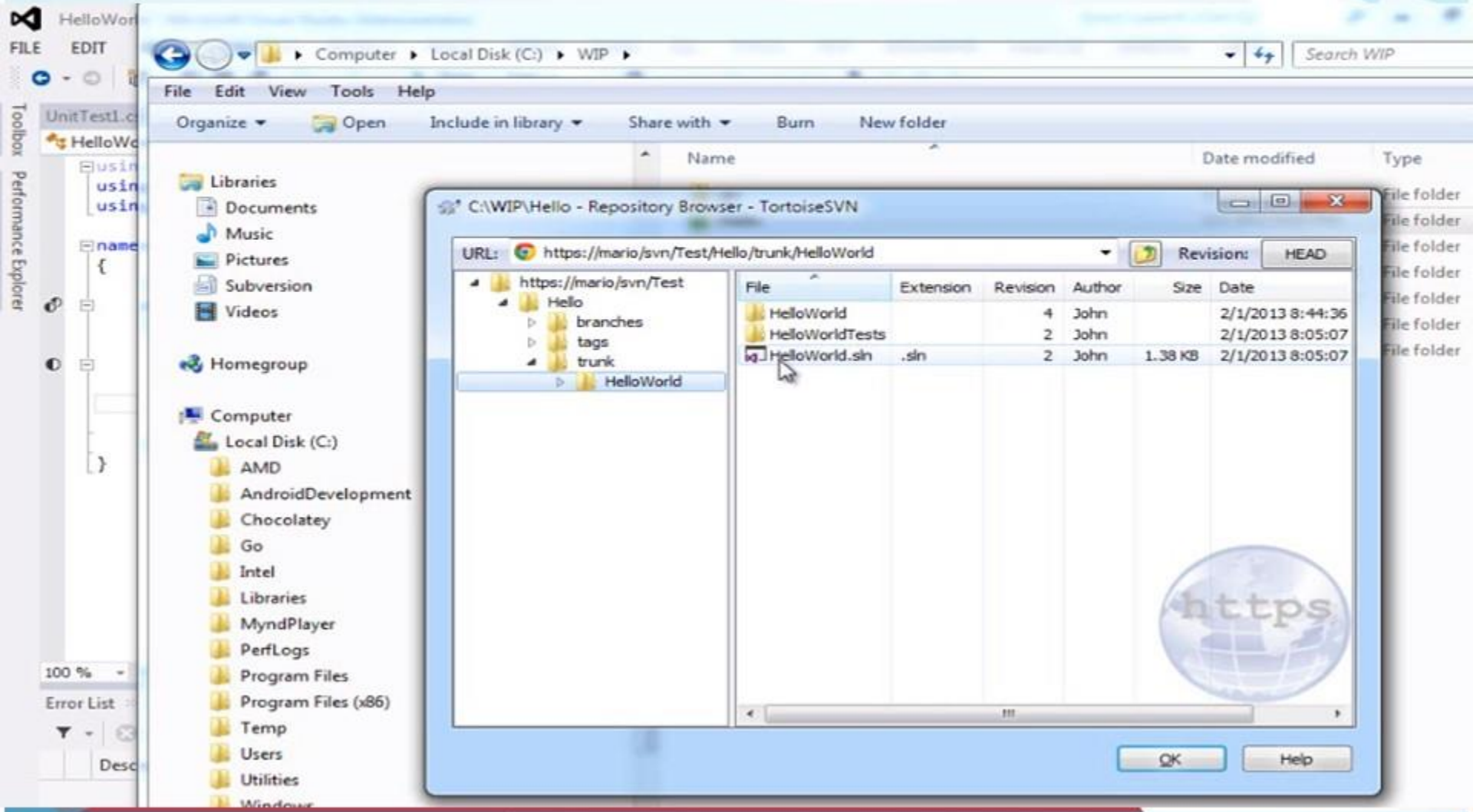
SkillAssure

Looking at the Code



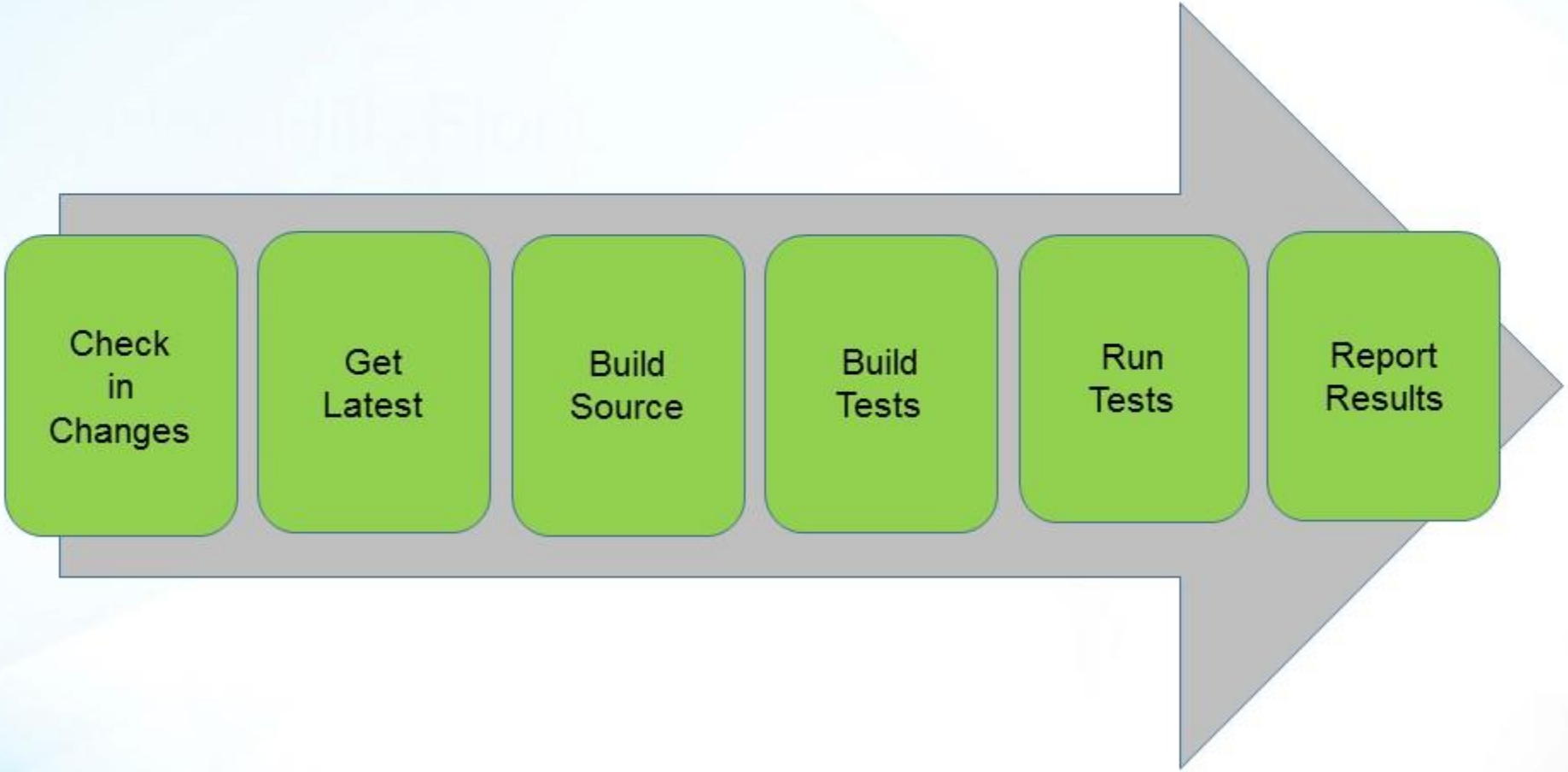
SkillAssure

Looking at the Code



SkillAssure

Moving to Jenkins



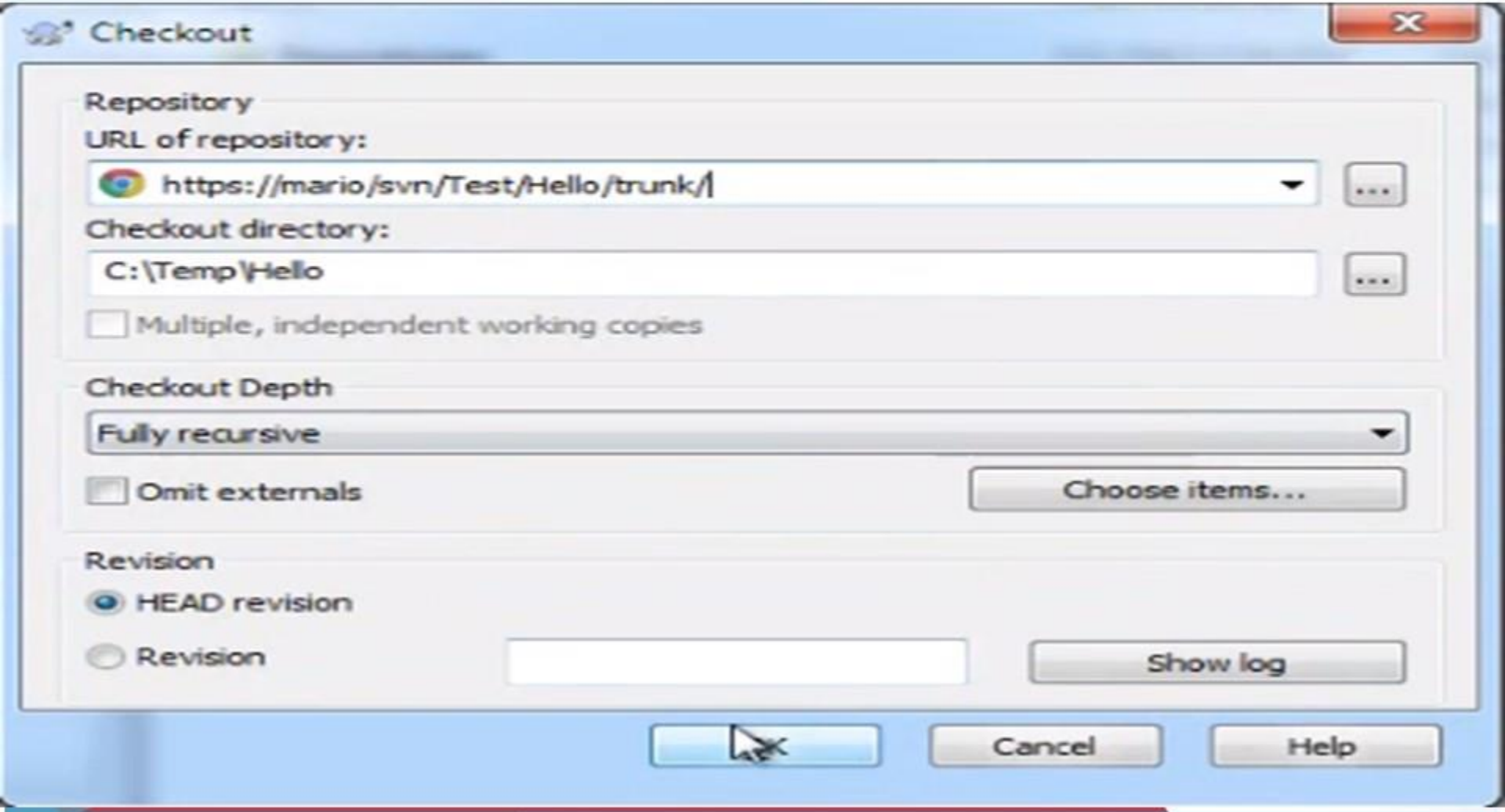
SkillAssure

Moving to Jenkins

1. Diagram manual process
2. Follow manual steps
3. Get most basic automation working
4. Add each step
5. Never leave the build in a failed state

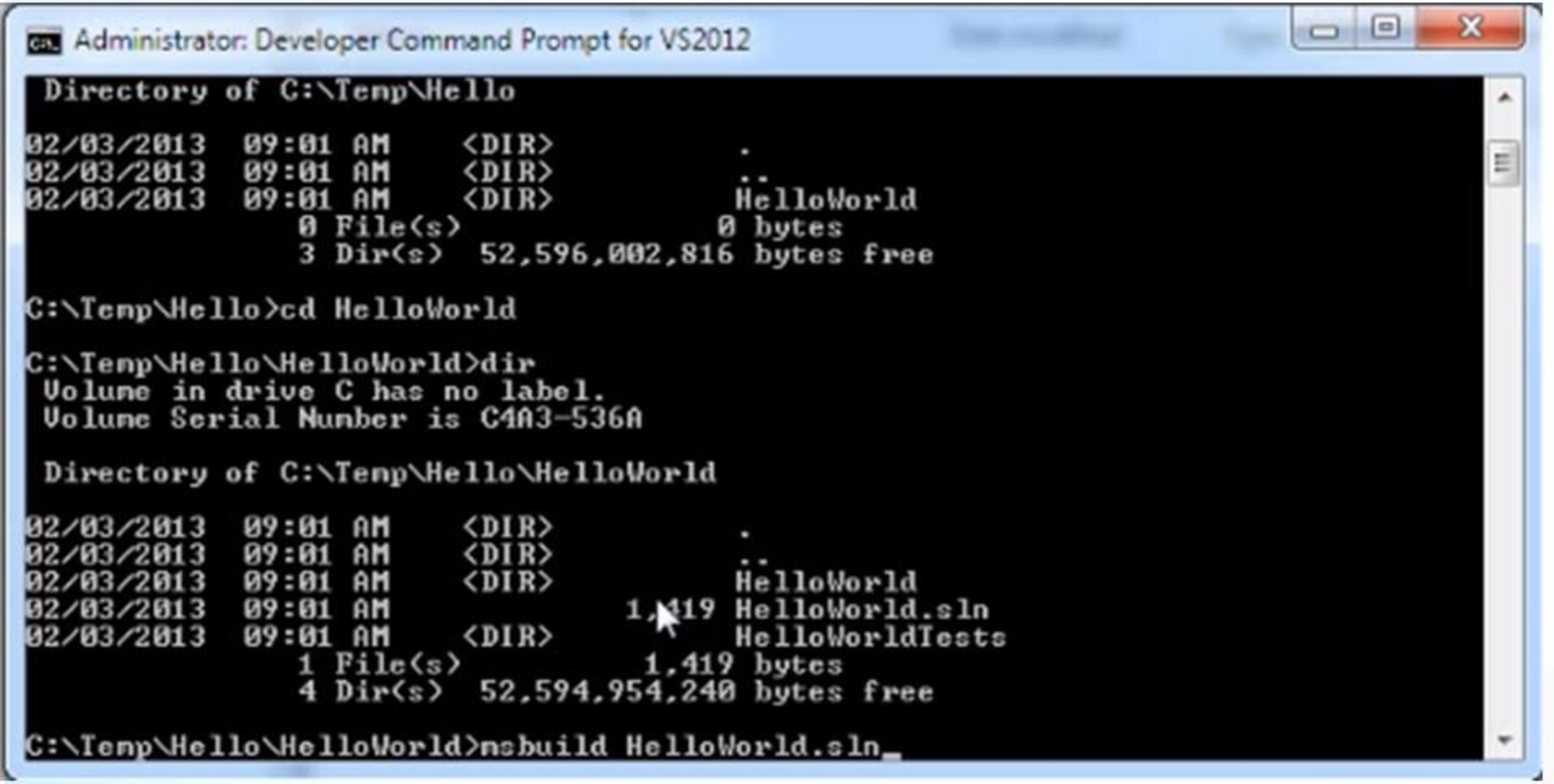
SkillAssure

Manually Building



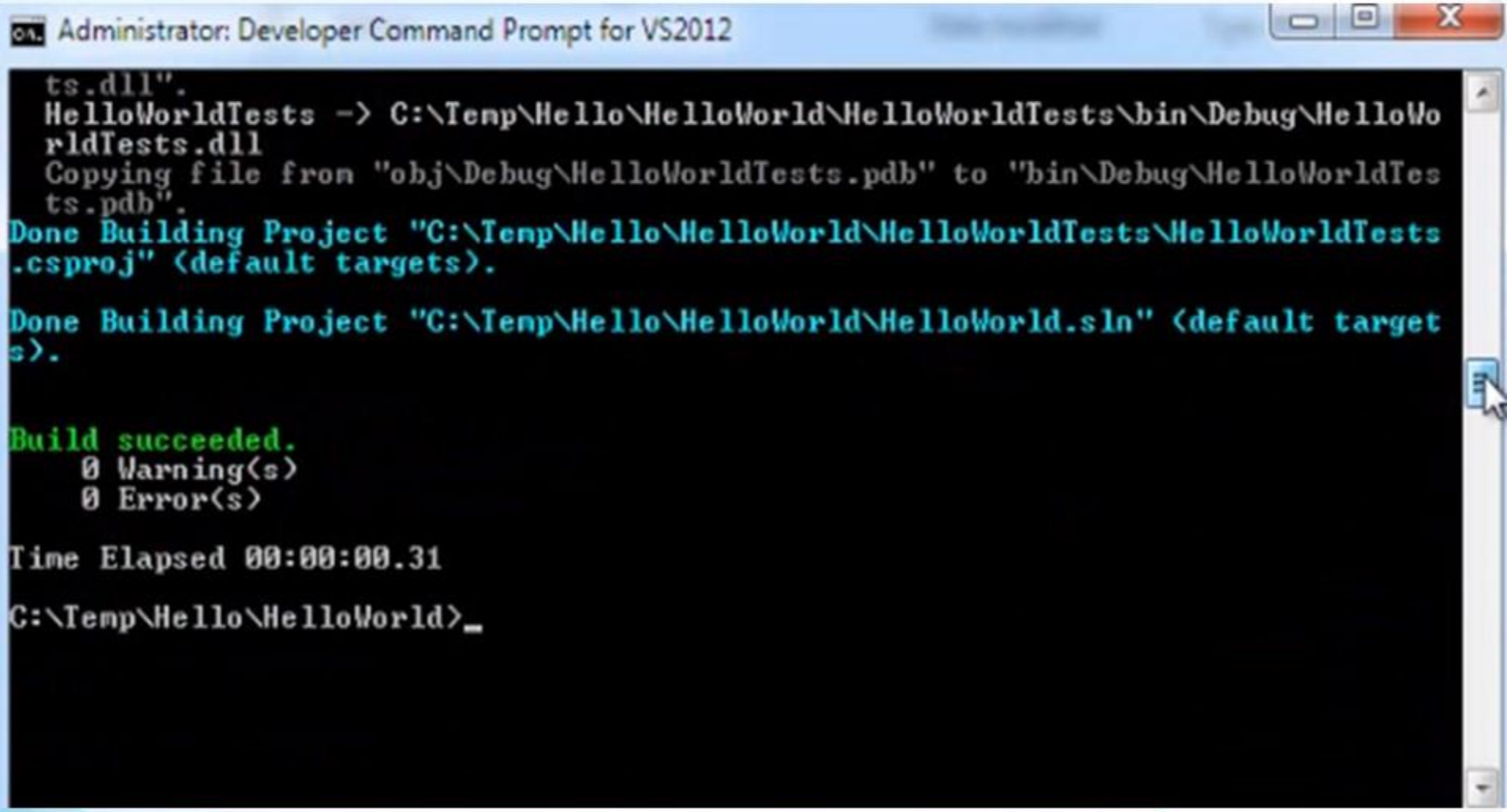
SkillAssure

Manually Building



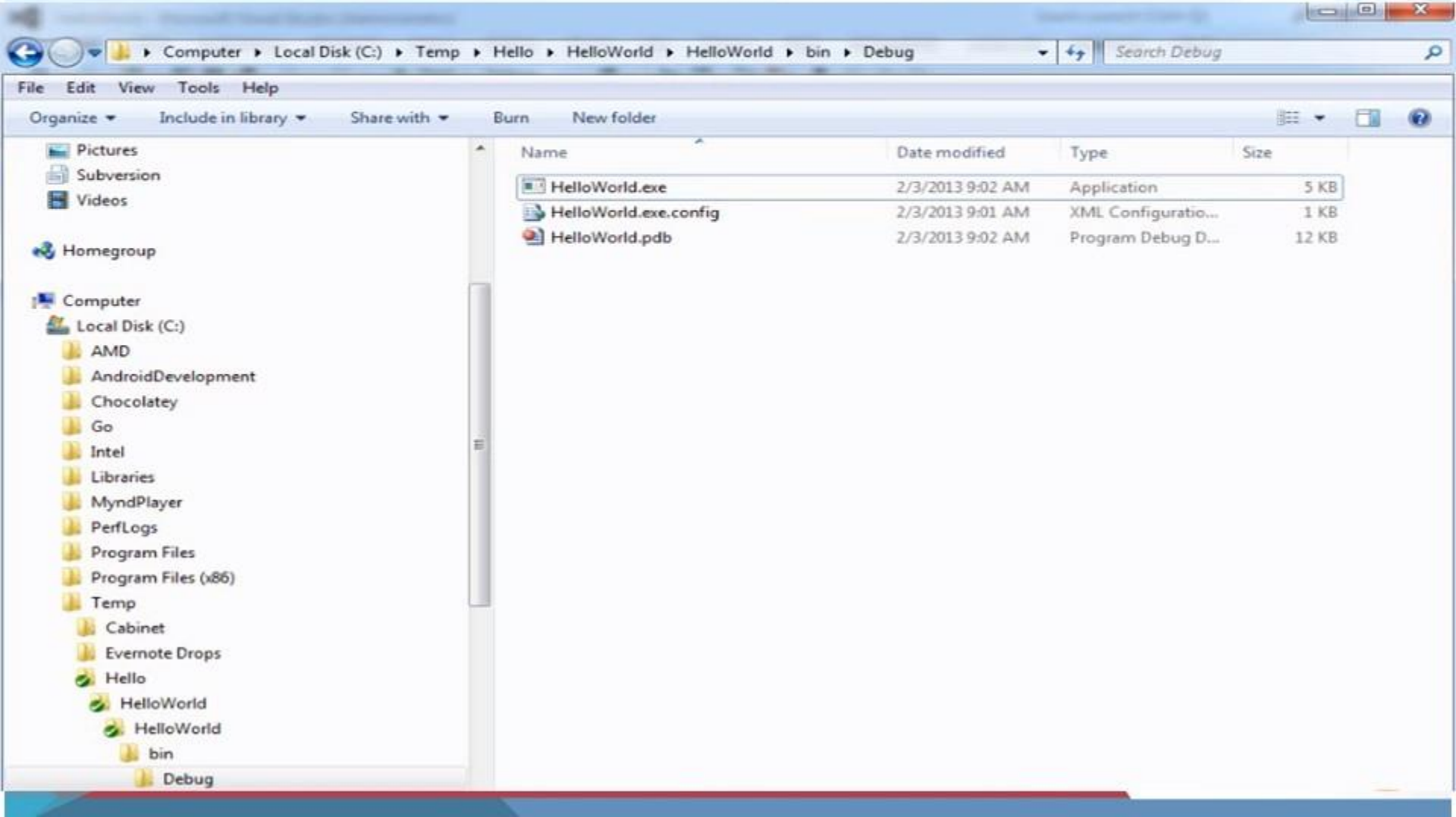
SkillAssure

Manually Building



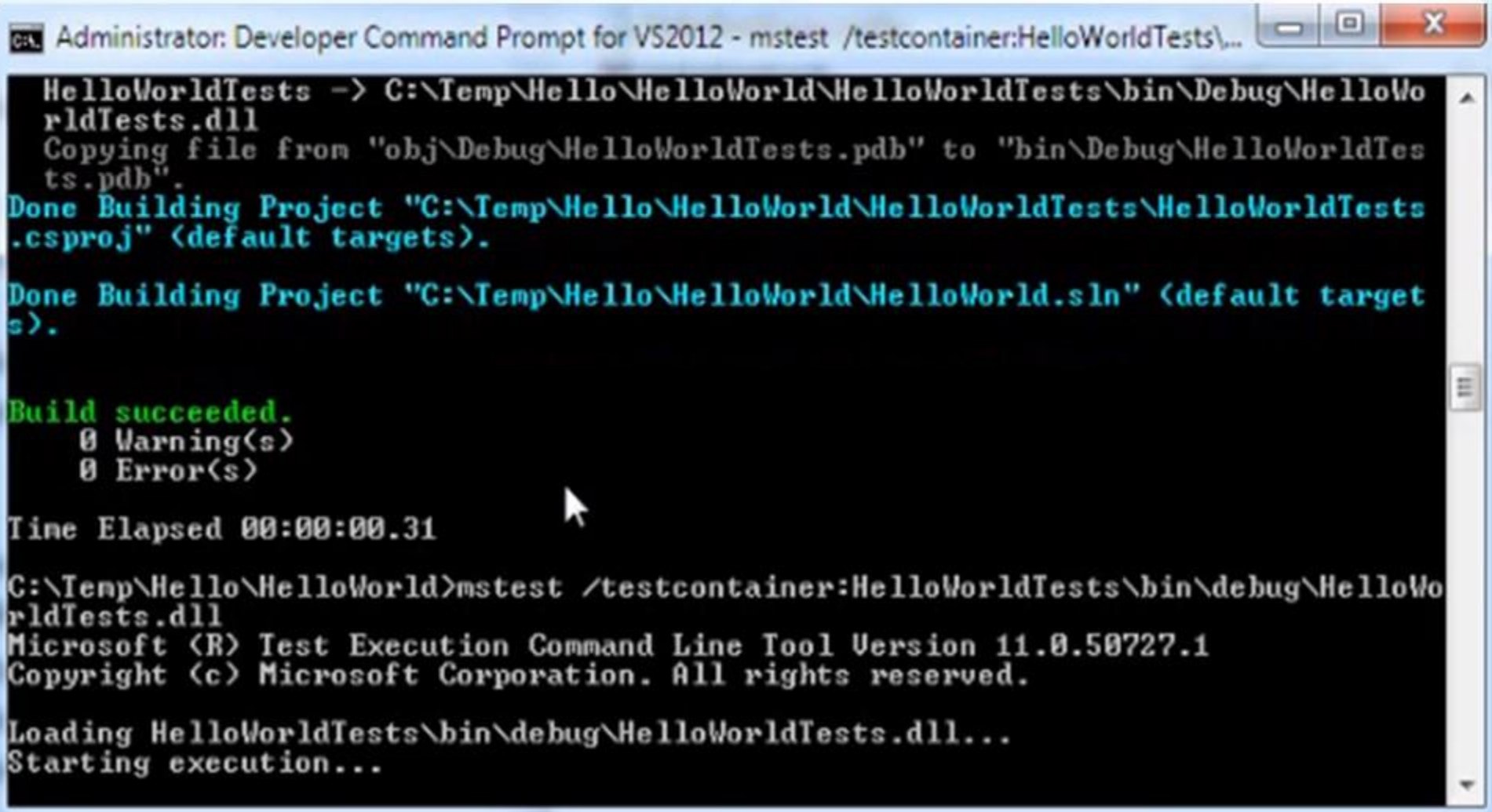
SkillAssure

Manually Building



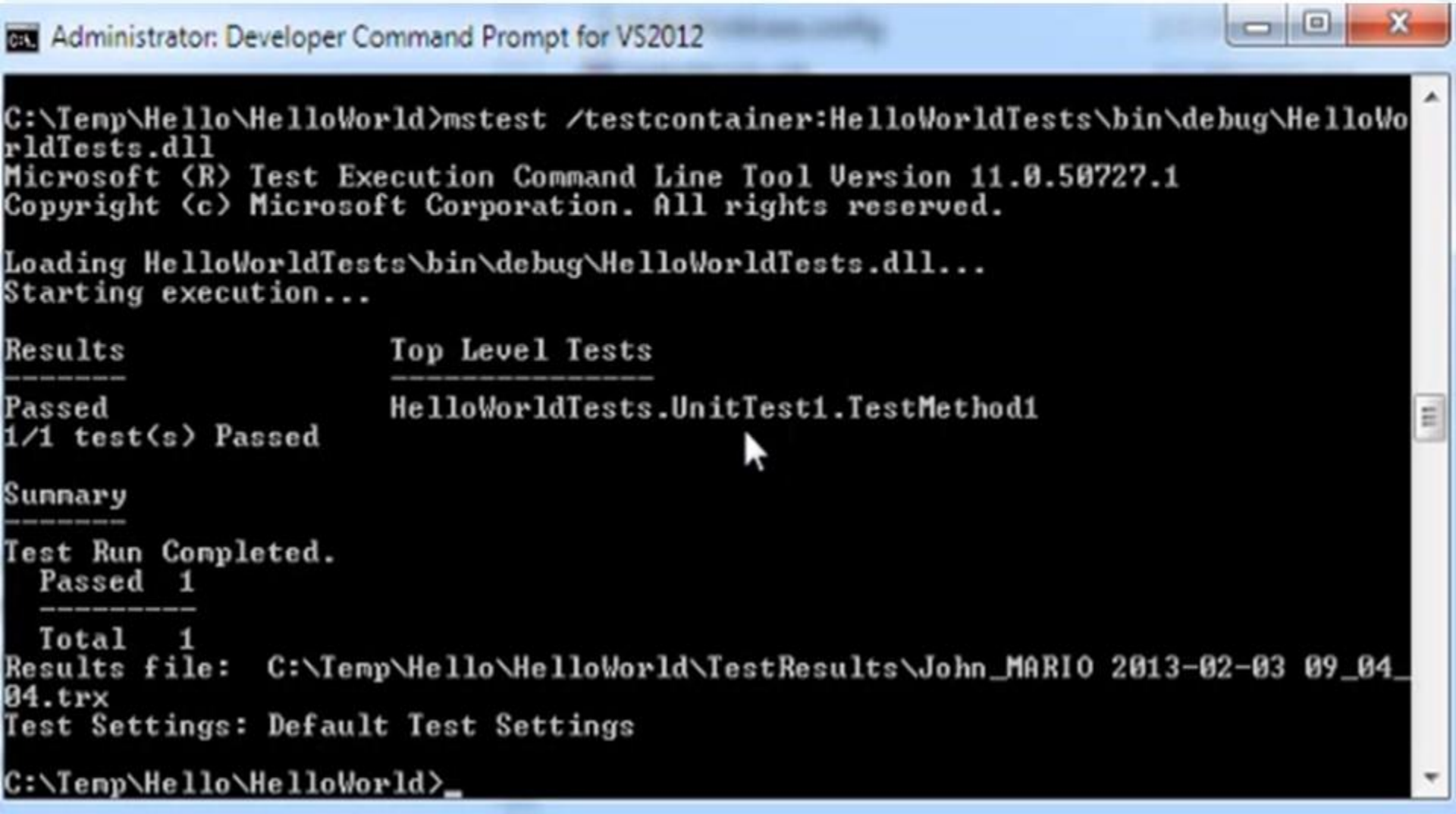
SkillAssure

Manually Building



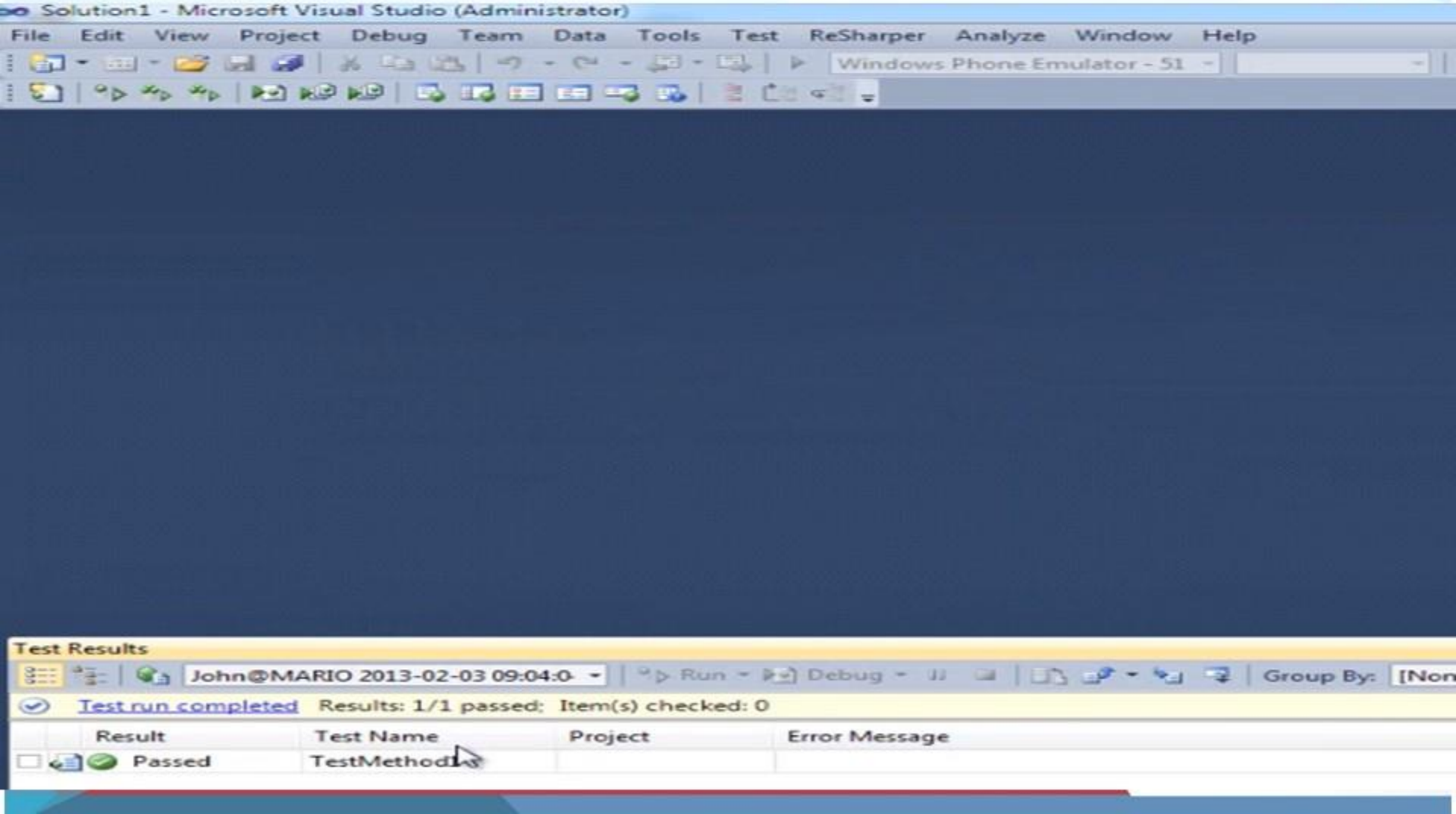
SkillAssure

Manually Building



SkillAssure

Manually Building



SkillAssure

Build Triggers

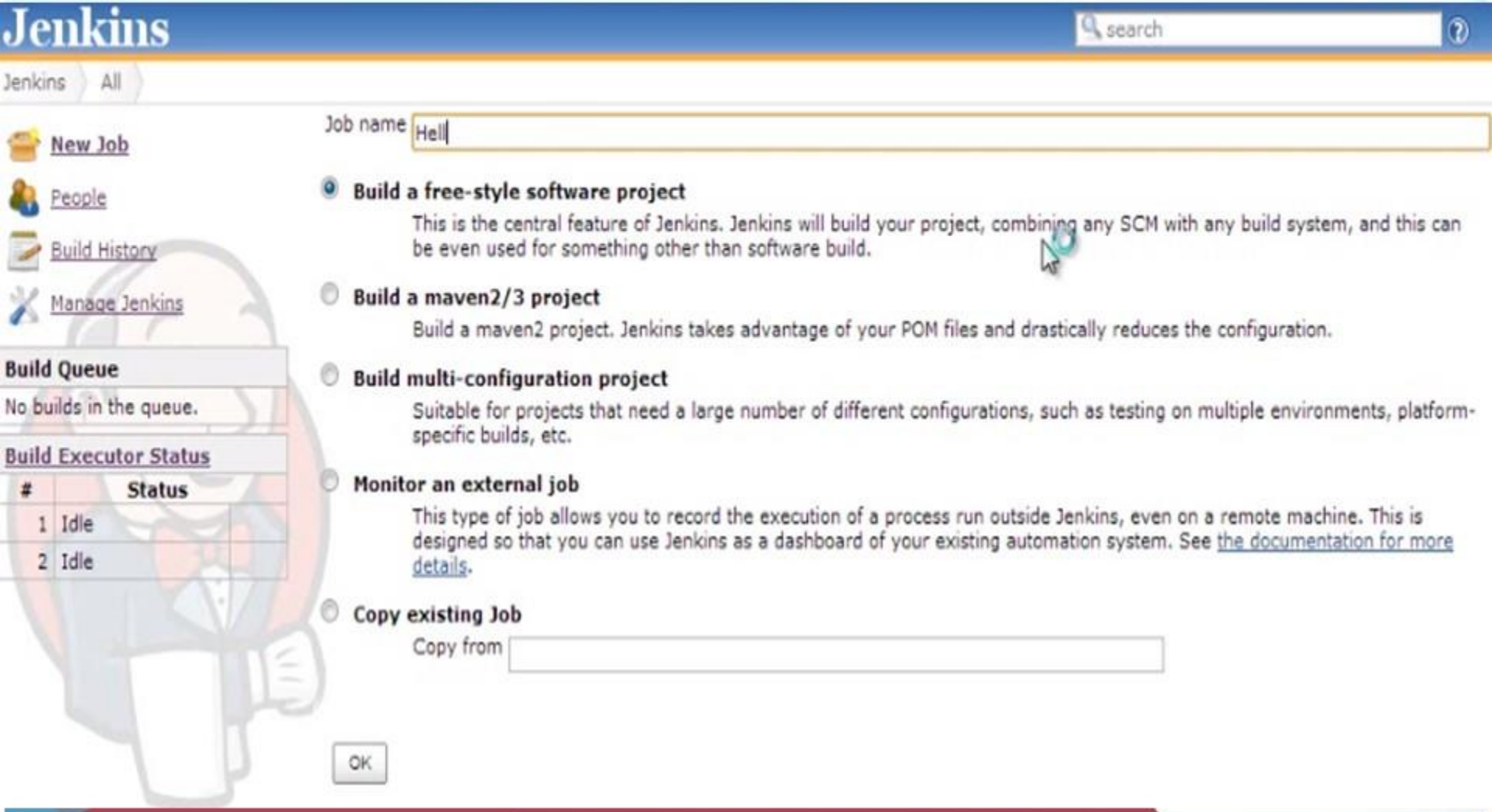
After other projects

Periodically

Poll source control

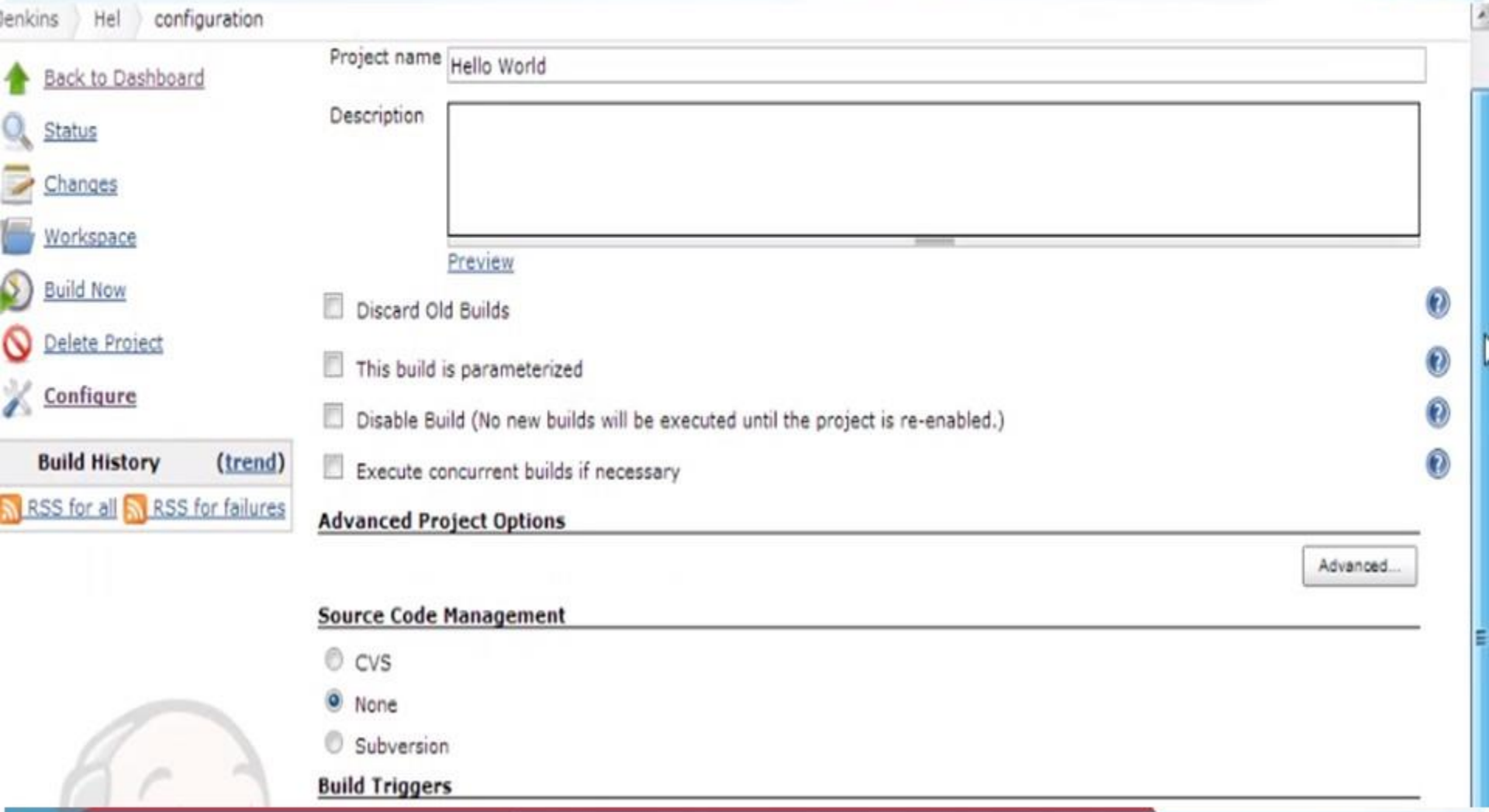
SkillAssure

Creating a job



SkillAssure

Creating a job



SkillAssure

Creating a job

Source Code Management

- CVS
- None
- Subversion

Modules

Repository URL: https://mario/svn/Test/Hello/trunk

Local module directory (optional): .

Add more locations...

Check-out Strategy: Use 'svn update' as much as possible

Use 'svn update' whenever possible, making the build faster. But this causes the artifacts from the previous build to remain when a new build starts.

Repository browser: (Auto)

Advanced...

SkillAssure

Creating a job

Build Triggers

- [ ] Build after other projects are built
- [ ] Build periodically
- [x] Poll SCM

Schedule

This field follows the syntax of cron (with minor differences). Specifically, each line consists of 5 fields separated by TA whitespace:

```
MINUTE HOUR DOM MONTH DOW
```

MINUTE Minutes within the hour (0-59)
HOUR The hour of the day (0-23)
DOM The day of the month (1-31)
MONTH The month (1-12)
DOW The day of the week (0-7) where 0 and 7 are Sunday.

To specify multiple values for one field, the following operators are available. In the order of precedence,

- '*' can be used to specify all valid values.
- 'M-N' can be used to specify a range, such as "1-5"
- 'M-N/X' or '*/X' can be used to specify skips of X's value through the range, such as "*/15" in the MINUTE field "0,15,30,45" and "1-6/2" for "1,3,5"
- 'A,B,...,Z' can be used to specify multiple values, such as "0,30" or "1,3,5"

To allow periodically scheduled tasks to produce even load on the system, the 'H' token can be used. For example, pec often use '0 0 * * *' for a daily job, but this ends up causing a large spike in midnight. In contrast, doing 'H H * * * would still execute a job once a day, but the actual time of the day this gets executed will be spread over by Jenkins.

The 'H' token can be used with a range. For example, 'H H(0-7) * * *' means some time between midnight to 7:59a

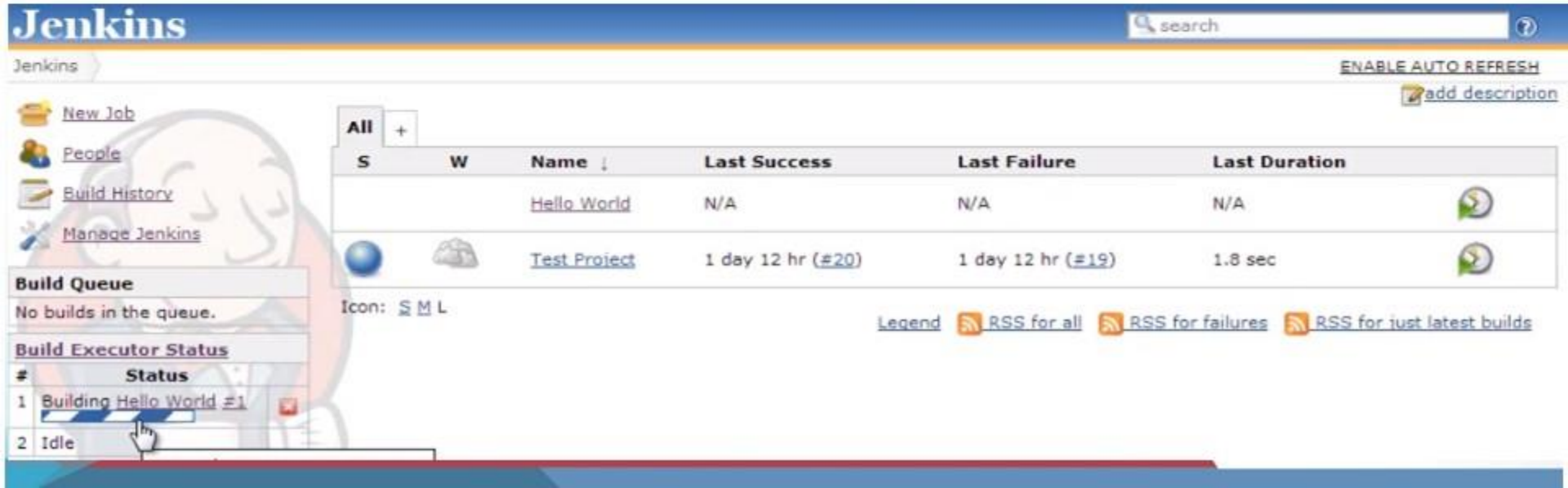
The 'H' token can be thought of as a random value over a range, but it actually is a hash of the job name, not a rando function, so that the value remains stable for any given project.

Empty lines and lines that start with '#' will be ignored as comments.

Save Apply

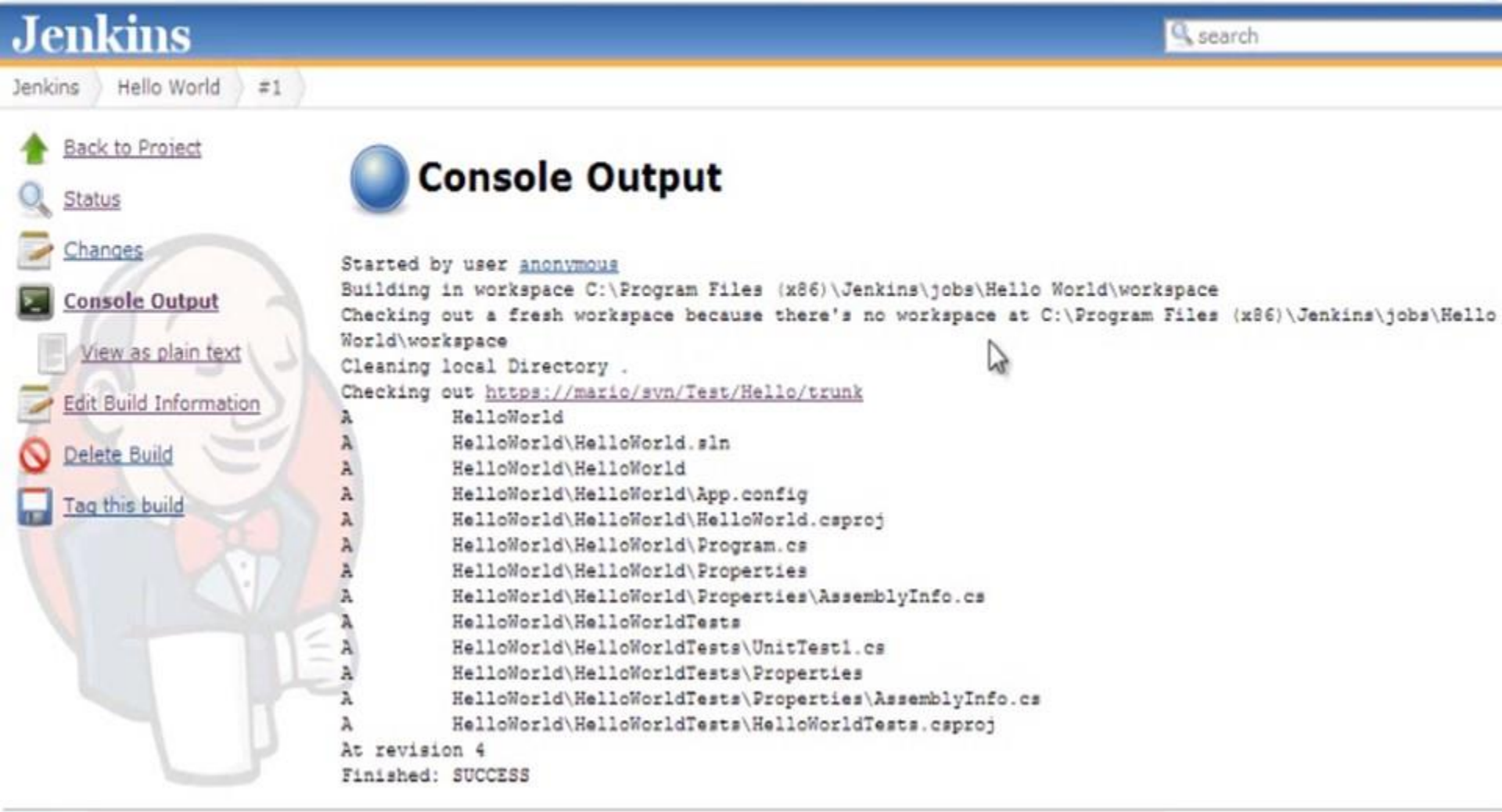
SkillAssure

Creating a job



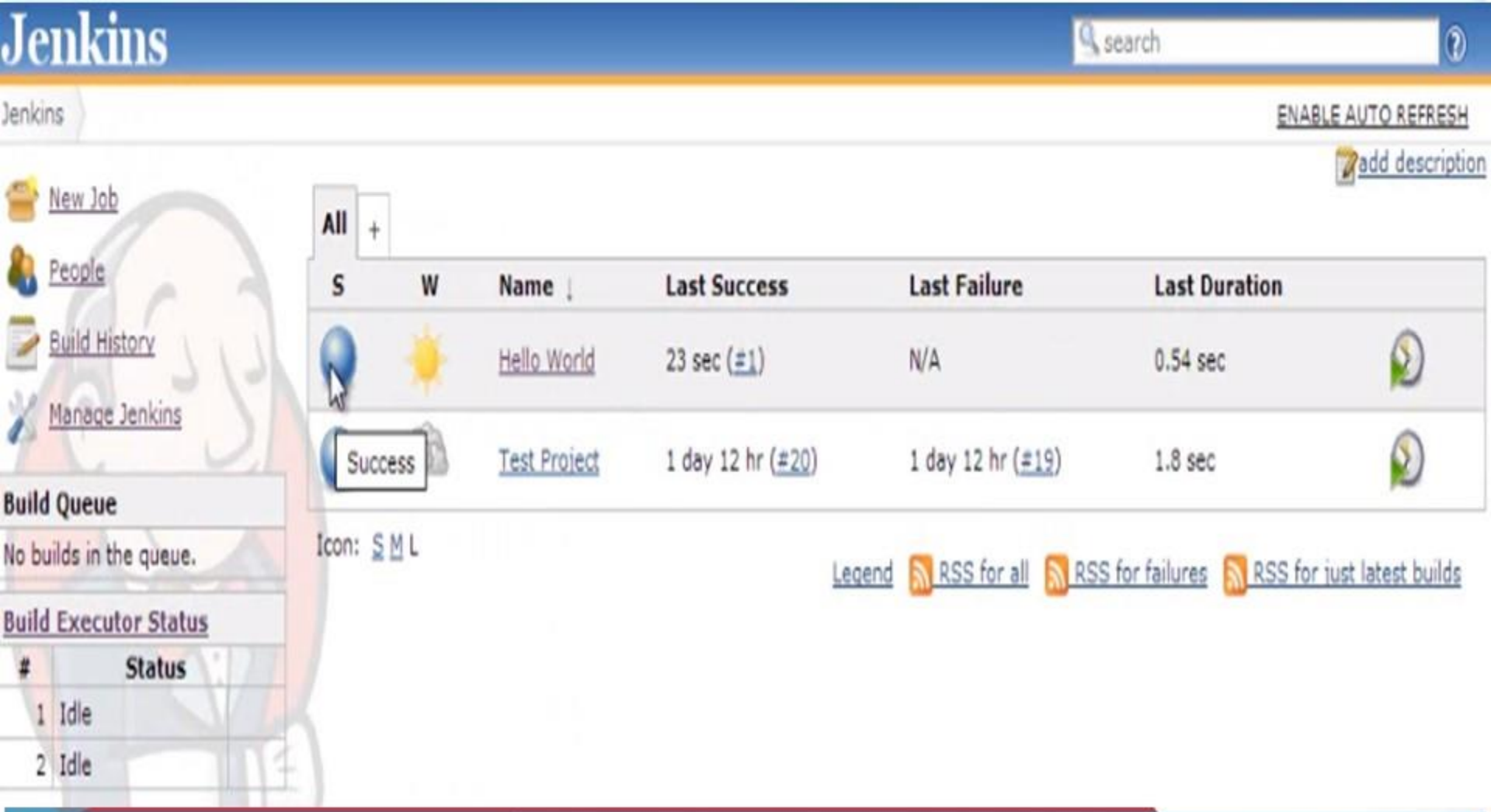
SkillAssure

Creating a job



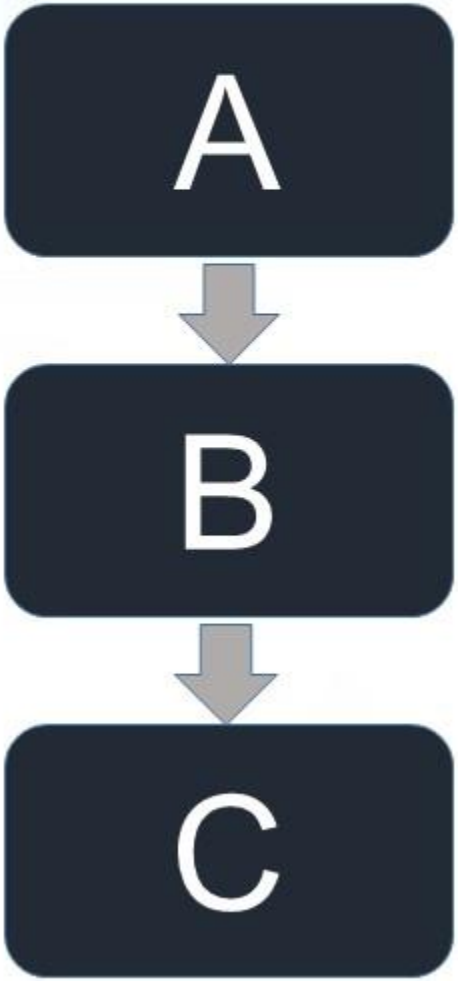
SkillAssure

Creating a job



SkillAssure

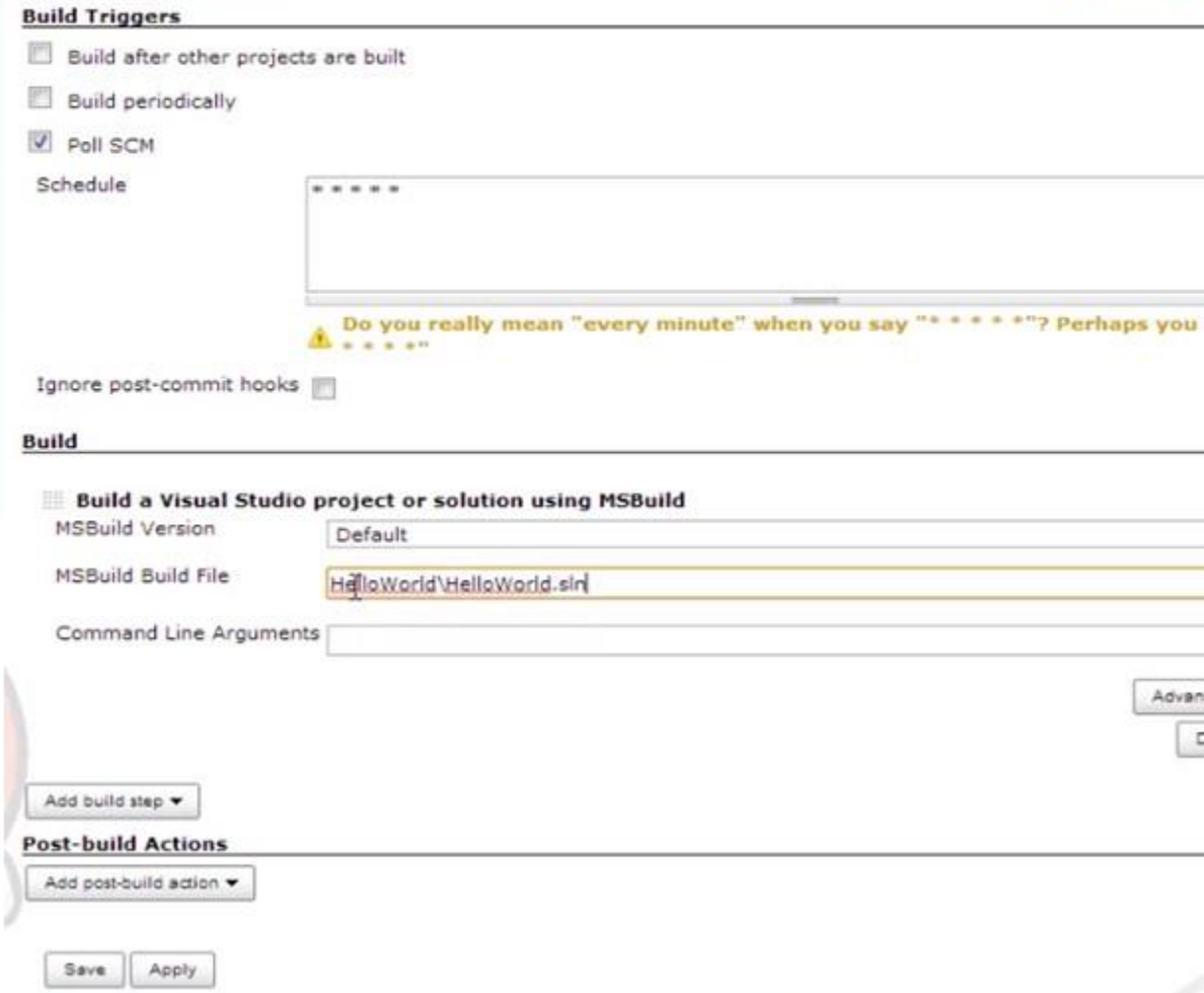
Build Steps



SkillAssure

First Build Step

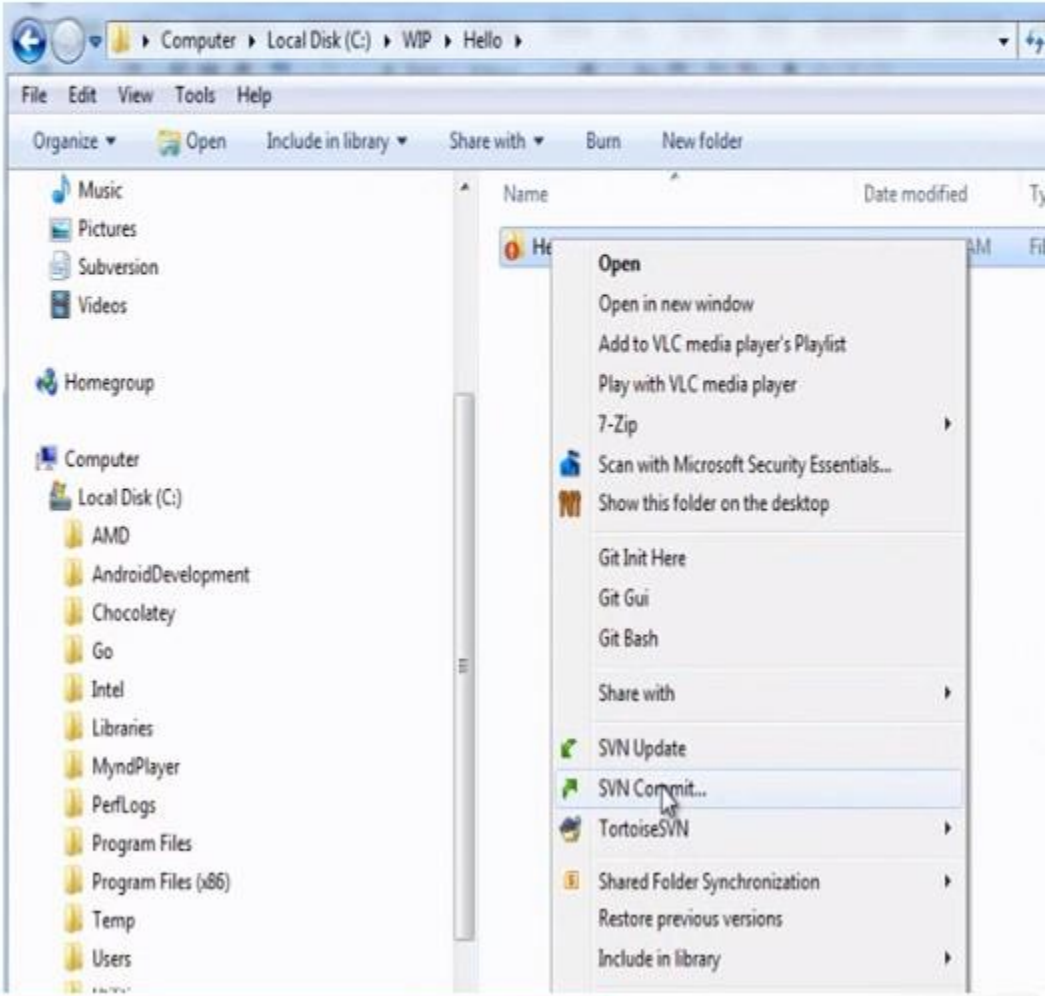
- Click Add Build Step
- Select Build Type
- Select MSBuild Version
- Add MSBuild build file (Solution File)
- Click Save



SkillAssure

Testing the Trigger

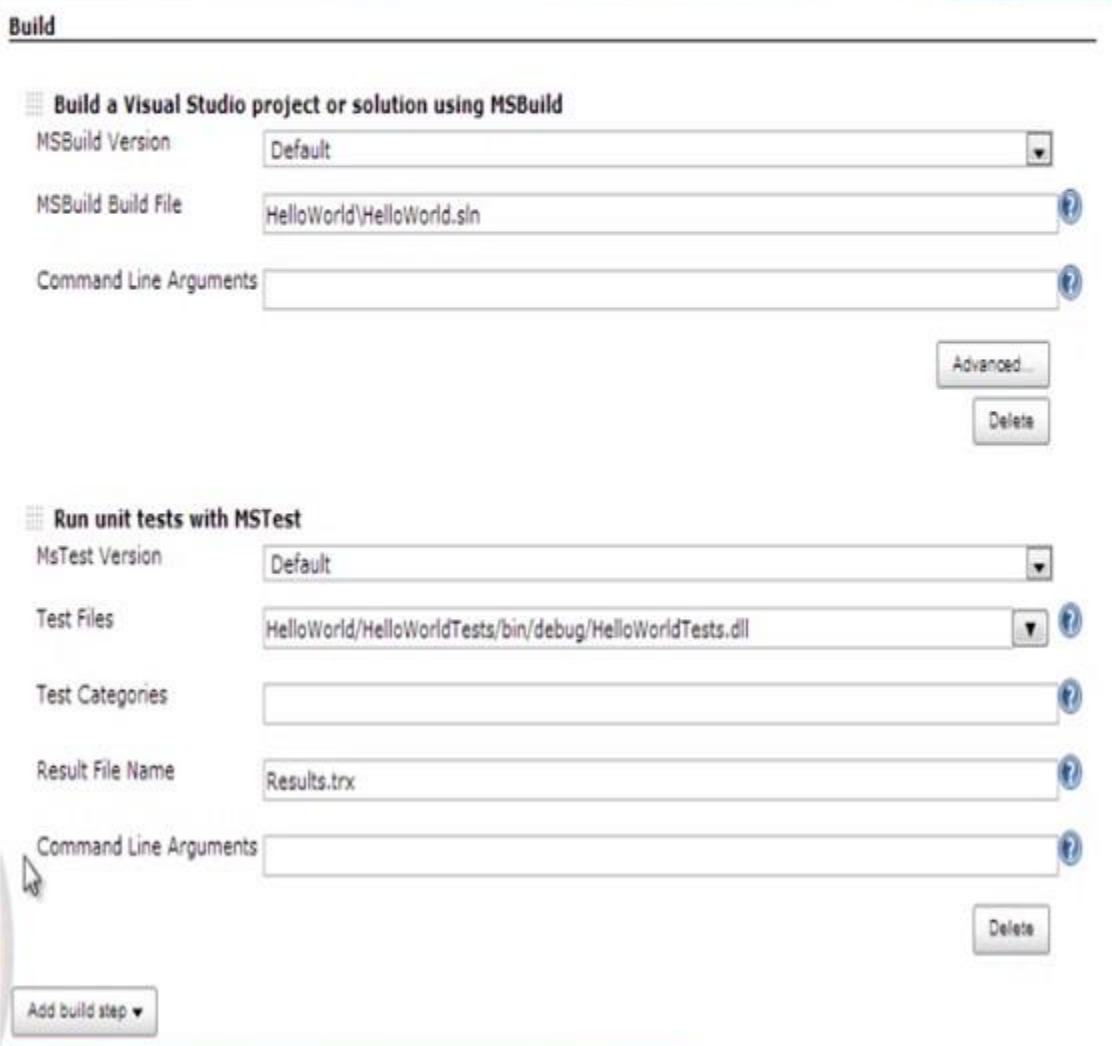
- Make some change in Code
- Enable Auto refresh in Jenkins
- Commit this change in SVN



SkillAssure

Add Tests

- Add Build Step
- Select Run Unit Test with MSTests
- Select MSTest Version
- Add Location of Test File
- Add Result File Name
- Click Save



SkillAssure

Workspaces

SkillAssure

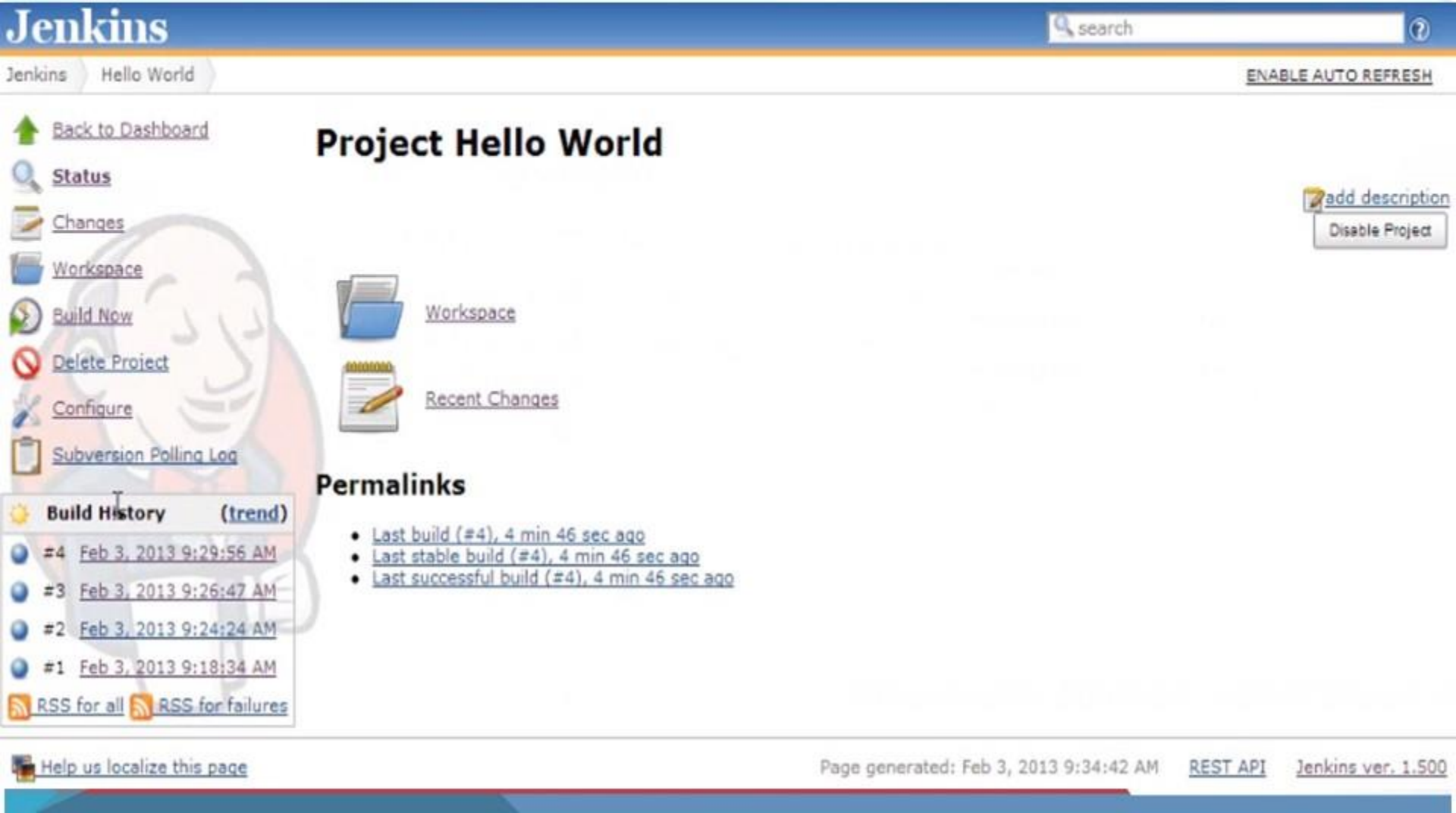
Post Build Actions

DONE!

Suzie's parents were so proud of her Successful Building of her enterprise application

SkillAssure

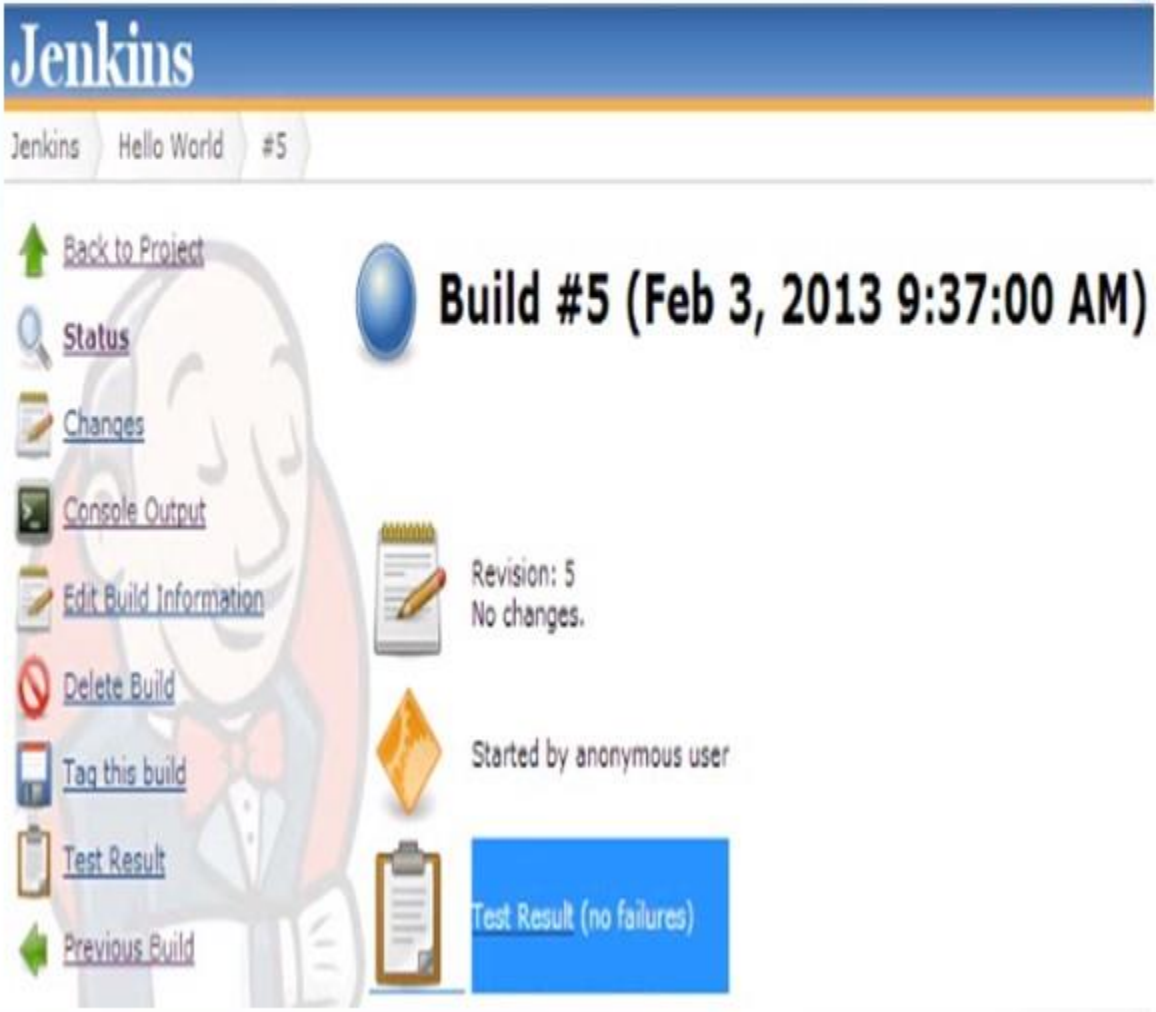
Reporting Test Results



SkillAssure

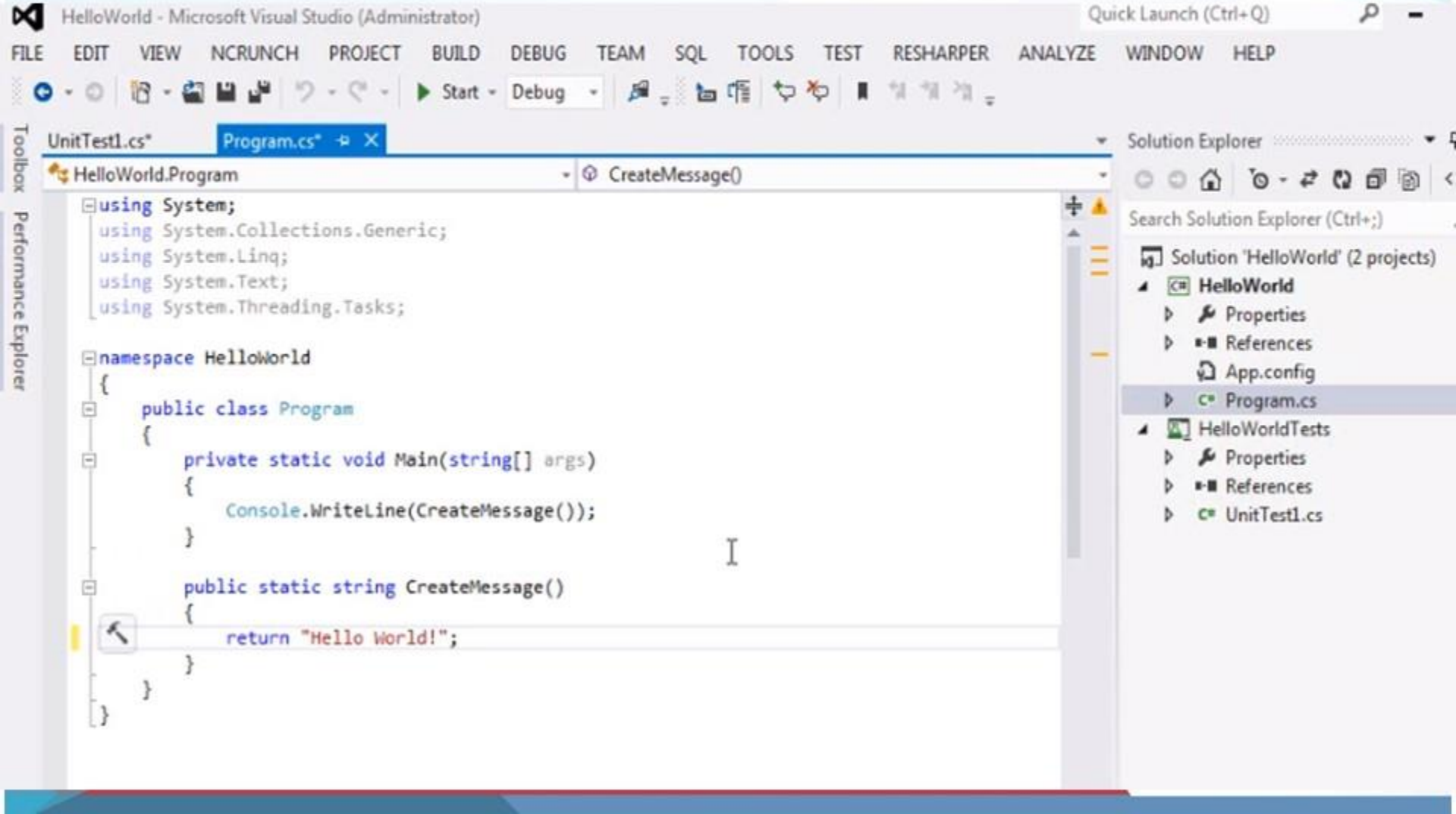
Reporting Test Results

- Go to Project
- Select Configuration
- Click Add post-build action
- Select Publish MSTest test result report
- Add Test report file name
- Click Save



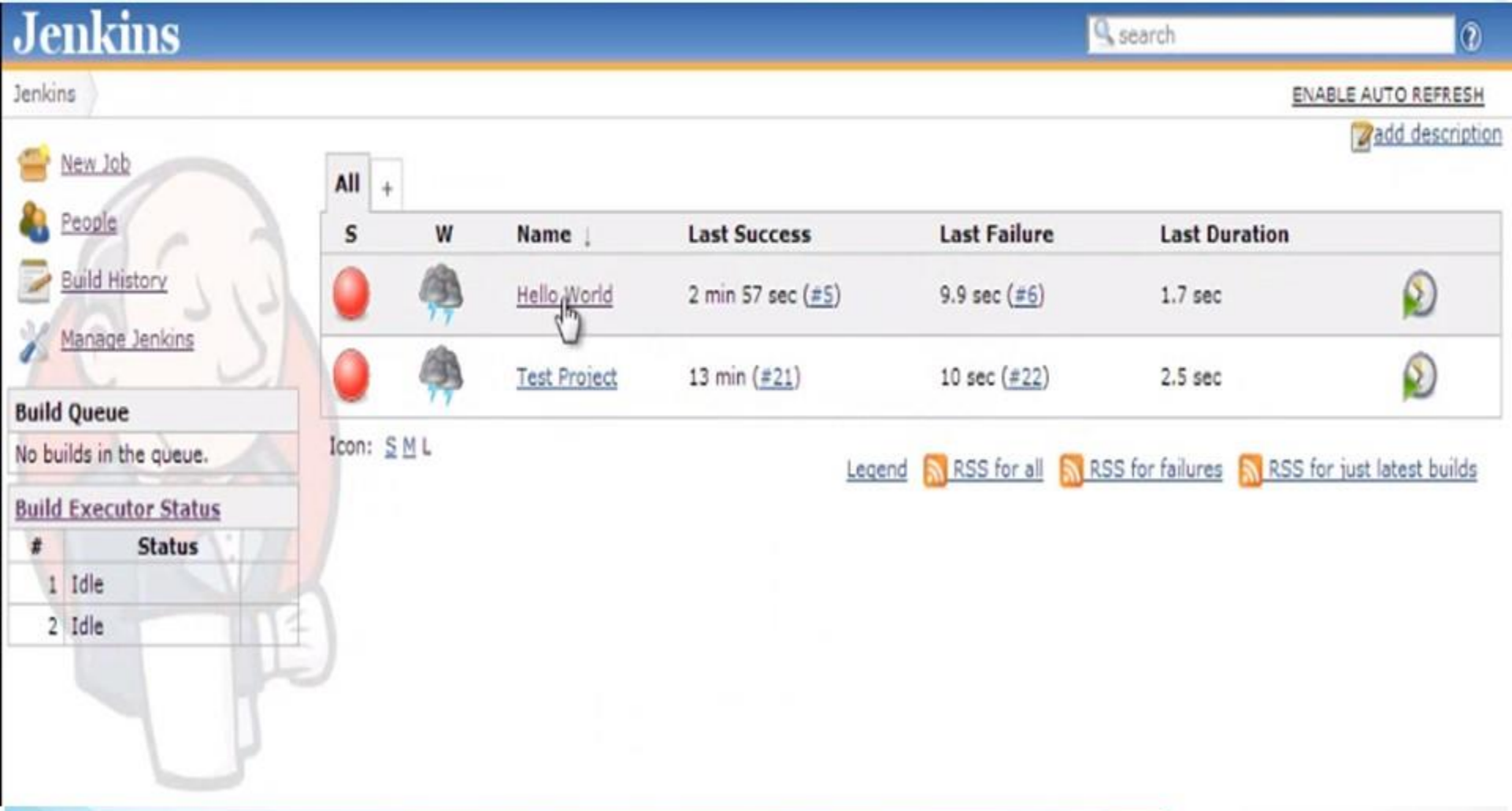
SkillAssure

Failing the Build



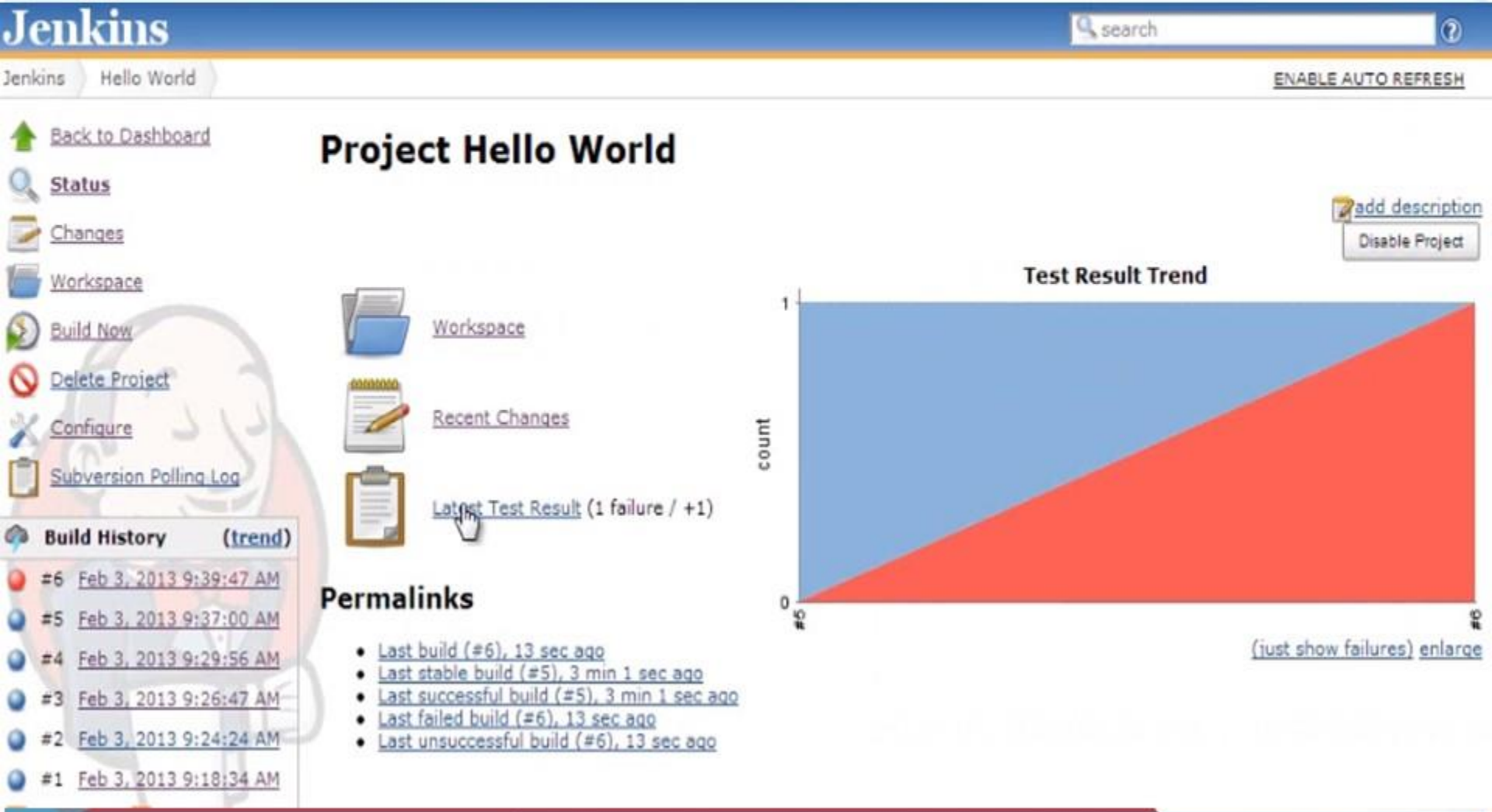
SkillAssure

Failing the Build



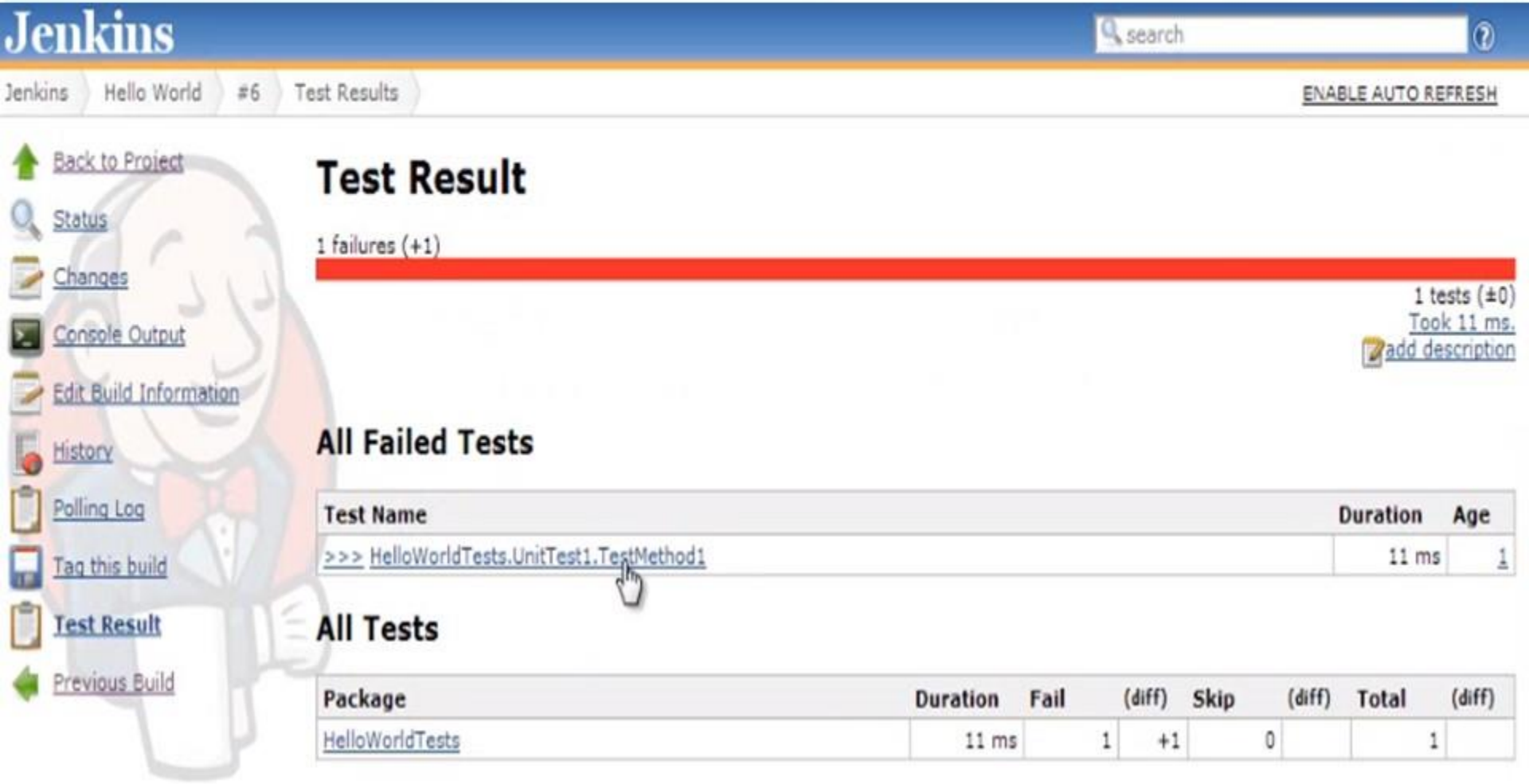
SkillAssure

Failing the Build



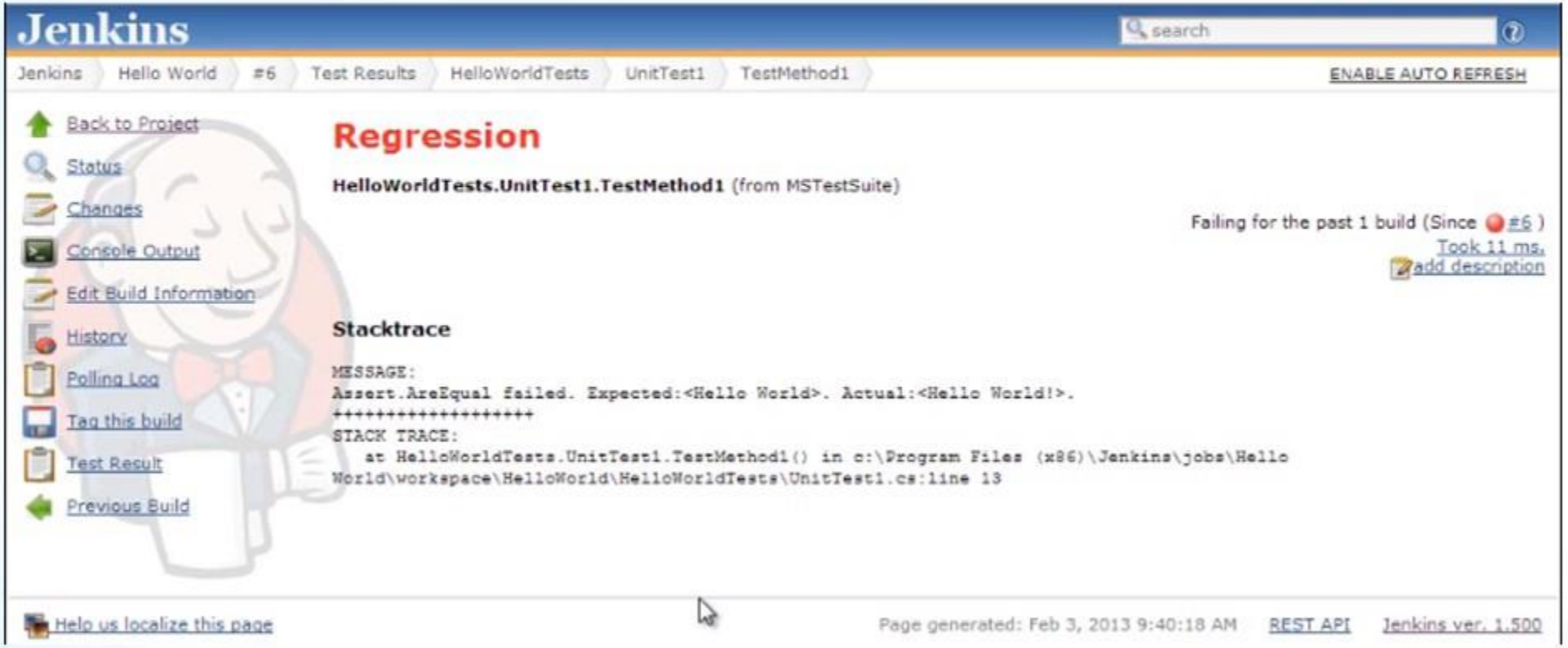
SkillAssure

Failing the Build



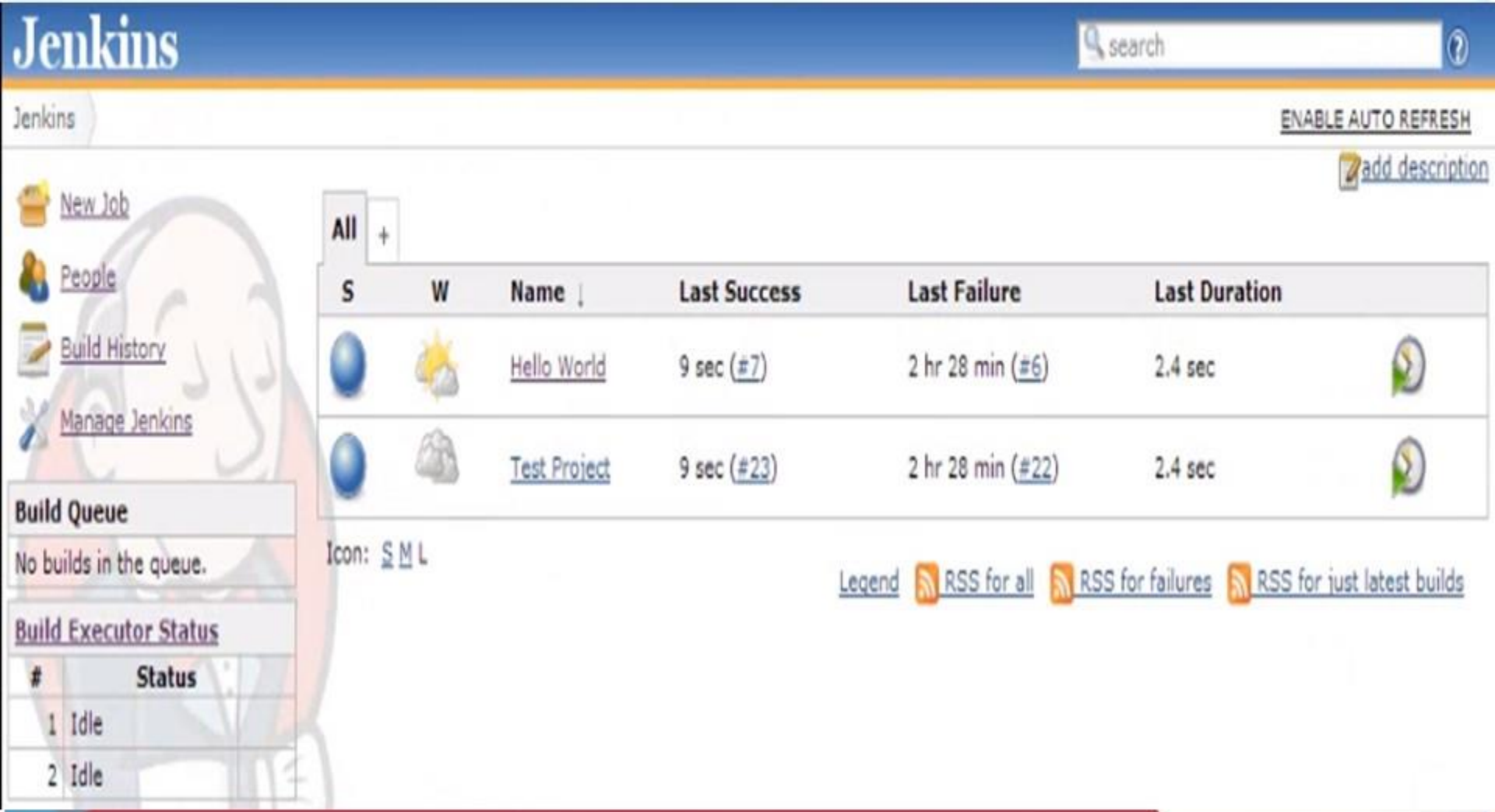
SkillAssure

Failing the Build



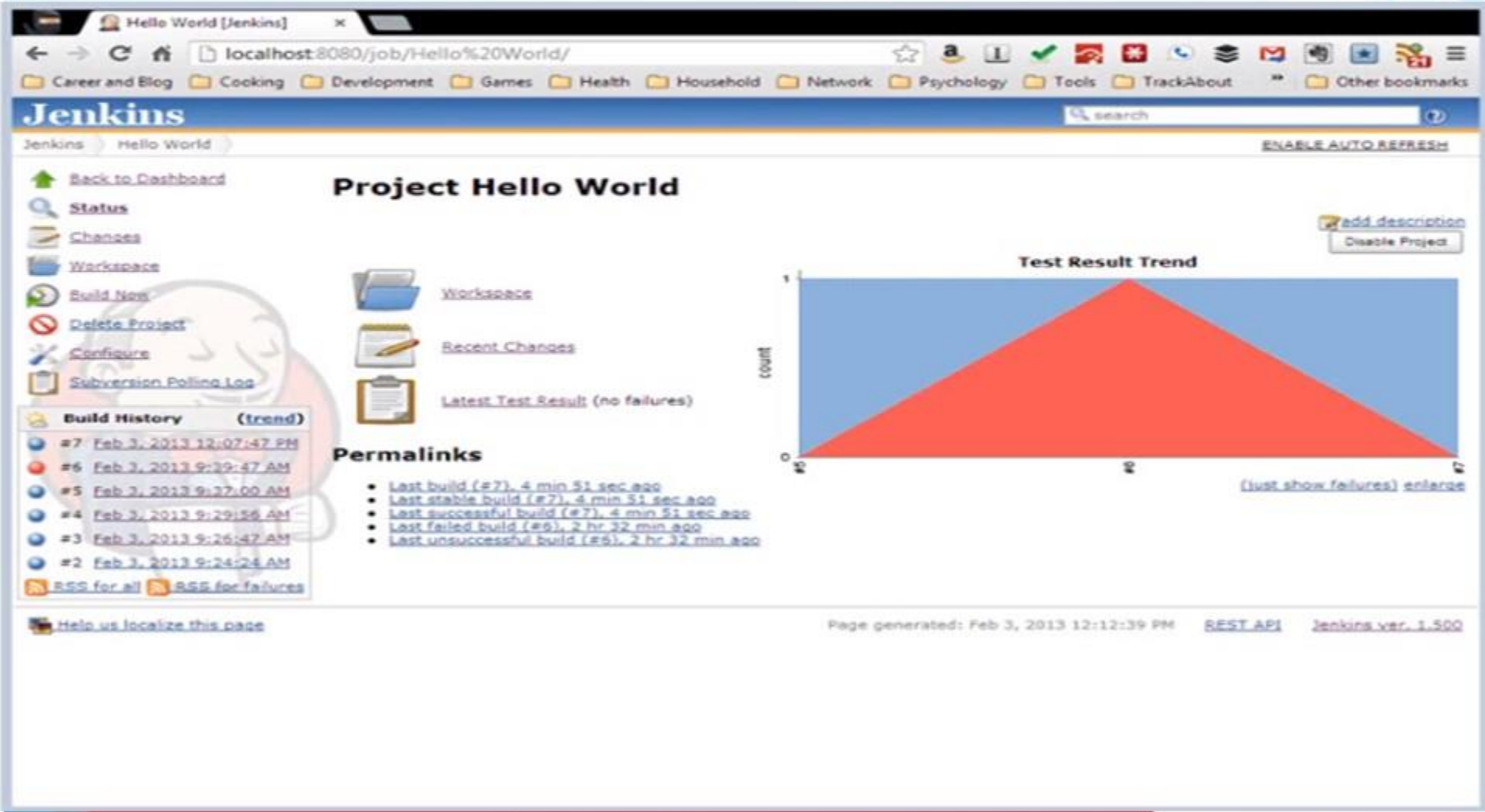
SkillAssure

Fixing the Build



SkillAssure

Fixing the Build



SkillAssure

Exercise

- Run a Jenkins Job in Ubuntu Agent and print Build Number.
- Run a Jenkins Job in Windows Agent and print Build Number.

Exercise

- Configure a single Jenkins job to make the release from multiple branches (e.g. trunk or feature branches)

Exercise

- Checkout a Project from SVN.
- Checkout a Project from Github.
- Checkout a Project from GitLab.

SkillAssure

Exercise

- Integrate JAVA with Jenkins.
- Integrate Maven with Jenkins.
- Enable Jenkins to send email notifications.
- Build a Maven Project in Jenkins.
- Build a .NET Project in Jenkins.

Exercise

- Integrate Jenkins & SonarQube.
- Perform SonarQube analysis for a Java project using Jenkins as CI Tool.
- Perform SonarQube analysis for a .NET project using Jenkins as CI Tool.
- Run recurring SonarQube analysis for any of the above project every mid-night.
- Save the reports of recurring SonarQube Analysis by associating it with Jenkins Build number.
- Make it optional for user to run SonarQube Analysis or not.

Exercise

- As a DevOps Engineer I did setup a lot of jobs in Jenkins. Accidently I lost all the jobs. It took me 5 Days to recreate the jobs. What solution do you propose so that it does not take so much time if it happens again?
- As a user I want the build to be triggered at a specific schedule. The schedule of the build is random. What solution do you propose to my problem?
- Looking at the console output it becomes difficult to differentiate between error and normal message. What solution do you suggest?
- How do you create a new job which is an exact copy of existing job?

SkillAssure

Exercise

- On completion of 1st job trigger 2nd job.
- Make it optional for user if he wants to trigger 2nd job or no.
- Print the Build number of 1st job in 2nd job

Thank You

SkillAssure

Exercise

- Project: HappyTrip
- Source Control: GIT
- Application Server: Tomcat

- Problem Statement:
  - Step 1: Get the HappyTrip Project from GIT
  - Step 2: Build the HappyTrip Project.
  - Step 3: Poll the SCM and build the project every minute.
  - Step 4: Send an email notification at the end of the build
  - Step 5: Deploy the HappyTrip Project to Tomcat Server.